Quantum Algorithms from the Near to Future Term

Gregory Boyd

Merton College

University of Oxford

Supervised by
Professors Simon C. Benjamin and Bálint Koczor

Doctor of Philosophy

Hilary Term 2025

Acknowledgements

Firstly, I must extend my gratitude to my supervisors Simon Benjamin and Bálint Koczor for providing me so much of their time to assist me over the course of my Doctorate, giving guidance on interesting research topics, directions of investigation, technical comments and help with research code, they have both made my Doctorate an extremely enjoyable experience, and I cannot thank them enough for their contributions to this thesis.

I must also extend my heartfelt gratitude to the University of Oxford and EPSRC, who provided the funding that made this research possible.

I thank the members of my research group who have provided excellent advice and support over the course of my thesis, in particular I am grateful to senior group members who gave me advice along the way, Cai, Daniel, Sam, Adrian, Richard, Arthur, Hans, and Cica, as well as my contemporary Matt who acted as a first-class technical and emotional sounding board, as well as keeping me on my toes by asking me questions I did not know the answer to.

I also wish to thank the QTDA team at Quantinuum, Steven Herbert, Ismail Akhalwaya, Adam Connelly and Julien Sorci for making me feel a welcome and useful part of the team, as well as providing guidance in my development during my internship there, and particularly to Steven Herbert for reminding me that the real quantum advantage is the friends we make along the way.

On a personal level, there is my family who have stalwartly endured many hours of monologue on quantum computation, provided a safe environment from me to get away from work when I felt overwhelmed, and sent me motivating and delicious care packages.

I want to thank my friends in Oxford: Aleksy, Ryan and Izzy for making my time there so enjoyable. In particular, I should thank Aleksy for his culinary tuition and his (much needed) suggestions that I get a grip.

Max Langtry also deserves a mention for the many cups of tea, advice, and unparalleled ability to act as a rubber duck. He has greatly contributed to me writing this thesis in a reasonable timeframe, despite his belief that I should be much more relaxed about work.

Lastly, and most importantly, I have to thank Athena Stamper, without whom none of this work would not have been possible, for providing endless love, support and motivation.

Abstract

Many important physical and mathematical problems are difficult to solve on classical computers, resulting in difficulty in finding new advanced materials, simulating quantum field theories and solving large combinatorial optimisation tasks. Quantum computers have shown great promise in being able to solve these (and many other) problems faster than their classical counterparts, and therefore should be thoroughly investigated as a significant tool for enabling the acceleration of scientific progress.

This work develops methods to tackle several problems across the spectrum of quantum algorithms, providing methods suitable for the near-term hardware, to the future where large fault-tolerant quantum computers exist. Therefore, the works in this thesis range from efficiently making use of limited quantum hardware to taking advantage of hardware benefits (e.g. excess qubits) as quantum computation advances.

Covered topics include: an alternative paradigm for the optimisation of variational circuits and quantum neural networks; techniques for improving energy estimation of quantum states through the collection and processing of large amounts of randomised measurement data; the use of entanglement in guiding the automatic discovery of quantum circuits; investigating the role of symmetries in adiabatic state preparation of molecules and the large scale parallelisation of a common fault-tolerant subroutine, potentially producing large runtime speed-ups for the simulation of chemistry on quantum computers.

Although there are many challenges remaining from both hardware and algorithms perspectives before the large-scale utility of quantum computers can be demonstrated, the contributions of this work make steps towards the efficient utilisation of quantum hardware that will be available in the coming years.

Contents

List of Figures

1 Introduction

Contents

1.1 Motivation

It has been 40 years since the inception of quantum computation, an alternative computing paradigm able to solve computational tasks beyond the power of any classical machine [1], and we are closer than ever to the realisation of a universal quantum computing device. However, current hardware is severely restricted in terms of qubit count and error rates, prohibiting the use of Quantum Error Correction (QEC), which would allow near perfect execution of arbitrary-depth algorithms.

In the short term whilst we remain in the Noisy Intermediate-Scale Quantum (NISQ) [2] era, we are motivated to search for heuristic algorithms more suited for near-term

devices. These algorithms often rely on a hybrid quantum-classical optimisation loop, where a classical coprocessor aims to reduce the requirements on a quantum device, only using it to do classically hard tasks such as measuring quantum observables. It is still unclear if these algorithms will be able to provide useful applications of quantum computers, with a narrow window of applicability where hardware is sophisticated enough to run NISQ algorithms that are not classically simulable, but not sophisticated enough to run useful fault-tolerant algorithms.

Looking towards the future, even fully fault-tolerant algorithms that are able to deliver useful applications of quantum computers will face challenges, namely dauntingly high resource costs in terms of machine quality or runtime. Therefore, along with the hardware, further developments in fault-tolerant algorithms are also needed to bring forward the era of useful fault-tolerant quantum computing.

1.2 Contribution

This thesis will describe several algorithms ranging from near term, NISQ era, impact, to the optimisation of important subroutines for fault-tolerant algorithms. Firstly, in the NISQ regime, Chapter 2 presents an alternative optimisation paradigm for variational quantum circuits, providing several advantages over traditional gradient descent-based methods, including faster convergence and resilience against local minima. This work is a significant step forward in the training of these circuits and may well retain its usefulness through efficiently varying circuit parameters into the fault-tolerant era.

The work in Chapter 3 lies on the edge of NISQ and early fault-tolerance, providing techniques for improved energy estimation of quantum states via extremely high dimensional subspace expansions, enabled by randomised measurement data. As a technique that allows for improved performance at the expense of circuit repetitions, this is suited to the NISQ/ early fault-tolerant regimes where circuit depths are limited.

In Chapter 4 I present an optimisation of the common fault-tolerant subroutine of Linear Combination of Unitaries, allowing for arbitrary parallelisation of the

common situation where the unitaries to be applied are Pauli terms. As such, this work is suited to well into the fault-tolerant era, where more than the strictly minimum number of qubits required are available to speed up computation.
Finally, I present some ongoing work on exploiting entanglement measures in variational circuit compilation and the use of symmetries in adiabatic state preparation as a demonstration of symmetric guiding states outperforming other states with lower energy.

1.3 Quantum Computing: A Brief History

The idea of quantum computing was born from the desire to solve the classically intractable problem of simulating many-body quantum systems. Dirac wrote in 1929: *"The underlying physical laws necessary for the mathematical theory of a large part of physics and the whole of chemistry are thus completely known, and the difficulty is only that the exact application of these laws leads to equations much too complicated to be soluble."* [3]. Despite the widespread success of simulating weakly correlated systems with classical techniques, in general, the exponential growth of the number of degrees of freedom with the number of particles makes classical simulation of systems exhibiting strong correlations to reasonable precision impossible. In 1981, Feynman popularised the idea of using a controllable quantum system to act as a computer, allowing for the efficient simulation of many-body quantum systems out of reach of classical techniques (although this idea had been investigated by Paul Benioff a year before [4]).
In 1985, the idea was formalised by Deutsch and Penrose into the *universal* quantum computer, a controllable system capable of simulating any other quantum system [5]. For the first decade since its inception, quantum computation existed merely as an interesting thought experiment for physicists, but in 1992 Deutsch and Jozsa developed the first quantum algorithm with a provable exponential speed-up compared to any classical algorithm [6] demonstrating that quantum computers can solve some problems that are practically impossible for classical computers. In 1994, Peter Shor discovered another exponential speed-up for a problem of practical use.

Shor's algorithm [7] can determine the prime factors of a number in polynomial time, which is particularly important as the modern day RSA cryptosystem that underlies the security of much of digital communication is based on the assumption that no efficient algorithm exists for this problem. Naturally, this discovery led to a great deal of interest in quantum computing, both from physicists and computer scientists attempting to find more quantum speed-ups, and from governments with an interest in the use of Shor's algorithm for decrypting messages. With the discovery of a quadratically-faster quantum algorithm for database search by Grover two years later [8] the community became convinced of the usefulness of physically building a quantum computer and a great deal of attention turned towards how this might be achieved.

The primary practical barrier to constructing a quantum computer is the problem of decoherence [9], where a quantum system interacts with its environment and rapidly becomes entangled with environmental degrees of freedom, causing the state of the system itself to effectively become classical [10]. The timescale over which decoherence occurs is extremely short, seemingly requiring correspondingly rapid operation of quantum computers to outpace the destruction of entanglement. Unruh calculated that as the thermal decoherence timescale at 1K is on the order of 10^{-11}s, a quantum computer would have to operate at a frequency $\sim 10^{15}\mathrm{s}^{-1}$ in order to factor a 1000 bit number before decoherence destroys the calculation [11], that is, at frequencies millions of times greater than the GHz classical processors common today. To protect against decoherence, in 1996 Shor *et al.* produced a quantum error correcting scheme which used additional physical qubits to protect one logical qubit [12]. A series of developments in quantum error correcting codes (QECC) then occurred, leading to the *threshold theorem*, stating that as long as the error in individual gates is below a certain constant threshold, it is possible to correct errors to perform an arbitrary length quantum computation [13]. Kitaev proposed a topological QECC [14] with the benefits of a high threshold and only requiring local measurements for error correction which led to the surface code [15] which is the QEC basis of many of the current proposals for future error-corrected

quantum hardware. Despite these theoretical results confirming the plausibility of a machine that can act as an ideal quantum computer, there nevertheless remains the substantial challenge of actually building a device with a sufficient number of qubits and low enough error rates to implement QEC.

By 1998, the basic fundamentals of hardware had been demonstrated, with qubits being realised and 2-qubit gates being demonstrated in NMR experiments, and used to run a 2-qubit Deutsch-Jozsa algorithm [16]. By the mid 2010s, quantum computers based on ion traps were producing controlled entanglement between 14 qubits [17] and superconducting qubits had reached 2-qubit gate fidelities of 99%, approaching the threshold for QEC with the surface code [18]. In the early 2020s, hardware had improved to the point where prototype quantum computers could run a series of random sampling experiments believed to be out of the reach of classical computation [19–22], and many prospective architectures reached the threshold for the surface code using a small number of qubits. As of 2024, a series of impressive QEC experiments have been performed in hardware, with demonstration of below threshold logical qubits [23, 24], demonstration of more advanced non-local codes on architectures with non-local connectivity [25, 26], as well as demonstration of logical preparation of magic states [27] and early algorithms [28]. The primary hardware challenge remaining is how to scale up to the tens of thousands of qubits as the minimum that will be required for a useful application on a fault-tolerant quantum computer, while keeping gate fidelities high.

It may be possible to run useful applications on pre-QEC quantum computers: co-design of algorithms, compilation schemes, error mitigation and clever choice of problem may make it possible to demonstrate quantum advantage (when a quantum computer first performs a useful, classically intractable task) possible in the near-term future.

1.4 Quantum Logic

The basic operations of classical computers are logic gates, which act as binary functions that usually map two bits onto a single output (with the addition of the only

classical single-bit gate, the NOT gate). In a digital quantum computer, the basic information element is the qubit, which is a two-level quantum system whose states are usually denoted as $|0\rangle$ and $|1\rangle$: these states make up a 2-dimensional Hilbert space. A single qubit can exist in a superposition of these states as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ where α and β are complex numbers giving the probability amplitudes of the two states. The state of n qubits can be described by a 2^n-dimensional vector of complex amplitudes, one for each of the possible states of n bits.

Quantum states may also be in a mixture of these 'pure' states described above, for example due to random errors that may or may not have occurred, or as the description of subsystem of a system that may itself be in a pure state. In this case, they can be represented as, $2^n \times 2^n$ dimensional *density matrices* which are a probabilistic mixture of pure states $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$. Apart from when discussing noise in Chapters 2 and 3, and subsystems in Section 5.3 I will limit my discussion to pure states $|\psi\rangle$.

Quantum logical operations are described by unitary operators U (where $UU^\dagger = I$) that act on the computer's state as $|\psi\rangle \to U|\psi\rangle$. Fortunately, there exist basic sets of single and two-qubit unitaries that are sufficient to construct any n-qubit unitary (such a gate set is called *universal*) [29], so the act of programming a quantum computer is primarily the problem of constructing a series of gates to produce a desired unitary. A key property of unitaries is that they have an inverse and are therefore *reversible*, meaning that in contrast to standard models of classical computation, any unitary quantum circuit must be able to be run backwards to reproduce its inputs.

The other component of quantum logic is *measurement*, where the state of a qubit $|\psi\rangle$ is non-deterministically (and non-reversibly) collapsed onto the output $|0\rangle, |1\rangle$ with probability $|\langle m|\psi\rangle|^2$, providing the measurement outcome m. This measurement corresponds to the measurement of the X observable of that qubit, and although it is possible to measure any observable directly, usually other observables are measured by first performing a unitary transformation to rotate the system onto the correct basis such that standard measurement effectively measures the desired

observable. Measurement is of course required for obtaining any useful output from a quantum computation, as well as being central to QEC, but it can also be useful as a tool within algorithms. Despite the non-deterministic nature of measurement, it is sometimes possible to perform corrections based on measurement results, also known as *feedforward*: allowing for fast and deterministic circuits involving measurements [30–32]. Feedforward also allows for the teleportation of quantum information, transferring information from one qubit to another through the use of entanglement and classical communication [33].

A helpful way of presenting quantum logic is through quantum circuit diagrams, which lay out gates on lines denoting different qubits, with time proceeding from left to right. See Fig. 1.1 for an example (the gates in this circuit will be described below).

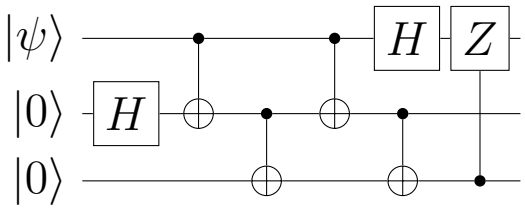
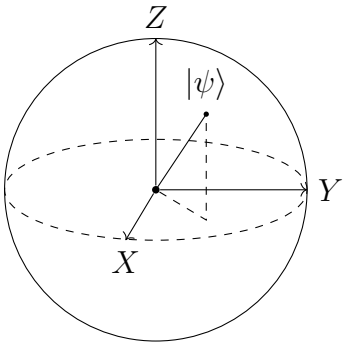


Figure 1.1: Example quantum circuit diagram, containing both single and two-qubit gates, and showing input states on the left.

The state of a single qubit can be represented as a point on the surface of a sphere and therefore the possible operations on one qubit can be described by rotations about three orthogonal axes.



$$R_X(\theta) = e^{\frac{-i\theta X}{2}}$$
$$R_Y(\theta) = e^{\frac{-i\theta Y}{2}} \qquad (1.1)$$
$$R_Z(\theta) = e^{\frac{-i\theta Z}{2}}$$

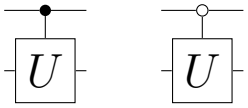
Where X, Y, Z are the Pauli operators, which are also logic gates.

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{1.2}$$

An example two-qubit gate is the controlled-NOT (CNOT) gate which flips the target qubit only if the control qubit in the $|1\rangle$ state.

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{1.3}$$

In fact, any unitary U can be promoted to a controlled operation through the notation:



Where the filled in circle indicates that the operation is performed if the control qubit is in state $|1\rangle$ and the empty circle for state $|0\rangle$.

The three Pauli operators, along with the CNOT, Hadamard (H) and phase (S) gates

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \tag{1.4}$$

construct the Clifford group, which is a non-universal gate set, and in fact is one that is efficiently simulable, i.e. it is possible to efficiently simulate a quantum algorithm that only uses these gates [34]. In order to obtain universality, one must additionally add one non-Clifford gate to your gate set, which is often chosen to be the $T = \sqrt{S}$ gate, leading to the Clifford+T gate set.

Unfortunately, there is a no-go theorem that prohibits any universal gate set being fault-tolerantly implementable in any QEC [35], for example in the surface code, T gates cannot be implemented fault-tolerantly. This hurdle can be bypassed using expensive magic-state distillation protocols [36], but this means that in the error-corrected regime, T gates are much more expensive than any other operation, meaning algorithms must be designed to limit the number of T gates required. Conversely, in the NISQ regime, arbitrary single-qubit rotations are possible with similar levels of noise meaning T gates are not abnormally expensive, but two-qubit

gates produce substantially higher levels of error, meaning algorithms must be designed to minimise the number of two-qubit gates.

With the basics of quantum logic in place, I will now set out to review parts of both near and future term quantum algorithms relevant to the work in this thesis.

1.5 Variational Algorithms

The Noisy Intermediate-Scale Quantum era was coined by Preskill in 2018 [37] as a means to describe the regime of quantum computing where qubit counts and quality are sufficiently high (50-100s of qubits) that brute force classical simulation is no longer possible, but before error correction is able to remove the serious limitations on what noisy quantum devices are able to achieve. The primary hope for useful algorithms in this era lies in Variational Quantum Algorithms (VQAs). These rely on a hybrid quantum-classical optimisation loop, where a classical coprocessor aims to reduce the requirements on a quantum device, only using it to do classically hard tasks such as measuring quantum observables.

In 2018, Preskill's view on the potential for these algorithms in the coming years was "nobody knows, but we're going to try it and see how well we can do" [37]. In the following 6 years, many have tried, with 1743 papers produced since 2018 in a search for 'variational quantum algorithm' on webofscience.com. Numerous challenges have been found and investigated without being substantially solved, including: the ubiquity of barren plateaus in the optimisation of variational algorithms [38–40], along with the widespread simulability of special cases where they are not present [41]; extremely high measurement costs due to the $O(1/\epsilon^2)$ measurements at each optimisation step; the effect of noise on NISQ algorithms [42], with new classical simulation techniques for noisy systems pushing the bar ever higher for NISQ devices to outperform classical simulations [43, 44] resulting in some NISQ proposals requiring error rates of 10^{-5} [45], which is well below current and projected hardware error rates.

Even so, the ideas developed in this period may yet be useful as we move into the early

fault-tolerant era and beyond, and at the end of Chapter 2, I discuss some regions of applicability where issues in training variational algorithms may be less pronounced.

Introduction

Variational Quantum Algorithms, the first of which was proposed by Peruzzo *et al.* in 2014 [46] aim to achieve usefulness in the NISQ regime by sidestepping the high-depth requirements of other quantum algorithms by using a low-depth Parametrised Quantum Circuit (PQC), $U(\underline{\theta})$, usually called an ansatz, to evaluate some classically hard cost function. This PQC is supplemented by classical variation of the circuit parameters by optimisation of this cost function. This basic concept leads to a great deal of variety in the VQA space due to a degree of modularity in choice of cost function, ansatz and optimiser elements which I will separately cover in this section.

Variational algorithms are based on the observation that the energy expectation value of any state is bounded below by the lowest eigenvalue of the Hamiltonian

$$\langle\psi(\underline{\theta})|H|\psi(\underline{\theta})\rangle \geq E_0 \tag{1.5}$$

with equality when $|\psi(\underline{\theta})\rangle$ is the ground state of the Hamiltonian, meaning we can input values of $\underline{\theta}$ into our quantum computer, and use a classical multivariate optimiser to minimise the energy expectation value to hope to find the ground state of H, which we have chosen to encode the solution of a problem. It may be the case that H represents a real Hamiltonian and therefore the ground state represents the real ground state of a physical system, or that H has been constructed to encode the solution to a combinatorial problem or a compilation task on the quantum device.

Components of a VQA

Cost Function

The cost function of the VQA encodes the problem that the algorithm aims to solve, and is accessed by making measurements on qubits after having prepared

the ansatz state $|\psi(\underline{\theta})\rangle = U(\underline{\theta})\,|0\rangle$, typically to calculate the expectation value of a problem Hamiltonian, which will henceforth just be referred to as 'the Hamiltonian'

$$\langle H\rangle_{\underline{\theta}} = \langle 0|U^{\dagger}(\underline{\theta})HU(\underline{\theta})|0\rangle \tag{1.6}$$

However, Hamiltonians are Hermitian operators and cannot directly be applied to a quantum state without expensive qubitisation methods [47] which have requirements beyond the scope of NISQ technology. In order to evaluate the expectation value of H in Eq. (1.6), the Hamiltonian needs to be decomposed into a linear combination of operators whose expectation values can be evaluated on the device. This is done by expressing the Hamiltonian as a sum of Pauli strings, $P = \otimes_{j=1}^{n}\sigma_j$ where σ_j is a Pauli operator for the jth qubit chosen from $\{I, X, Y, Z\}$ where I is the identity operator. The Hamiltonian can then be written as:

$$H = \sum_{k=1}^{M} c_k P_k \tag{1.7}$$

Where c_k is a complex coefficient for the kth Pauli string, and the desired Hamiltonian expectation value can be written as

$$\langle H\rangle_{\underline{\theta}} = \sum_{k=1}^{M} c_k \langle P_k\rangle_{\underline{\theta}} \tag{1.8}$$

The P_k are unitary operators so can be easily applied to the state $|\psi(\underline{\theta})\rangle$ to measure the expectation values. There are more advanced methods to speed up this measurement process, such as simultaneously measuring groups of commuting operators [48] and randomised measurement techniques that can allow for the creation of a classical representation of $|\psi(\underline{\theta})\rangle$ which can then be used to calculate expectation values [49].

The choice of cost function in a VQA is crucial as it can impact the convergence to the desired solution. Not only does the cost function need to have a global minimum that correctly encodes the solution of the problem, and ideally to be 'operationally meaningful', i.e. that smaller cost function values correspond to a state closer to the solution, but it is also desirable for it to be free of regions of

vanishing gradient, known as barren plateaus, which often plague VQAs. Both of these issues in optimisation can be impacted by the choice of cost function used [38]. Particularly, global cost functions, where the P_k in the Hamiltonian have support over many qubits, suffer from barren plateaus at much shallower circuit depths than cost functions which only contain local (i.e. with a limited support over qubits) terms [38].

Different complementary cost functions may also be combined to assist in achieving in a single goal. For example, when it is known that the solution to a problem contains a particular symmetry, an element preferring that symmetry can be added to the cost function [50, 51]. For example, when using the VQE to find the ground state of a molecule, often the expected symmetries or spin state of the ground state will be known, and a term favouring this symmetry can be included in the cost function.

Expectation values of operators are not the only kind of cost function useful for VQAs another example is the *fidelity* between the output state of the PQC and a target state $|\Psi\rangle$

$$F(|\Psi\rangle, |\psi(\underline{\theta})\rangle) = |\langle\Psi|\psi(\underline{\theta})\rangle|^2 \tag{1.9}$$

Of course, in order to do this, one needs to have access to a circuit that can create the desired state $|\Psi\rangle$, this is the case if the goal is the variational *compilation* of a circuit that can create this state into PQC form (discussed in Section 1.5), which may be desirable to create a circuit with reduced length or noise-sensitivity. This fidelity can be measured using a SWAP test [52] or Loschmidt echo test, where the ansatz $U(\underline{\theta})$ and the inverse of the unitary V that creates $|\Psi\rangle$ are applied in sequence, and when success is achieved, the qubits will be in the all 0 computational basis state $|0\rangle^{\otimes n}$.

Ansatz

The most critical element of a VQA is the ansatz itself. This ansatz must be capable of properly representing the solution state, the quality of an ansatz to produce a large number of output states is its *expressibility* [53] and the ansatz, along with

the cost function itself, must produce a cost function landscape that is amenable to optimisation, i.e. it must have a favourable *trainability*. Unfortunately, these qualities are not entirely compatible, ansätze which are chosen to be as expressive as possible, such as Hardware Efficient choices, are more likely to suffer from barren plateaus. To be more precise, circuits which can approximate a unitary 2-design (where sampling over the circuit with randomised parameters is approximately equivalent to sampling over the space of all unitaries for the calculating expectation values and variances of observables), are guaranteed to suffer from barren plateaus [54, 55]. For this reason, it is desirable to choose ansätze that are problem-specific, with limited expressivity, but that have been chosen carefully such that it is expected there is a good solution within the states reachable by the ansatz. Another option to improve the expressibility-trainability trade-off is to use variable circuits that are modified automatically over the course of the optimisation, that individually have limited expressibility but via varying the ansatz, a good solution can hopefully be reached.

Typically, an ansatz is represented in a series of layers

$$U(\underline{\theta}) = U_L(\theta_L)...U_2(\theta_2)U_1(\theta_1) \tag{1.10}$$

with

$$U_l(\theta_l) = \prod_m e^{-i\theta_{l,m}H_m}W_m \tag{1.11}$$

where each layer consists of a series of gates with Hermitian generators H_m and unparametrised gates W_m, which are often a sequence of entangling gates.

Hardware Efficient Ansätze are designed to be optimised for low circuit depth and maximal expressibility such that $U_l(\theta_l)$ are typically chosen as the gates native to the given hardware platform and the rotation angles θ_l of each gate are treated as parameters to be optimised [56]. Unfortunately, due to the lack of structure in Hardware Efficient Ansätze, they are rarely useful for problem sizes beyond a few

qubits due to barren plateaus.

Unitary Coupled Cluster ansatz (UCC) this was the first ansatz used in Peruzzo's initial Variational Quantum Eigensolver paper [46] and is used for quantum chemistry applications. It is inspired by the observation that it is reasonable to consider small excitations from a known reference state $|\psi_0\rangle$, to capture quantum correlations in molecules. Typically, the reference state is the easy to prepare Hartree-Fock state and the ansatz acts as $e^{T(\underline{\theta})-T(\underline{\theta})^\dagger}|\psi_0\rangle$. Here the cluster operator $T(\underline{\theta}) = T_1(\underline{\theta})+T_2(\underline{\theta})$ [57] is often restricted to single and double excitations, leading to the 'UCCSD' ansatz (SD for single and double), with

$$\begin{aligned} T_1(\underline{\theta}) &= \sum_{\substack{i\in \text{ occ} \\ j\in\text{virt}}} \theta_i^j \hat{a}_j^\dagger \hat{a}_i \\ T_2(\underline{\theta}) &= \sum_{\substack{i_1,i_2\in \text{ occ} \\ j_1,j_2\in\text{virt}}} \theta_{i_1,i_2}^{j_1,j_2} \hat{a}_{j_2}^\dagger \hat{a}_{i_2} \hat{a}_{j_1}^\dagger \hat{a}_{i_1} \end{aligned} \tag{1.12}$$

where the i and j are summed over occupied and virtual orbitals respectively.

Hamiltonian Variational Ansatz (HVA), first proposed in [58], was developed to reduce the number of parameters required and improve convergence. This ansatz has the benefit of being theoretically motivated by consideration of a time-discretised adiabatic evolution that is guaranteed to drag the eigenstate of a trivial Hamiltonian H_0 to the desired problem Hamiltonian H for a sufficiently deep ansatz. The evolution time θ_l of each piecewise constant, trotterised evolution $U_l(\theta_l) = \prod_m e^{-i\theta_l H_m}$ [59] is variationally optimised to find the ground state. The HVA shows high expressibility but is still 'structured enough to allow for efficient optimisation' [60], although it may have a high depth requirement to reach accurate solutions, despite theoretical guarantees about its connection to adiabatic evolution in the large depth regime.

A phenomenon not specific to any particular ansatz is that of *overparametrisation* [61] where an ansatz can reach a regime where it has higher than a critial number of parameters N_c where there is a phase transition in the cost landscape which

removes spurious local minima, however the number of parameters (and therefore depth) required may be prohibitive to achieve overparametrisation before too much noise builds up in the quantum computer.

Optimiser

As the output from the quantum computer is simply in the form of real numbers that make up the cost function(s), in principle, any multivariate optimisation method could be used to find a minimum, and can leverage extensive research into optimisation techniques used in machine learning. However, more specifically, the optimisation procedure needs to satisfy the conditions of being robust to the noisy cost function values and gradient estimators from the quantum computer, as well as being as efficient as possible due to the relatively high cost of function evaluation. The optimisation methods used in VQAs can be broadly separated into gradient-based and gradient-free methods. Although it is likely that many possible optimiation methods have been tested on VQAs, I present here some prominent and promising methods in the literature.

Gradient-based approaches are a common route to function optimisation, where the gradient of a function $f(\underline{\theta})$ with its parameters, which I will denote $\partial_i f(\underline{\theta})$ for the parameter θ_i. The gradient indicates the direction of greatest change of the function, and this local information can be used to make small iterative steps

$$\theta_i^{(t+1)} = \theta_i^{(t)} - \eta \partial_i f(\underline{\theta}) \tag{1.13}$$

where η is the learning rate that is small enough to be consistent with the fact that the gradient is only giving local information, but large enough to ensure the optimiser is moving across the landscape.

One way of evaluating gradients that is always accessible when access to the function is available is the finite difference method, where one computes gradients as $\partial_i f(\underline{\theta}) \approx (f(\underline{\theta} + \epsilon \underline{e}_i) - f(\underline{\theta} - \epsilon \underline{e}_i))/2\epsilon$. However, given that the accuracy of the gradient estimate depends on ϵ being small, the accuracy of the two function evalutations must be high to achieve an accurate gradient, which requires more

shots on the quantum device to achieve this accuracy, leading to an inefficient way to estimate gradients.

An alternative is the *parameter-shift method*, first proposed in [62], and developed in [63] which uses the structure of the gates in a quantum circuit to create an expression for the gradient which looks like a finite-difference method, but is instead an exact analytical expression of the gradient. For a single parametrised gate with a generator with only two eigenvalues $\pm\lambda$ (this includes all single qubit rotation gates), the gradient for this parameter can be written as

$$\partial_i f(\underline{\theta}) = \lambda(f(\underline{\theta}_+) - f(\underline{\theta}_-)) \tag{1.14}$$

where $\underline{\theta}_\pm = \underline{\theta} \pm (\pi/4\lambda)\underline{e}_i$. This rule can be generalised to gates that share multiple parameters and parameters that share multiple gates [64], achieving a speedup in calculating gradients. It can be also be used to calculate higher order derivatives [65].

Quantum Natural Gradient. Additional information about the structure of the spaces that optimisation is being performed in can be used to supplement the process. Optimisation is not only occuring in the Euclidean parameter space, but also in the Hilbert space that the states occupy. This information can be used in the following Quantum Natural Gradient (QNG) update rule [66]

$$\theta_i^{(t+1)} = \theta_i^{(t)} - \eta \mathcal{F}_{ij}^{-1}(\underline{\theta})\partial_j f(\underline{\theta}) \tag{1.15}$$

where $\mathcal{F}(\underline{\theta})$ is the Fubini-Study, or quantum Fisher information tensor that indicates how much variation in parameters causes a change in Hilbert space, with entries

$$\mathcal{F}_{ij} = \mathrm{Re}\left(\langle\partial_i\psi(\underline{\theta})|\partial_j\psi(\underline{\theta})\rangle - \langle\partial_i\psi(\underline{\theta})|\psi(\underline{\theta})\rangle\langle\psi(\underline{\theta})|\partial_j\psi(\underline{\theta})\rangle\right) \tag{1.16}$$

QNG is inspired by the use of natural gradient methods in machine learning optimisation, but the method is actually equivalent to a previously proposed method, Quantum Imaginary Time Evolution (QITE) [67] which was inspired by physics-based consdierations of reaching the ground state of a system through

evolution by an operator $e^{-\tau H}$. QNG shows advantages in terms of convergence speed and increased resiliance to becoming stuck in local minima [67] and can be generalised to noisy quantum circuits or non-unitary processes [68].
Alternative schemes that are not based on gradient descent are possible, and I will discuss one such method in Chapter 2.

Applications

Here I give a brief overview of some of the possible applications of VQAs to various problems, and some special techniques applicable to certain problems. This will be a necessarily incomplete overview of applications for VQAs, as they have been proven to be a universal model for quantum computation [69], meaning a variational algorithm can be devised to attempt to solve any problem soluable by a quantum computer.

Many-body Physics and chemistry

One of the primary uses for future quantum computers will be in simulating chemistry and physics, the original application they were conceived for. Here by 'simulating', I mean calculating both static and dynamical properties of quantum mechanical systems. Calculating chemical reaction mechanisms, which will allow for efficient design of new materials, drugs and catalysts is posed to be one of the most promising use cases of quantum computers [70].
For the simulation of molecules on quantum computers, a representation of the Hamiltonian of that system that can represented by a series of qubits must first be found. The key issue is that the systems of electrons we wish to simulate obey Fermi-Dirac statistics, and the native qubits on our device act as a series of distinguishable spin-1/2 particles, so we must map from a fermionic hamiltonian to a spin system. This can be done by a series of transformations that map the fermionic annihilation and creation operators $\hat{f}_i$,$\hat{f}_i^\dagger$ corresponding to each mode/orbital of the electron system. The most prominent examples of these transformations are the Jordan-Wigner [71] and Bravyi-Kitaev [72] transformations, although there is ongoing research to find new transformations that can improve in operation

cost or locality of the transformed operators [73, 74]. Once this mapping has been achieved, the expectation value $\langle\psi(\underline{\theta})|H|\psi(\underline{\theta})\rangle$ can be evaluated a variational scheme performed to find the ground state of the molecule. From there, techniques to find excited states such as adding terms to the Hamiltonian to project out the ground state [75] can be used.

Spin models are a good target for variational algorithms, as they have simple Hamiltonians, and even simple one-dimensional models can be classically intractable to study and can display interesting physical phenomena such as many-body localisation, relevant to condensed matter systems [76].

Compilation

Another key element of the near-term quantum computer stack is the task of compiling a desired quantum operation V into a circuit that can be executed on hardware. This executable circuit must be written in the native gateset of the device and satisfy the connectivity constraints of the device, as well as being as noise-resilient as possible.

One possible path to effective compilation of circuits is the use of variational methods to approximate the action of V, either on a single state or attempting to reconstruct the whole unitary, with a PQC $U(\underline{\theta})$ [77]. This can either be done using a fixed ansatz, but there are also various varible ansatz schemes [78–80] that aim to find an ansatz that is as short as possible that is able to find a good approximation to V. A series of cost functions can be used, including directly using fidelity measures as in the Hilbert-Schmidt test used in [77], where the register the operations are being applied on is entangled with another, so that the operations act on a superposition of every computational basis state $\sum_{i=1}^{2^n}|i\rangle$ to learn the whole unitary. Alternatively, energy based methods can be used [81], which devise a synthetic Hamiltonian that can be minimised to solve the problem.

Work in generalisation performance of quantum learning [82, 83], shows that a small polynomial number of training states can in fact be used to approximately variationally compile a whole unitary, allowing for the parallelisation of this task

and removing the qubit overhead required for the Hilbert-Schmidt test. I discuss using this scheme along with measures of entanglement in Section 5.3.
Compilation tasks are a good target for VQAs, as even for small circuits acting as subroutines of a larger quantum program, an improvement in circuit depth or error rate could allow for less stringent hardware requirements for quantum advantage.

Machine Learning

The field of quantum machine learning uses optimisation of PQCs as a quantum equivalent of classical neural networks, even calling the circuits used in machine learning tasks Quantum Neural Networks (QNNs), although they are not composed of virtual neurons analagous to the classical perceptron.
Various schemes exist to use variational methods in supervised learning to classify data encoded into quantum states [84, 85]. Usually these models create a variational state as

$$|\psi(x,\underline{\theta})\rangle = U(\underline{\theta})\,|\phi(x)\rangle \tag{1.17}$$

where x is the input classical data that is encoded into the state $|\phi(x)\rangle$ by a circuit that depends on the x. From there, the parameters can be trained by matching these states to the labels y_i provide the classes of the training data X_i. The following infidelity can be minimised

$$C(\underline{\theta}) = \sum_{i=1}^{D}(1 - |\,\langle y_i|\psi(x,\underline{\theta})\rangle\,|^2) \tag{1.18}$$

The quantum feature map that creates the states $|\psi(x)\rangle$ is a key step in the process, as the PQC $U(\underline{\theta})$ acts linearly on these states, so any needed non-linearities must come from the encoding. We also must expect that in order to achieve any quantum advantage, this encoding step must be classically intractable, and well represent the dataset. A series of data encodings have been proposed [86–88]. However, as with classical machine learning, many of the results rely on heuristics, and it is therefore uncertain as to whether using quantum computers, even with classically hard encodings, can be used to accelerate machine learning tasks in this way. The

source of potential advantage of the large Hilbert space is not so clear as it is for simulation of quantum mechanics. Nevertheless, quantum machine learning is one of the primary areas where it is possible that variational methods will continue to play a key role even after the advent of error corrected quantum computers, as the parameters in a PQC play a very natural role as the parameters of a machine learning model. Whereas in other areas, like simulation, variational algorithms are likely to be used rarely as they are surpassed by more advanced techniques suitable for fault-tolerant machines.

1.6 Shadow Tomography

As a quantum state on n qubits can only be fully specified in general with 2^n complex amplitudes, it seems a futile task to attempt to fully characterise (perform tomography on) states that are the output of some quantum computation. Aaronson in 2018 introduced the alternative paradigm of shadow tomography [89] which only attempts to learn limited amounts of information about particular classes of observables (although the algorithm presented there was not computationally efficient). In 2020, a simple and efficient shadow tomography procedure named classical shadows was introduced [90], including variants for estimating local Pauli operators and operators with low trace norm. The general procedure of estimating observables through classical shadow algorithms is as follows:

- Apply a random unitary U from some ensemble $\mathcal{U}$ to rotate the state. and we measure each qubit to obtain N-bit measurement outcomes $|\hat{b}_i\rangle \in \{0,1\}^N$.

- Generate the classical shadows by applying the inverse of the measurement channel $\mathcal{M}$. The classical snapshots are generated as $\hat{\rho}_i = \mathcal{M}^{-1}(U^\dagger |\hat{b}_i\rangle\langle\hat{b}_i| U)$, and the classical shadows are collections of these snapshots $S(\rho; N) = [\hat{\rho}_1, ..., \hat{\rho}_N]$.

- From these classical shadows we can construct K estimators of ρ from our N_{batch} snapshots as $\hat{\rho}_{(k)} = \frac{1}{r}\sum_{i=(k-1)r+1}^{kr} \hat{\rho}_i$ with $r = \lfloor N_{batch}/K \rfloor$ and

classically calculate estimators of the expectation values using a median-of-means to ensure that all estimators are within the desired precision $\hat{o}_i(N,K) = \text{median}\{\text{tr}\Big(O_i\hat{\rho}_{(1)}\Big), ..., \text{tr}\Big(O_i\hat{\rho}_{(K)}\Big)\}$.

- The sample complexity is then determined by the 'shadow norm' of the operators $\|O\|_{\text{shadow}}$ which depends on the ensemble of unitary transformations used to create the classical shadow.

The simplest shadow tomography algorithm is the classical shadow procedure for local Pauli operators [90], where the unitaries are chosen randomly from single qubit Clifford gates on each qubit and the procedure is equivalent to randomly selecting to measure in the X, Y or Z bases. The sample complexity of obtaining these estimators of M Pauli operators of locality l to error ϵ is $\mathcal{O}[3^l \log(M)/\epsilon^2]$, as the shadow norm depends exponentially on the weight of the Pauli operator. The required measurements of this variant are very near-term friendly, and I will make use of this technique in Chapters 2 and 3.

The unitary ensemble $\mathcal{U}$ can also be made up of *global* Clifford transformations, resulting in a shadow norm proportional to $\text{tr}(O^2)$, but requiring circuit depths that are prohibitive for near-term applications or uses of classical shadows in tomography experiments where full quantum control is not available.

Following on from these initial classical shadow schemes, several ideas have been produced for schemes that interpolate between the two extremes, with Bertoni *et al.* [91] producing a scheme with depth $\log n$ which retains the desirable properties of both extremal schemes whilst, in contrast to the random Clifford scheme, also remain experimentally feasible, with further analysis of the scheme in Ref. [92]. Hu *et al.* provide results for any 'locally scrambled' ensembles [93] and use the example of an ensemble of time-dependent Hamiltonian evolution that generates approximately locally scrambled dynamics to provide an example of hardware-efficient approximate tomography.

Another useful class of observables that were not covered in the initial introduction of classical shadows are fermionic observables $a_i, a_i^\dagger$, which when transformed to

Pauli operators via the standard Jordan-Wigner transformation produce non-local 'Z strings' $\prod_i^{k-1} Z_i$, making their estimation by either local or global Clifford classical shadow techniques inefficient. Instead, alternative methods have been developed for the estimation of fermionic observables with classical shadows [94, 95], making use of an alternative classically simulable ensemble of gates that acts naturally on fermionic operators after the Jordan-Wigner transformation, matchgates [96].

Despite the efficiency of the local Pauli estimation in terms of the number of observables M, the overheads when using observables of moderate locality as well as the $1/\epsilon^2$ scaling necessitating many shots for high precision mean that in practice, classical shadows can require a dauntingly high number of shots to obtain good expectation values for input into further processing. Fortunately, there are a variety of techniques that have been developed for reducing the sample overhead: with one class being methods that only allow for estimation of an even further restricted class of observables, with locally entangled Bell measurements allowing only terms that overlap fully with the dimers to be estimated [97], and 'real' classical shadows only allowing estimation of Pauli strings containing X, Z operators [98].

Another set of techniques involve biasing the randomised measurement process in the case where a specific set of observables are desired to be measured in advance. Derandomisation [99] aims to minimize measurement variance in the estimation of Pauli observables by deterministically selecting specific measurement settings based on the target observables rather than relying on random sampling. By derandomising the process, the method can achieve an equivalent or better estimation accuracy with fewer measurements, often by an order of magnitude.

A related but alternative scheme is local biasing [100], where rather than sampling uniformly, measurement settings are chosen based on their relevance to the observables of interest. If the goal is to measure a specific Pauli observable, measurements aligned with that observable's structure will be more frequently sampled. By biasing measurements, local biasing reduces the variance in estimating the target observable, thus requiring fewer samples for accurate estimates. In practice, local biasing often

means adaptively selecting measurements based on the target observables [101], often resulting in slight performance improvements over derandomisation.

Recent works also allow for the reduction of mean-squared error of the estimators produced from classical shadows via a bias-variance trade-off which would be effective at reducing the error on the high-weight observables. This can be performed either through rescaling conventional shadow estimators [102], or using cumulant expansions for estimating k-RDMs of fermionic systems [103]. I also note that recent works provide a way to reduce the sampling overhead for high-weight observables by optimising the shadow inverse map [104–106]. Error mitigation strategies both *for* the output of classical shadows [102, 107–110] and error mitigation strategies *using* classical shadows [111, 112] have been proposed as ways to improve the near-term prospects of classical shadow techniques.

Algorithmic applications of classical shadows include estimating spectroscopic properties of Hamiltonians through time evolution [113], achieving backpropagation scaling in training variational quantum circuits [114], optimising parametrised quantum circuits (Chapter 2) and performing high dimensional subspace expansions (Chapter 3).

Although this section has largely focussed on classical shadow schemes, other forms of efficient shadow tomography do exist, for example a scheme that can estimate any of the 4^n Pauli oberservables using two copies of a state ρ [115], albeit at the cost of a $O(1/\epsilon^4)$ scaling, limiting the practical application of the method. Schemes based on similar ideas produce results for learning Hamiltonians from their time evolution [116] and learning shallow quantum circuits [117].

1.7 Fault-Tolerant Algorithms

In order to make this thesis as self-contained as possible, and to give a rounded overview of the spectrum of quantum algorithms moving into the future, I will briefly explore some of the relevant primitives of fault-tolerant algorithms. For a more comprehensive overview, see the review by Dalzell *et al*[118].

To begin with, I will discuss two primitives which are the prototypical sources of

quantum speedups, as well as being widely useful in the field. Phase estimation, which is the source of the exponential speedup of Shor's algorithm, as well as a general tool for the projection of a state onto eigenstates of an operator, and amplitude estimation, which is the source of the quadratic speedup of Grover's algorithm, as well as the closely related amplitude amplification which is a widespread tool for producing deterministic results from unitaries that would otherwise require measurements to select the desired outcome.

I will then discuss Block Encodings and Quantum Singular Value Transformation as a unifying picture behind many quantum algorithms, with the parallelisation of these techniques being the motivation for the work in Chapter 4.

Phase Estimation

Quantum Phase Estimation (QPE) is the prototypical source of quantum advantage, being the source of the original exponential speedup in Shor's algorithm [119], as well as for the guided Hamiltonian problem [120] and is a widespread primitive within fault-tolerant algorithms.

The problem that phase estimation aims to solve is to take an eigenstate of a unitary $U|\psi\rangle = e^{2\pi i\phi}|\psi\rangle$ and estimate the value of ϕ. This can be done through the following scheme visualised in Fig. 1.2 [121]

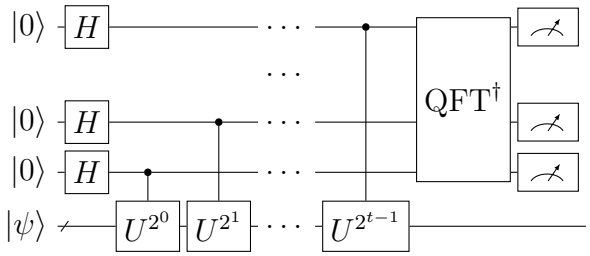


Figure 1.2: Coherent or 'textbook' Quantum Phase Estimation, figure from [121]

Before the application of the $\text{QFT}^\dagger$ block (to be explained momentarily) the state of the qubits can be written (rewriting ϕ into its binary representation $\phi = 0.\phi_1 \dots \phi_t$) as

$$\frac{1}{2^{t/2}}(|0\rangle + e^{2\pi i 0.\phi_t}|1\rangle)(|0\rangle + e^{2\pi i 0.\phi_{t-1}\phi_t}|1\rangle)\dots(|0\rangle + e^{2\pi i 0.\phi_1\dots\phi_{t-1}\phi_t}|1\rangle) = \frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{2\pi i\phi k}|k\rangle \tag{1.19}$$

The final inverse QFT block in the circuit performs the quantum fourier transform, performing the unitary

$$\frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{2\pi i\phi k}|k\rangle \rightarrow |\phi\rangle \tag{1.20}$$

resulting in the bits of ϕ being written out onto the register for measurement.

In the case where in the input state $|\psi\rangle$ is not an eigenvalue of U, the measurements of the phase instead project it onto the eigenvectors $|\phi\rangle$ of U with probability $|\langle\phi|\psi\rangle|^2$. Since the development of the so-called 'textbook' variation of QPE above, it has been shown that the Fourier transform need not be done on the quantum device. These statistical QPE methods extract time-series data from the quantum computer using the circuit in Fig. 1.3 before performing Fourier analysis classically [122]. [1]

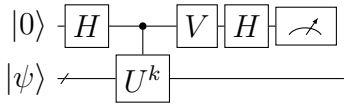


Figure 1.3: Statistical Quantum Phase Estimation using a Hadamard test circuit. Setting $V = 1$ or $S^\dagger$ allows estimation of the real or imaginary parts of $\langle\psi|U^k|\psi\rangle$ respectively.

The runtime of quantum phase estimation scales as $O(1/\epsilon)$ where ϵ is the desired error in the phase estimate.

Shor's algorithm [7] also proceeds via phase estimation, although was not presented in this way in Shor's original paper. There the problem to be solved is to find the least r such that $x^r \equiv 1 \mod n$, this is achieved by creating the superposition over q input states:

$$\frac{1}{q^{1/2}}\sum_{a=0}^{q-1}|a\rangle|x^a \mod n\rangle \tag{1.21}$$

[1] In fact, recent work has shown that it is possible to forgo the requirement for controlled application of U altogether [123], at the expense of more shots.

performing a fourier transformation on the first register to give

$$\frac{1}{q^{1/2}} \sum_{a=0}^{q-1} \sum_{c=0}^{q-1} \exp(2\pi i ac/q) \left|c\right\rangle \left|x^a \mod n\right\rangle \tag{1.22}$$

and measuring c, which, enables finding r through finding the integer d (coprime to r) such that

$$\left|\frac{c}{q} - \frac{d}{r}\right| \leq \frac{1}{2q} \tag{1.23}$$

Shors algorithm can instead be more easily interpreted through the lens of QPE. Indeed the circuit looks quite similar to Fig. 1.2, with a controlled action on a 'main' register before an (inverse) Fourier transform. If we choose for our phase estimation unitary.

$$U_x \left|y\right\rangle = \left|xy \mod n\right\rangle \tag{1.24}$$

This has eigenvalues $\exp(2\pi i d/r)$, and therefore the classical step illustrated by Eq. (1.23) is just reading the output bits of the phase and rounding to the nearest possible value of d/r.

Amplitude Amplification and Estimation

Another key primitive in quantum algorithms is amplitude amplification [124], and the related concept of amplitude estimation, which act as a generalisation of Grover's algorithm [8].

The central problem these algorithms attempt to solve is that of taking a component of a superposition, and either producing that state (amplification), or determining the amplitude of that component (estimation).

Take the unitary:

$$A \left|0\right\rangle = \left|\psi\right\rangle = \sqrt{a} \left|\psi_0\right\rangle + \sqrt{1-a} \left|\psi_1\right\rangle \tag{1.25}$$

We are interested in using this unitary to produce the 'good' state $\left|\psi_0\right\rangle$. To perform amplitude amplification, we can apply an operator composed of two reflections.

$$\begin{aligned} &Q = -S_\psi S_{\psi_0} \\ &\text{where} \\ &S_\psi = I - 2\left|\psi\right\rangle\!\left\langle\psi\right| \\ &S_{\psi_0} = I - 2\left|\psi_0\right\rangle\!\left\langle\psi_0\right| \end{aligned} \tag{1.26}$$

The reflection operator S_ψ can be implemented as we know the unitary that produes $|\psi\rangle$ as $S_\psi = A(I - 2\left|0\right\rangle\!\left\langle 0\right|)A^\dagger$ and S_{ψ_0} is usually constructed from properties known about the good state $|\psi_0\rangle$.

Taking a look at the action of Q on the relevant states:

$$\begin{aligned} Q\left|\psi_0\right\rangle &= (1-2a)\left|\psi_0\right\rangle - 2\sqrt{a(1-a)}\left|\psi_1\right\rangle \\ Q\left|\psi_1\right\rangle &= (1-2a)\left|\psi_1\right\rangle + 2\sqrt{a(1-a)}\left|\psi_0\right\rangle \end{aligned} \tag{1.27}$$

We find that the unitary Q acts as a rotation in the 2-dimensional subspace spanned by $\{\left|\psi_0\right\rangle, \left|\psi\right\rangle\}$, which when writing $\sqrt{a} = \sin\theta$ is

$$Q = \begin{pmatrix} \cos(2\theta) & -\sin(2\theta) \\ \sin(2\theta) & \cos(2\theta) \end{pmatrix} \tag{1.28}$$

If we apply this unitary to $|\psi\rangle$ n times, we arrive at

$$Q^n\left|\psi\right\rangle = \sin((2n+1)\theta)\left|\psi_0\right\rangle + \cos((2n+1)\theta)\left|\psi_1\right\rangle \tag{1.29}$$

giving a probability of measuring the good state $|\psi_0\rangle$ of $\sin^2((2n+1)\theta)$ which is maximised at $n = \lfloor \frac{\pi}{4\theta} \rfloor$, meaning we require $O(1/\sqrt{a})$ repetitions to produce the desired state with high probability.

A slight modification to this scheme is *oblivious amplitude amplification* [125], where the input is the state $|0\rangle|\psi\rangle$ and instead of reflecting about the all 0 state, we reflect about the 0 state on the ancilla.

$$U\left|0\right\rangle\left|\psi\right\rangle = \sin(\theta)\left|0\right\rangle V\left|\psi\right\rangle + \cos(\theta)\left|\Phi^\perp\right\rangle \tag{1.30}$$

We can do amplitude amplification with the oracle

$$Q = -URU^\dagger R$$

where $R = I - 2\Pi$, with $\Pi = |0\rangle\langle 0| \otimes I$ is the reflection about the $|0\rangle$ state on the ancilla.

Modifications to this scheme exist. For example, in cases where we can only lower bound the success amplitude $a \geq a_0$, it is common to use fixed-point amplitude amplification [126].

So far, I have discussed the case where have a process outputting a desired state with a known amplitude that we wish to produce (Quantum Amplitude Amplification, QAA). The alternative situation is where we have this process, but we do not know the value of a, and wish to estimate this value (Quantum Amplitude Estimation QAE), analogous to the classical scenario where we have a biased coin that outputs heads with an unknown probability p, and we wish to estimate this probability.

In order to achieve this, we can note that the operator Q in Eq. (1.28) has eigenvalues $e^{\pm 2i\theta}$, and perform phase estimation on Q to produce a QAE algorithm with complexity $O(1/\epsilon)$, producing a quadratic improvement over the classical algorithm. Although all QAE algorithms operate on this basic principle, a variety of specific algorithms for QAE have been constructed, including unbiased [127] and nondestructive [127–129] variants.

QAE and QAA have widespread applications in quantum algorithms, but require repeated calls to potentially complex unitaries, usually placing them well into the regime of fault-tolerant algorithms. As many quantum algorithms require sampling to estimate probabilities, QAE is a near-universal tool for reducing the error scaling of this estimation to $O(1/\epsilon)$ [2]. Grover's algorithm (a special case of QAA) allows for unstructured search with a quadratic improvement over brute force classical search, and we will see in the following section how QAA is an essential component in a unified picture of quantum algorithms.

[2]However, for estimating small $O(\epsilon)$ probabilities, $O(1/\epsilon)$ scaling can in fact be achieved classically. Recent work has provided a quantum algorithm utilising prior knowledge to improve this to $O(1/\sqrt{\epsilon})$ [130].

Block Encodings and Signal Processing

The basic components of fault-tolerant algorithms I have been discussing can now be brought together into a unified framework for quantum algorithms, that is of embedding matrices within blocks of unitaries and then performing flexible polynomial transformations on them [131].

I will first discuss the topic of Block Encodings, where a block encoding of a matrix A is a unitary that contains the matrix A (which may need to be rescaled by a factor α to reduce the spectral norm to $||A/\alpha|| \leq 1$ to ensure the unitarity of U_A) within a block.

$$U_A = \begin{pmatrix} A/\alpha & \cdot \\ \cdot & \cdot \end{pmatrix} \tag{1.31}$$

In the usual notation, an (α, a, ϵ) block encoding is one that requires rescaling factor α, a ancilla, and the block is distance ϵ from the desired A.

$$||A - \alpha(\langle 0|^{\otimes a} \otimes I)U_A(|0\rangle^{\otimes a} \otimes I)|| \leq \epsilon \tag{1.32}$$

The ability to manipulate block encodings is what makes this paradigm of quantum computation so powerful.

To begin with we will discuss the manipulation (Quantum Signal Processing [132]), of a single qubit reflection operator

$$R(a) = \begin{pmatrix} a & \sqrt{1-a^2} \\ \sqrt{1-a^2} & -a \end{pmatrix} \tag{1.33}$$

By repeated application of this operator and rotations $e^{i\phi_j Z}$ with some set of angles $\{\phi_j\}$ we can implement a polynomial of a.

$$\prod_{j=1}^{d}(e^{i\phi_j Z}R(a)) = \begin{pmatrix} P(a) & \cdot \\ \cdot & \cdot \end{pmatrix} \tag{1.34}$$

$P(a)$ needs to satisfy several conditions

- The degree d of the polynomial is equal to the number of applications of $R(a)$
- Parity(P) = Parity(d)

- $|P(a)| \leq 1 : x \in [-1, 1]$

which are satisfied by Chebyshev polynomials.

The 'qubitisation' of an operator allows us to perform transformations on its eigenvalues as if we are performing quantum signal processing on this single qubit unitary [47].

Consider the action of U_A on an eigenvector λ of A

$$U_A \left|0^a\right\rangle \left|\lambda\right\rangle = \lambda \left|0^a\right\rangle \left|\lambda\right\rangle + \sqrt{1-\lambda^2} \left|\perp\right\rangle \tag{1.35}$$

on the $\{\left|0^a\right\rangle \left|\lambda\right\rangle, \left|\perp\right\rangle\}$ subspace, this has the action of the matrix

$$\begin{pmatrix} \lambda & \sqrt{1-\lambda^2} \\ \sqrt{1-\lambda^2} & -\lambda \end{pmatrix} \tag{1.36}$$

and so allows us to perform exactly the same kind of transformation as Eq. (1.34) to produce polynomials of the matrix by performing additional rotations with angles $\{\phi_j\}$ on the subspace where the ancilla is in $\left|0^a\right\rangle$

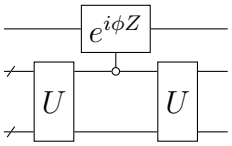


Figure 1.4: QSVT circuit with block encoding U, including a rotation on the $\left|0^a\right\rangle$ ancilla subspace

resulting in a block encoding of $P(A)$.

Now equipped with this block encoding, we can apply the unitary

$$U_{P(A)} \left|0\right\rangle \left|\psi\right\rangle = \sin(\theta) \left|0\right\rangle P(A) \left|\psi\right\rangle + \cos(\theta) \left|\perp\right\rangle \tag{1.37}$$

and assuming $P(A)$ is unitary (therefore the amplitude of the first part is the same for any input state $\left|\psi\right\rangle$) we can use oblivious amplitude amplification to apply $P(A)$ with certainty.

The versatility of this technique is unmatched, allowing nearly every other known quantum algorithm to be expressed in this framework, often leading to improved

versions making use of the additional power afforded by polynomial transformation of operators.

A common technique to produce block encodings is the Linear Combination of Unitaries (LCU) technique, where we wish to apply a block encoding of an operator

$$A = \sum_i \alpha_i U_i \tag{1.38}$$

The improvement of Linear Combination of Unitaries is the motivation for the work in Chapter 4, where I improve the runtime scaling for the common situation where these U_i are Pauli terms.

2

Training Variational Quantum Circuits using Root Finding

Contents

2.1 Foreword

This section is relevant to the work published in Physical Review X

- **G. Boyd**, Bálint Koczor
  Training Variational Quantum Circuits with CoVaR : Covariance Root Finding with Classical Shadows [133]

B.K. produced the original idea of training variational circuits with root finding method on covariance functions, and we jointly extended this idea to reduce the measurement cost by incorporating classical shadows. B.K. provided the theoretical contributions from this work which I will briefly summarise at the beginning of this chapter. I developed and performed all the numerical simulations presented. All writing below is my own, in places using text originally authored by myself verbatim from the manuscript.

2.2 Introduction

The most prominent candidate NISQ algorithm is the variational quantum eigensolver (VQE) whereby a circuit of shallow depth is constructed of parametrised quantum gates such that the emerging quantum state is powerful enough to express the ground state of a problem of interest, e.g., the Hamiltonian of a chemical system. Nearly all such techniques proceed by efficiently estimating the energy (expected value of the Hamiltonian) or an equivalent cost function via sampling with a quantum computer and then the circuit parameters are variationally optimised to find the solution to the desired problem using variants of gradient descent. There are many challenges with these approaches, especially with the parameter optimisations which suffer from the presence of local traps, flat regions as barren plateaus and high sampling costs [61, 134–136].

Here we make significant progress towards addressing some of those difficulties by providing a modification to the variational algorithm paradigm. In contrast to usual minimisation of a single cost function, we use the observation that eigenstates of a Hamiltonian must satisfy certain uncertainty relations with respect to observable measurements. We define these properties as covariances [137–139] between the

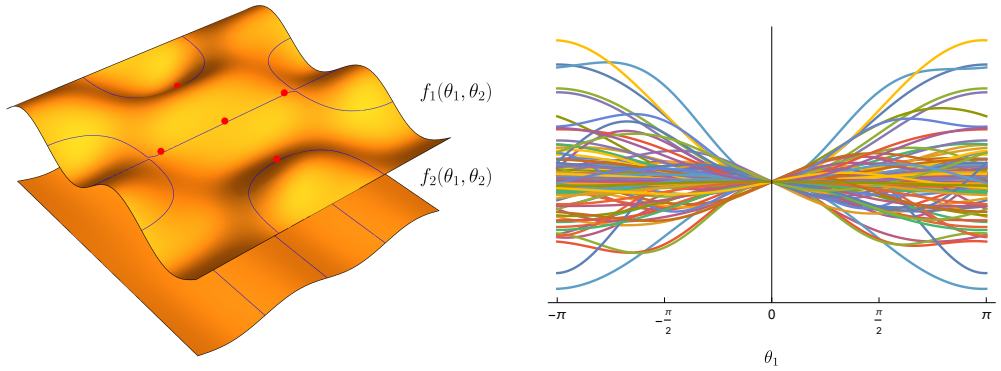


Figure 2.1: (left) A toy-example of a 2-qubit problem whose eigenstates we aim to find by finding parameters of a variational quantum state $|\psi(\theta_1, \theta_2)\rangle$ prepared by two parametrised gates. Covariances $f_k := \langle O_k, \mathcal{H} \rangle_\psi$ between our problem Hamiltonian and between observables O_k span classical surfaces (orange surfaces) and express uncertainty relations between the operators. Blue lines show regions in parameter space where these uncertainties (covariances) vanish, i.e., a root refers to the parameters $\underline{\theta}$ for which the equation is satisfied $f_1(\underline{\theta}) = 0$. Intersections of the lines in the above surface with those of the below surface (red dots) guarantee eigenstates of the problem Hamiltonian as joint roots $f_1(\underline{\theta}) = f_2(\underline{\theta}) = 0$. (right) We use the extremely powerful classical shadow techniques to determine a very large number of these covariances whose slices along the parameter θ_1 are shown in a practically relevant variational circuit (solid lines). We initialise at $\theta_1 \neq 0$ and iteratively find the joint root at $\theta_1 \to 0$ by linearising the covariances through computing a Jacobian and solving the resulting large, overdetermined linear system of equations.

problem Hamiltonian and elements of an operator pool of our choice. This definition gives flexibility in the choice of operator pools and the ability to pose the problem of finding eigenstates as finding joint roots of all possible covariances. As we illustrate in Fig. 2.1(left) these covariances form surfaces as a function of circuit parameters and roots of the individual covariances then submanifolds (as blue lines in Fig. 2.1(left)). Intersections of these as joint roots (red dots) then correspond to eigenstates of the problem Hamiltonian. In our CoVariance Root finding (CoVaR) approach we randomly select a large number of such covariances as illustrated in Fig. 2.1(right) and apply powerful classical numerical techniques. The most significant advantage of CoVaR is that we can reconstruct these covariances with an extreme efficiency using classical shadows: we prove that the cost of estimating a very large Jacobian is comparable to a standard gradient estimation and grows only logarithmically with the number of covariances. I show in a comprehensive set

of numerical experiments that the efficacy of root finding is significantly increased by employing such large datasets and our optimisation procedure is highly robust to shot noise and noise due to random sampling of constraints. We also expect (as supported by evidence from numerics) that randomly generating constraints has the advantage of being more robust against getting stuck in local traps.

We cover a number of important practical applications, such as recompilation and finding ground and excited states of local Hamiltonians, where CoVaR is particularly powerful as we demonstrate in numerical simulations.

The structure of this chapter is the following:in the rest of this introduction section I briefly introduce covariances and related basic notations in Section 2.2.1. The main, general results are presented in Section 2.3 where we state conditions for finding eigenstates based on covariances and pose our problem as root finding. In Section 2.4 I introduce the CoVaR approach that uses classical shadows to find eigenstates of local Hamiltonians and relies on finding joint roots of very large systems. In Section 2.5 I numerically demonstrate the power and utility of CoVaR in solving practical problems and compare our technique to subspace expansion techniques in Section 2.6.

2.2.1 Preliminaries: operator covariances and their properties

Given two arbitrary Hermitian operators A, B we can define a covariance functional between them that depends on a pure quantum state $|\psi\rangle$ as:

$$\langle A, B\rangle_\psi := \langle\psi|AB|\psi\rangle - \langle\psi|A|\psi\rangle\langle\psi|B|\psi\rangle \tag{2.1}$$

When considering these operators, it will be useful to have in mind an operator pool $\mathcal{P}$ for example, Pauli strings $O_k \in \{\mathrm{Id}_2, X, Y, Z\}^{\otimes N}$ form a complete orthonormal set with respect to the Hilbert-Schmidt scalar product $\mathrm{tr}\left[P_k^\dagger P_l\right]/2^N = \delta_{kl}$. Here δ_{kl} denotes the Kronecker delta.

With respect to operators in this pool, $O_k, O_l \in \mathcal{P}$ we can define an associated Hermitian covariance matrix $\mathbf{C}(\psi)$ of a pure quantum state $|\psi\rangle$ that has the matrix entries

$$\begin{aligned}[\mathbf{C}^\dagger(\psi)]_{kl} =&[\mathbf{C}(\psi)]_{kl} := \langle O_k, O_l\rangle_\psi \\ =&\langle\psi|O_kO_l|\psi\rangle - \langle\psi|O_k|\psi\rangle\langle\psi|O_l|\psi\rangle.\end{aligned} \tag{2.2}$$

Note that the above covariance matrix $\mathbf{C}(\psi)$ expresses fundamental uncertainty relations between the observables O_k in the operator pool via the matrix inequation $\mathbf{C} \geq 0$ [139]. We remark that our definition above of a covariance matrix has complex entries: While this definition will simplify our following arguments, it is worth noting that in the literature other conventions are also commonly used [137–139]. For example, ref. [139] defines a covariance matrix in terms of the anticommutator as the real part

$$\tfrac{1}{2}\langle\psi|\{O_k, O_l\}|\psi\rangle - \langle\psi|O_k|\psi\rangle\langle\psi|O_l|\psi\rangle = \mathrm{Re}[\mathbf{C}(\psi)], \tag{2.3}$$

while the imaginary part is often referred to as the commutator matrix $\frac{1}{2}\langle\psi|[O_k, O_l]|\psi\rangle = i\mathrm{Im}[\mathbf{C}(\psi)]$. We will find it convenient to compactly describe the covariance matrix with complex entries thereby simultaneously referring to both the anticommutator and commutator matrices.

Given the decompositions $A = \sum_k a_k O_k$ and $B = \sum_k b_k O_k$ of Hermitian operators A and B in terms of the orthonormal operator basis we can obtain the covariance functional from the covariance matrix as

$$\langle A, B\rangle_\psi = \sum_{k,l} a_k b_l [\mathbf{C}(\psi)]_{kl} = \underline{a}^\intercal\, \mathbf{C}(\psi)\, \underline{b},$$

where $\underline{a}, \underline{b} \in \mathbb{R}^{r_p}$ are coefficient vectors of the operators and $\mathbf{C}(\psi)$ is the covariance matrix in this operator basis. We will later find it useful to express the special cases as the vector of covariances, or *covariance functions* $\langle O_k, A\rangle_\psi = \mathbf{C}\,\underline{a}$, as the primary quantities of concern will be covariances with the system Hamiltonian $\langle O_k, \mathcal{H}\rangle_\psi$ The variance of any Hermitian operator A can also be calculated conveniently from the covariance matrix using the decomposition of A into orthogonal operators.

$$\mathrm{Var}[A] = \langle A, A\rangle_\psi = \sum_{k,l} a_k a_l [\mathbf{C}(\psi)]_{kl} = \underline{a}^\intercal \, \mathbf{C} \, \underline{a},$$

We will later find it useful to express the variance in terms of covariance functions $\mathrm{Var}[A] = \sum_k a_k \langle O_k, A\rangle_\psi$.

2.3 General results: Finding Eigenstates by Finding Roots

We introduce a different paradigm for training variational circuits than that set out in Section 1.5; Instead of searching for the minimum of a single classical function $E(\underline{\theta})$, we efficiently estimate a large number of covariances that each depend on the set of parameters $\underline{\theta}$ and thus each corresponds to a unique surface as a function of $\underline{\theta}$ as illustrated in Fig. 2.1.
Given a variational quantum state $|\psi(\underline{\theta})\rangle := U(\underline{\theta})|0\rangle^{\otimes N}$ produced by the action of a variational quantum circuit $U(\underline{\theta})$ we define the covariances

$$f_k(\underline{\theta}) := \langle O_k, \mathcal{H}\rangle_{\psi(\underline{\theta})}, \quad \text{with} \qquad O_k \in \mathcal{P}. \tag{2.4}$$

This section introduces the main theoretical underpinnings of our approach in a general setting. In contrast, in Section 2.4 we will introduce CoVaR which is a specific, practically motivated approach where we restrict operators to local Pauli strings which in return allows us to utilise classical shadows. We note that we will also investigate another theoretically interesting special case of operator pools in Appendix A.

2.3.1 Finding eigenstates of a problem Hamiltonian

Finding ground states is a problem of central importance in quantum computing, though finding eigenstates in general is not without its uses. Excited states are also of particular importance for analysing chemical reactions, [140] which is key for using quantum computers to discover new drugs and catalysts. Applications for finding eigenstates also exist outside quantum simulation, for example, in solving classical optimisation problems using quantum hardware via the Quantum Approximate

Optimisation Algorithm (QAOA), originally introduced to solve problems such as constraint satisfaction and Max-Cut problems [141]. Finding eigenstates is also highly relevant to the continued design and improvement of applications and quantum algorithms via techniques, such as variational recompilation and preparation of logical states in Quantum Error Correction Codes as these can all be cast as eigenstate finding procedures [77, 142].

Many techniques for analysing Hamiltonians rely on their sparsity in some basis, commonly for physical problems they can be written in the Pauli basis fact that the problem Hamiltonian $\sum_{a=1}^{r} h_a \mathcal{H}_a$ decomposes into a polynomially growing number $r \in \text{poly}(N)$ of Pauli operators whose expected values can be estimated efficiently with a quantum computer. In the following we will denote the collection of these Pauli strings as $\mathcal{Q} = \{\mathcal{H}_a\}_{a=1}^{r}$.

Given a fixed quantum state $|\psi\rangle$, simultaneous roots of all covariances

$$\langle \mathcal{H}_a, \mathcal{H} \rangle_\psi = 0, \quad \forall \mathcal{H}_a \tag{2.5}$$

provide a sufficient condition for the eigenvalue equation $\mathcal{H}|\psi\rangle = \langle \mathcal{H} \rangle |\psi\rangle$ to hold. To see this, note that the variance of the operator $\mathcal{H}$ can be expressed as $\text{Var}[\mathcal{H}] = \langle (\mathcal{H} - \langle \mathcal{H} \rangle)\psi | \mathcal{H} - \langle \mathcal{H} \rangle)\psi \rangle$ and therefore the condition $\text{Var}[\mathcal{H}] = 0$ implies the eigenvalue equation $\mathcal{H}|\psi\rangle = \langle \mathcal{H} \rangle |\psi\rangle$. Given our expression $\text{Var}[\mathcal{H}] = \sum_a h_a \langle \mathcal{H}_a, \mathcal{H} \rangle_\psi$ simultaneous roots as $\langle \mathcal{H}_a, \mathcal{H} \rangle_\psi = 0$ for all a guarantee that $\text{Var}[\mathcal{H}] = 0$.

Further necessary conditions can be introduced through covariances of other operators

$$\langle O_k, \mathcal{H} \rangle_\psi = 0, \quad O_k \tag{2.6}$$

with respect to any basis operator in our operator pool O_k, as the covariance $\langle A, \mathcal{H} \rangle_\psi = \langle \psi | A\mathcal{H} | \psi \rangle - \langle \psi | A | \psi \rangle \langle \psi | \mathcal{H} | \psi \rangle$ simplifies when $|\psi\rangle$ is an eigenstate of $\mathcal{H}$. We note that the individual covariance functions may vanish without implying the presence of eigenstates of the problem Hamiltonian, for example $\langle \mathcal{H}_a, \mathcal{H} \rangle_\psi = 0$ can be satisfied for a single index a in the special case when $|\psi\rangle$ is an eigenstate of $\mathcal{H}_a$. We therefore require that all covariance functions in our operator pool simultaneously vanish (red dots in Fig. 2.1) for all indexes k which necessarily implies an eigenstate

of the problem Hamiltonian. The problem of searching for eigenstates of $\mathcal{H}$ then becomes that of finding simultaneous roots of a system of covariances.

The above ensures us that in an eigenstate all the exponentially many covariances vanish, however, it is sufficient to verify only that the polynomial number of covariances including operators in $\mathcal{Q}$ are zero.

While the covariances $\langle \mathcal{H}_a, \mathcal{H} \rangle_\psi$ are a sufficient cost function for finding an eigenstate of $\mathcal{H}$. We will show that adding other operators from the pool O_k that are orthogonal to our problem Hamiltonian $\mathrm{tr}[\mathcal{H}O_k] = 0$ increases the efficacy of our optimisation algorithm.

2.3.2 Finding joint eigenstates of commuting observables

Many problems of practical interest are concerned with finding joint eigenstates of a group of observables that all commute with each other. For example, in the case of finding a logical state of a Quantum Error Correction Code, we wish to produce an eigenstate of the generators of the corresponding stabiliser group, which are mutually commuting [121].

Another example is the case of recompilation of quantum circuits. Here we wish to transform a given gate sequence V into a native gate sequence U with an optimal circuit depth, as well as make it resilient to noise. Both in the case of Full Unitary Matrix Compilation (FUMC) [77] and Fixed Input State Compilation (FISC) [143] the problem can be stated as applying $U^\dagger$ after V onto our reference state $|\underline{0}\rangle$ (see section 2.5.1 for details) which at the solution $V = U$ would correspond to the identity operation $U^\dagger V = \mathrm{Id}$ and the resulting state $|\underline{0}\rangle$ is then the ground state of the Hamiltonian $-\sum_{j=1}^{N} Z_j$. While one ultimately aims to find the ground state of this Hamiltonian, note that we can also accept any computational basis state $|n\rangle$ which are simultaneous eigenstates of the mutually commuting terms $\{Z_j\}$. This allows us to make a useful simplification to the root finding procedure, as we only need to separately find eigenstates of all the $\mathcal{H}_a$.

2.3.3 Conventional techniques for finding roots

There are a large number of well-established techniques for finding simultaneous roots of vector-valued functions and almost all such techniques are in some way related to Newton's original method [144, 145]. Newton's method proceeds by linearising the vector of covariance functions $\mathbf{f}(\underline{\theta})$ via the first-order Taylor expansion as

$$\mathbf{f}(\underline{\theta} + \Delta\underline{\theta}) = \mathbf{f}(\underline{\theta}) + \mathbf{J}\Delta\underline{\theta} + \mathcal{O}(\|\Delta\underline{\theta}\|^2). \tag{2.7}$$

Given each covariance function is an infinitely differentiable, smooth function of the parameters $\underline{\theta}$ one can indeed apply Newton's method and can approximate roots by solving the equation $\mathbf{f}(\underline{\theta}) = \underline{0}$ using the above expansion $\mathbf{f}(\underline{\theta}) = -\mathbf{J}\Delta\underline{\theta}$ and neglecting second-order terms.

This results in an iterative procedure whereby at every iteration we compute the Jacobian $\mathbf{J}$ with a quantum computer and apply its (regularised pseudo)inverse to the vector of covariances $\mathbf{f}$ to compute the parameter-update rule as

$$\underline{\theta}_{t+1} = \underline{\theta}_t - \mathbf{J}^{-1}\mathbf{f} \qquad \text{with} \qquad [\mathbf{J}]_{kn} := \partial_n f_k(\underline{\theta}). \tag{2.8}$$

We derive expressions for computing derivatives required for the Jacobian with a quantum computer using well established techniques from the literature [146–148], e.g., such as parameter shift rules. Furthermore, we also discuss in Appendix B.1 that by stacking real parts of $\mathbf{J}$ and $\mathbf{f}$ on top of the imaginary parts results in real $\tilde{\mathbf{J}}$ and $\tilde{\mathbf{f}}$ – enforcing that the solution of the linear system of equations is a real vector $\Delta\underline{\theta}$. While the Jacobian and the covariances are obtained from a quantum computer, the resulting linear systems of equations are solved with a classical computer. It is important to note that we would obtain an under-determined system of equations if our operator pool were smaller than the number of ansatz parameters as $r_p < \nu$. This is the reason why we require that our operator pool, and thus the dimension of the vector $\mathbf{f}$ is at least as large as the number of circuit parameters. Later we will aim to set up highly over-determined systems of equations.

While powerful, the vanilla Newton method has its limitations and is only guaranteed to converge when starting near a root – given the linear model in Eq. (2.7) is only

accurate for small $\|\Delta\underline{\theta}\|$. Nevertheless, a number of advanced techniques have been developed to increase the radius of convergence and some variants of the Newton method have been proved to be globally convergent under mild continuity conditions of the functions [144, 149, 150].

One such family are the Levenberg-Marquadt (LM) methods which are additionally robust against ill-conditioned Jacobian matrices. The approach can be shown to be equivalent to the Gauss-Newton algorithm for least-squares minimisation with a trust-region method [144]. It attempts steps along what is formally a "regularised Newton direction" via the regularised inverse $\Delta\underline{\theta} = [A + \lambda \mathrm{Id}]^{-1}\tilde{\mathbf{f}}$ with $A = \tilde{\mathbf{J}}^{\intercal}\tilde{\mathbf{J}}$ and accepts the regularisation parameter λ based on some condition, e.g., such that $\|\tilde{\mathbf{f}}\|$ decreases. The regularisation matrix is in practice often chosen to be the diagonal matrix $\mathrm{diag}(\tilde{\mathbf{J}}^{\intercal}\tilde{\mathbf{J}})$.

2.4 Covariance Root Finding (CoVaR) via classical shadows

In the previous sections we have described the general theory as the basis of a quantum optimisation algorithm that finds eigenstates of a given Hamiltonian using covariances between the Hamiltonian and a (potentially large) operator pool. We now detail concrete settings where our approach may achieve significant practical value in exploiting near-term quantum devices given our aim is that the number of constraints in the linear system of equations is tractable but is significantly larger than the number of circuit parameters as $\nu \ll N_c$. For this reason we choose an operator pool of size $r_p \in \mathcal{O}(N^p)$ that consists of all Pauli strings that act non-trivially on only p qubits as $\mathcal{P} = \{P_k \in \{\mathrm{Id}_2, X, Y, Z\}^{\otimes N} : P_k \text{ is } p\text{-local}\}$.

This fits very well with the use of classical shadows for determining a very large number of local Pauli strings, and allows us in a NISQ-friendly way to measure N_c covariances and their derivatives using a measurement count that is *only logarithmic* in N_c (see Section 1.6). Therefore, it is possible to offload processing to the classical computer with only a small increase in the number of measurements (quantum resources) required. This combination is ideal for NISQ-era algorithms where

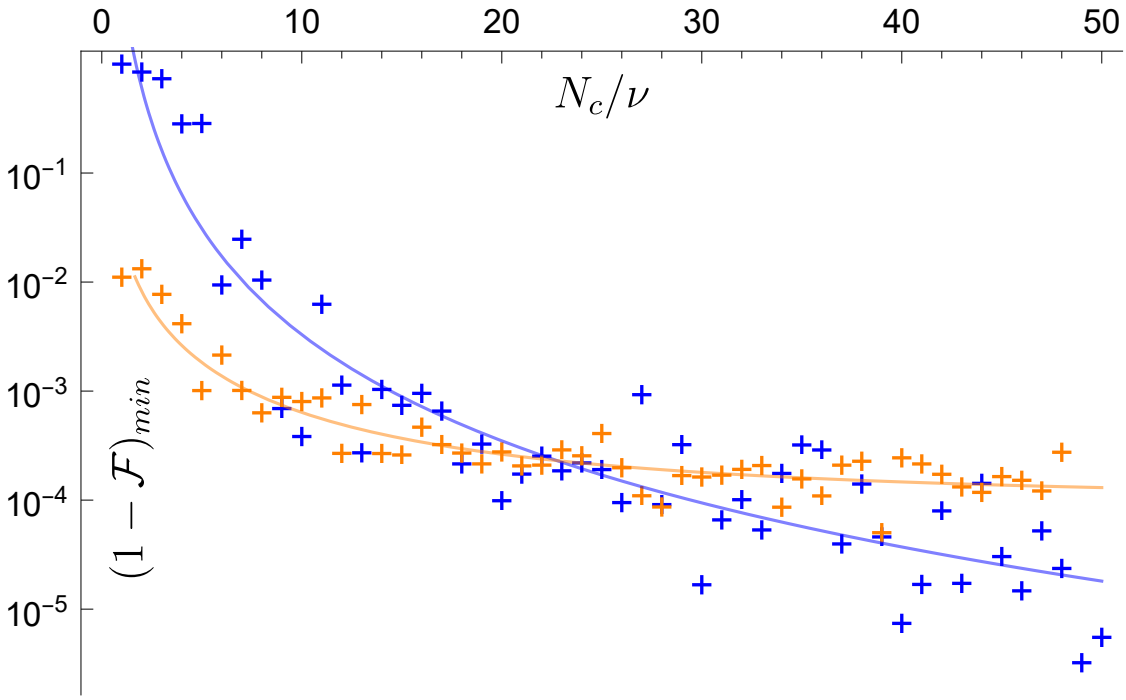


Figure 2.2: Performance improvement when we increase the number of constraints N_c illustrated on a 14 qubit recompilation problem of 'rediscovering' unknown parameters ($\nu = 124$) of an ansatz – minimum infidelity $(1-\mathcal{F})_{min}$ to the ground state achieved over 3 runs of root finding, each with 20 iterations (initial average fidelity $\mathcal{F} = 46 \pm 7\%$). Orange shows results including a shot noise for $N_s = 10^5$ shots and blue without. The operator pool used is all 3-local Pauli strings. Shot noise affects performance differently depending on N_c. First, at small N_c shot noise regularises the ill-conditioned linear system of equations and does seemingly improve performance. Second, while increasing N_c ideally improves performance (blue line), we observe a noise floor (orange line saturates) is approached asymptotically due to shot noise. Fit lines (details in Appendix B.2) are of the form $1 - \mathcal{F}_{min} = a(N_c/\nu)^{-b} + c$ and confirm a polynomially (in N_c) increasing performance, with $b \approx 3.2$ for the noiseless case here, although we expect the degree to depend on the problem.

quantum resources are limited, and it is generally to our benefit if we can offload large, but tractable calculations to a classical computer.

2.4.1 Stochastic optimisation with very large operator pools

Despite very promising experimental progress [19–22, 151], near-term quantum devices are noisy and in order to avoid practically prohibitive accumulation of errors the circuit depth $a(N)$ is required to be shallow and is usually assumed to grow poly-logarithmically as $a(N) \in \text{polylog}(N)$. The Jacobian is generally a non-square matrix with dimension $\mathbf{J} \in \mathbb{C}^{N_c \times \nu}$, where ν is the number of ansatz

parameters typically scaling as $\nu = N\text{polylog}(N)$ due to shallow circuit depth. As such, for a sufficiently large system we can always over-constrain the Jacobian just by including covariances with respect to only two-local Pauli strings given then the number of constraints $N_c = \mathcal{O}(N^2)$ grows faster than the number of circuit parameters. We can thus conveniently define a very large operator pool relative to the number of parameters in the ansatz circuit.

For this reason we set our operator pool $\mathcal{P}$ to contain all p-local Pauli strings and we randomly select constraints $\underline{f}$ of a large size N_c but much smaller than the full operator pool as $\nu \ll N_c \ll r_p$ – but still much larger than the number of circuit parameters. Randomly choosing constraints has the advantage of possibly navigating out of local traps and at every iteration the Jacobian has a tractable but large size. Alternative schemes for choosing constraints are possible, for example selecting constraints based on the magnitude of the covariance functions $|f_k|^2$ such that you attempt to minimise the contributions to the state which can be identified as providing signal that it is not close to an eigenstate (however, this removes some of the advantages of the ability of the stochastic constraint choice, e.g. resistance to local traps, which is why constraints were chosen randomly when using a local Pauli operator pool). In Appendix A, I discuss the particular interpretion of chosing constraints by covariance function magnitude in the case where operators are chosen from a pool that corresponds to an orthogonal basis in *Hilbert space*, rather than *operator space* as with the Pauli basis. Stochastic versions of Levenberg-Marquadt (LM) methods [152, 153] are ideal for accounting for the noise produced by random choice of constraints, as well as the hardware/shot noise on expectation values. Furthermore, bounds on the expected number of iterations to reach a stationary point have been proved under the assumption that the accuracy of covariance measurements is sufficiently large [152]. Note also that stochastic LM and stochastic gradient descent have been extremely popular in the classical machine learning context due to their robustness to noise as well as their robustness against getting stuck in local traps. It is worth noting here that stochastic gradient descent for VQE has been termed for instances when shot

noise on estimated gradients is significant [154] – whereas the present approach is stochastic due to random selection of constraints.

Let us first illustrate how the large, randomly generated dataset is highly advantageous for our purposes. Take for example the simple case when the ansatz circuit has a single parameter and thus the covariance function vector $\mathbf{f}(\theta)$ can be linearised via Eq. (2.7) as

$$\mathbf{f}(\theta + \Delta\theta) = \mathbf{f}(\theta) + \mathbf{J}\Delta\theta + \mathcal{O}(\Delta\theta^2), \tag{2.9}$$

where the Jacobian is $J_k = f_k'(\theta)$ (assuming J_k and f_k are real as we have stacked real and imaginary parts on top of each other). For each individual function $f_k(\theta)$ we can obtain an approximation to the root $\Delta_k\theta = -f_k(\theta)/J_k$, however, we incur an error $\|\mathbf{f}(\theta + \Delta\theta)\|^2 = \sum_{j\neq k}[f_j(\theta) - f_j'\Delta_k\theta]^2 due + ...$ due to the nonlinearity of $f_k(\theta)$. On the other hand, the least squares solution simultaneously takes into account all linearised constraints as $\mathbf{f}(\theta + \Delta\theta) = \underline{0}$ and is given analytically as $\Delta\theta = (\mathbf{J}^\intercal\mathbf{J})^{-1}\mathbf{J}^\intercal\mathbf{f} = -\sum_k J_k f_k(\theta)/[\sum_k J_k^2]$ which inherently minimises the error $\|\mathbf{f}(\theta + \Delta\theta)\|^2$. We illustrate the superiority of the least squares solution to the overconstrained system in Fig. 2.3 as a motivation for using large operator pools. Of course, in order for the error to keep improving as we increase the number of constraints, we need the new constraints to provide useful and non-redudant information for finding an eigenstate, which for general root finding problems $\mathbf{f}(\theta)$ may be very complex. We find evidence in our numerics (Fig. 2.2) that the error does continue to improve as we increase N_c.

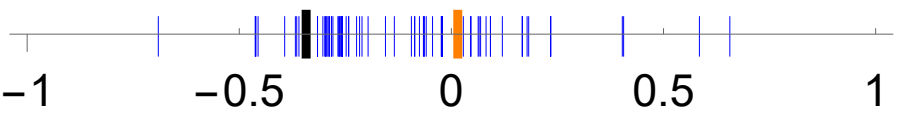


Figure 2.3: The same variational circuit as in Fig. 2.2 with all parameters optimal except for a single variable which was disturbed as $\theta = 0.5$ and thus the exact root is at $\theta \to 0$. Blue lines show the Newton steps computed for 100 individual covariance functions $\theta + \Delta_k\theta$ and their average is represented by the black line. With the orange line we obtain a better approximation of the root by solving a $N_c = 176$ over-determined system of linear equations (see main text).

We obtain a similar conclusion for the error (shot noise) propagation in the general multi-variate case by applying the error propagation formula of ref [136] for matrix inversion. In particular, the error in the update rule scales with the fourth power of the smallest inverse singular value (or regularisation parameter λ^{-4}) of $\mathbf{J}$. Given singular values of our $\nu \times N_c$-dimensional Jacobian matrix grow with N_c, we expect CoVaR is particularly robust against shot noise. Note that for example above in the one-dimensional $\nu = 1$ case the singular value $[\sum_{k=1}^{N_c} J_k^2]^{1/2}$ of $\mathbf{J}^\intercal\mathbf{J}$ indeed grows with the square root of N_c for non-zero derivatives $|J_k| > 0$.

We confirm numerically in Fig. 2.2 on a recompilation problem that the performance of the root finding algorithm increases as the number of constraints N_c/ν in the linear system of equations is increased, in part due to better conditioning on the Jacobian but also because we gain more information about the quantum state. Figure 2.2 also shows the improvement of the performance of root finding as N_c/ν is increased under shot noise and confirms our above analytical arguments: As we increase N_c/ν we better and better approximate the root up until a point when we reach a shot-noise floor $\mathcal{E}$ where the performance is no longer increased. This shot-noise floor is indeed below the precision of determining individual entries $N_s^{-1/2} \propto 10^{-2.5}$.

Computing the least-squares solution to the linear systems of equations is dominated by the step of computing $A := \tilde{\mathbf{J}}^\intercal\tilde{\mathbf{J}}$ which can be performed in time $\mathcal{O}(\nu^2 N_c)$ and, as such, scales linearly with the number of constraints. The rest of the procedure, including the computation of the inverse of the small square matrix $A \in \mathbb{C}^{\nu\times\nu}$ can then be computed in negligible additive time. Given the number of constraints grows at most as $N_c \leq \mathcal{O}(N^p)$ for our specific choice of p-local Pauli strings the computation time t grows at most as $t \leq \mathcal{O}[N^{p+2}\text{polylog}(N)]$ with the number of qubits N.

We estimate that a very large matrix with $N_c = 10^6$ constraints for a variational circuit of $\nu = 10^3$ can straightforwardly be computed in a matter of minutes and fits into the RAM of a typical single node – while distributed computation for larger datasets $> 10^7$ is possible with negligible communication between nodes. We expect determining necessary expected values from classical shadows has a comparable computation time.

2.4.2 Estimating a large number of covariances via classical shadows

Recall that a p-local problem Hamiltonian can be specified in terms of its Pauli decomposition as $\mathcal{H} = \sum_{a=1}^{r} h_k P_a^{(p)}$ where the Pauli strings $P_a^{(q)} \in \mathcal{Q}^{(p)}$ are p-local, i.e., they only act on p qubits non-trivially. Such local Hamiltonians are highly relevant in many important problems which include, for example, recompilation, spin models in materials science, boolean satisfiability problems (3SAT) and fermionic models using mappings that retain operator locality [73, 146, 148, 155–157].

Let us consider an operator pool $\mathcal{P}^q$ that contains all q-local Pauli strings and thus has size $r_p \in \mathcal{O}(N^q)$. As we have discussed we aim to estimate a large number of covariances/constraints f_k that we randomly choose as a subset of the full operator pool such that it has a large but tractable size $N_c \ll r_p$. We can establish that we need only reconstruct expected values of at most $p+q$-local Pauli strings in order to determine all covariances.

Writing the covariances as

$$\begin{aligned}\langle P_k^{(q)}, \mathcal{H}\rangle_\psi = &\sum_{a=1}^{r} h_k \langle P_k^{(q)}, P_a^{(p)}\rangle_\psi = \sum_{a=1}^{r} h_k \langle\psi|P_k^{(q)} P_a^{(p)}|\psi\rangle \\ &- \sum_{a=1}^{r} h_k \langle\psi|P_k^{(q)}|\psi\rangle\langle\psi|P_a^{(p)}|\psi\rangle. \end{aligned} \tag{2.10}$$

the largest weight product we need to estimate is $P_k^{(q)} P_a^{(p)}$, which is at most a $p+q$-local Pauli string with a possible complex prefactor.

Note that determining all covariances that satisfy the sufficient conditions in Eq. (2.5) require that the locality of the operator pool is at least as large as the locality of the problem Hamiltonian via $q \geq p$.

Using classical shadows allows us to reconstruct all $(p+q)$-local Pauli strings with a sample complexity that is merely logarithmic in the system size. This fits particularly well with the preset approach: when the locality $p+q$ of Pauli strings is modest then we can obtain a large, polynomially growing number of constraints $N_c \in \mathcal{O}(N^q)$. Furthermore, given the covariances are fully determined by expected values of (local) Pauli strings, we show that analytical derivatives of

the covariances can be estimated via expected values at shifted circuit parameters via parameter-shift rules [158]. In particular, each partial derivative in the Jacobian $[\mathbf{J}]_{kn} := \partial_n f_k(\underline{\theta})$ is determined by estimating expected values at two different shifted parameters [85] which then allows to reconstruct the full Jacobian of the system just using expected values of local Pauli strings.

We can therefore consider the sample complexity of CoVaR whose (quantum) cost is dominated by estimating the Jacobian and compare it to the cost of determining a gradient vector used in energy minimisation.

Given a p-local problem Hamiltonian $\mathcal{H}$ we use classical shadows to determine a large number N_c of covariances with respect to q-local Pauli strings. The sample complexity of determining the Jacobian of size $\mathbf{J} \in \mathbb{C}^{N_c \times \nu}$ to an error ϵ is

$$N_s = \mathcal{O}[3^{p+q}\, \nu r \log(rN_c)/\epsilon^2].$$

In contrast, determining the gradient of the energy expected value $\langle \mathcal{H} \rangle$ using classical shadows has a complexity $\mathcal{O}(3^p \nu r \log(r)/\epsilon^2)$. As such, determining a very large Jacobian is only logarithmically more expensive than determining an energy gradient (up to a multiplicative constant 3^q that depends on the modest locality of our choice, e.g, $q = 2, 3$).

Theorem 1 of ref. [90] established that M Pauli strings O_i of locality l, can be estimated to precision parameters ϵ, δ via the number of batches $K = 2\log(2M/\delta)$ and the number of samples in the individual batches as $N_{batch} = \frac{34}{\epsilon^2} \max_i \|O_i\|^2_{shadow}$. This results in an overall number of samples $N_{batch}K$ and the norm is given in Lemma 3 in ref. [90] as $\|O_k\|^2_{shadow} = 3^l$.

As we can determine the Jacobian matrix $\mathbf{J} \in \mathbb{C}^{N_c \times \nu}$ by applying the classical shadow procedure $2\nu + 1$ times at different circuit-parameter configurations using parameter-shift rules described in Section 1.5 with $M \leq 3rN_c$. As such, determining these Pauli strings of locality $l = p + q$ requires the number of samples $N_{batch}K \leq \frac{68}{\epsilon^2} 3^{p+q} \log(6rN_c/\delta)$. Given we apply the classical shadow procedure $2\nu + 1$ times we obtain the following upper bound on the sample complexity

$$N_s \leq (2\nu + 1) 3^{p+q} \tfrac{68}{\epsilon^2} \log(6rN_c/\delta)$$

In contrast determining a gradient vector for gradient descent requires 2ν applications of the classical shadow procedure each with $M = r$ and thus we obtain the sample complexity

$$N_s^{(grad)} \leq 2\nu 3^p \frac{68}{\epsilon^2} \log(2r/\delta).$$

In both cases we have determined the necessary Pauli strings to precision ϵ and both the energy gradient and the covariance Jacobian are then obtained from these as a linear combination with respect to Hamiltonian coefficients which leads to a worst-case error propagation of $\mathcal{O}(r)$.

Actually, this bound on the number of required measurements in terms of the locality is noted to be conservative and it is expected that the actual constants are *much smaller* in practice [90]. Furthermore, the inclusion of the development of derandomised classical shadows [159] has the potential to significantly reduce the number of required measurements, i.e., an order of magnitude reduction has been demonstrated in numerical experiments [90]. These techniques could thus greatly improve the speed at which the covariances can be extracted by optimising the Pauli measurement basis to the specific Pauli strings in our operator pool – but we do not consider these in our above performance bounds. Furthermore, the overhead of CoVaR relative to determining a single gradient vector in gradient descent is the constant 3^q (up to the logarithmic dependence on N_c) and is only due to the increase in the locality of Pauli strings with $q = 2, 3$ etc. We can thus expect that determining a *very large* Jacobian has a comparable complexity to determining just a single gradient vector in gradient descent. We will demonstrate in the following that this increased size of the Jacobian has significant advantages in practical applications.

2.5 Applications

2.5.1 Recompilation

The ability to recompile a given quantum circuit into an equivalent but practically feasible or more favourable representation is crucial for the successful exploitation of quantum computers. The ideas exist in many variants, from the application

of classically tractable analytical gate-replacement rules to automatic discovery techniques [160–162]. In variational recompilation, we want to find a parametrised unitary circuit $U(\underline{\theta})$ that approximates a target unitary V either in the entire Hilbert space in case of recompiling a Full Unitary using a Hilbert-Schmidt test [77] or just to approximate the action on a specific input state. After applying the circuits V and $U(\underline{\theta})^\dagger$ consecutively (see figure 2.4 for example circuits corresponding to Full Unitary and fixed input state respectively), the goal is to find circuit parameters $\underline{\theta}$ such that the state of the registers is in the ground state $|\underline{0}\rangle$ of the Hamiltonian $\mathcal{H} = -\sum_{j=1}^{N} Z_j$ [143]. However, the problem would be equally solved by finding ansatz parameters to produce any computational basis state, (i.e. any eigenstate of $\mathcal{H}$) given we can then just append single qubit X gates to the ansatz to correct the 'wrong' bits. This feature, along with the local Hamiltonian allowing for the efficient measurement of many covariances makes it particularly amenable to root finding which is not limited to only searching for the ground state.

For this reason we apply the ideas from Section 2.3.2 to the present problem and define our problem Hamiltonians as $\mathcal{Q} := \{Z_a\}_{a=1}^{N}$. We can enlarge this pool by further considering products of single-qubit Pauli Z operators. Our aim is then to find joint roots of the covariances which then guarantee that the solution corresponds to a joint eigenstate of all operators in $\mathcal{Q}$ as one of the computational basis states. After having found one of these computational states we just apply single qubit X rotations to our ansatz to map to the $|\underline{0}\rangle$ state.

Here we consider an example of *parameter rediscovery* as a benchmark, whereby I recompile a unitary $V = U(\theta^\star)$ that has the same form as the parametrised quantum circuit $U(\theta)$ but with the parameters fixed at some random solution values $\theta^\star$. This has the advantage of being a very hard problem to solve variationally, while also giving us a guarantee that our circuit is capable of expressing the solution – while note that below I will benchmark our CoVaR approach on practical problems as well. Fig. 2.4(c) shows the performance of root finding when we initialise relatively close to the solution (by disturbing parameters $|\Delta\theta_k| \leq 0.3$) on a 10 qubit, 2 layer parameter rediscovery problem and compares it to the performance of gradient

descent. In this recompilation problem our operator pool $\mathcal{P}$ contained all 3-local Pauli strings and we chose randomly N_c operators at every iteration, see details in Appendix B.3. Indeed, Fig. 2.4(c) confirms that root finding is able to converge significantly faster to the shot noise floor, i.e., a limitation due to finite sampling of expected values. Furthermore, Fig. 2.4(c) confirms that root finding has a significantly improved convergence rate (steeper slope), which is improved with a greater number of constraints (blue vs. light blue), while note that the quantum resources required for a single iteration is comparable to that of gradient descent. Fig. 2.4(d) demonstrates the performance of root finding when we start from a randomly chosen initial point in parameter space on the same problem. It fails to make any progress; This is due to the fact that root finding works well when there are only a small number of eigenstates that significantly contribute to the state produced by the PQC. In contrast, random states as nearly equal superpositions of a large number of basis states do not contain a dominant eigenstate towards which root finding could converge thus CoVaR fails to make significant progress. The performance of variance-VQE (a gradient based method that minimises the variance of the Hamiltonian) is also shown for comparison – it is another method which, like root finding, is not only searching for the ground state of the Hamiltonian. Variance-VQE is not stuck in the same way as root finding, but makes slow progress due to its relatively (compared to root finding) slow convergence speed. Also shown is root finding that has been 'initialised' from the random state by a short period of gradient descent that produces an initial state with an appreciable fraction of the lowest energy eigenstates, allowing root finding to efficiently converge. These signify the importance of parameter initialisation for root finding.

2.5.2 Spin Models

Spin models are highly relevant in the study of condensed matter physics and quantum statistical mechanics, as well as being relevant to many other problems that can be mapped to spin models, such as many NP problems [163], cf. approximate optimisation algorithms (QAOA) or spin systems in materials science [146, 148,

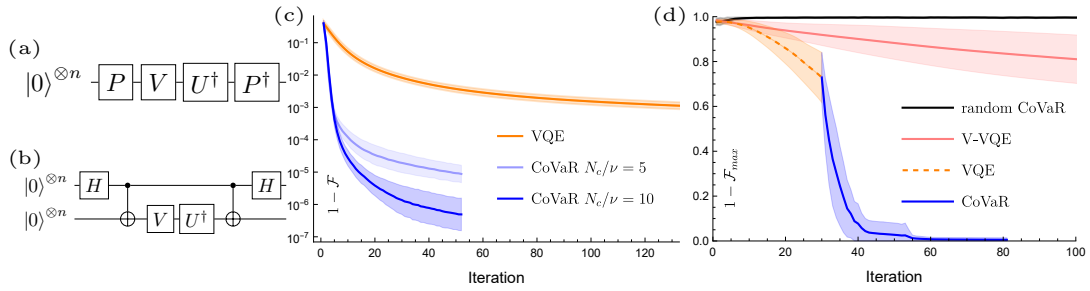


Figure 2.4: Demonstration of CoVaR in fixed input state recompilation of a random set of ansatz parameters for 10 qubits and 2 layers ($\nu = 88$), where the circuit in (a) is used with V and U having the same form but with V having fixed solution parameters and U being optimised. (b) shows the circuit for the recompilation of a full unitary. (c) compares the performance of CoVaR (two blue lines, with $N_c/\nu = 5, 10$ for the upper and lower lines respectively) with gradient descent (orange) when initialised close to the solution, showing the infidelity from the ground state. For initialisation the solution parameters were randomly perturbed, giving an average initial value of $1 - \mathcal{F} = 41 \pm 8\%$. (d) shows performance when initialised randomly in parameter space, showing the infidelity to the nearest computational basis state. Red shows progress for CoVaR and green for variance-VQE when both are initialised randomly, both fail to make significant progress, regardless of choice of N_c. Additionally shown is the use of a short period of gradient descent to 'initialise' (orange, dashed) and then CoVaR (blue, $N_c/\nu = 20$), which reaches a final infidelity of 0.5% on average. All curves show mean and standard deviation of infidelities over 20 runs with shot noise $N_s = 10^5$ other than the blue curve in (d) which averages over the 16/20 runs which were able to converge.

155–157]. Furthermore, lattice models of quantum field theories [164] typically have local Hamiltonians.

Here I perform simulations using root finding to search for low energy excited states of a spin chain described by the Hamiltonian

$$\mathcal{H} = J \sum_{i=1}^{N} \vec{\sigma}_i \cdot \vec{\sigma}_{i+1} + c_i Z_i.$$

With periodic boundary conditions (N+1=1) and where the c_i are randomly selected on-site interactions comparable in strength to the coupling strength J, and $\vec{\sigma}_i$ is the Pauli vector for the ith qubit. This Hamiltonian cannot be simulated classically for large N with reasonable computational resource despite its simple structure [165, 166] and is relevant for studying the phenomenon of many-body localization in condensed matter systems [76].

I use a hardware-efficient ansatz of 20 layers for 10 qubits. Before root finding was performed, the ansatz parameters were initialised to $\underline{\theta}_{imag}$ using Imaginary Time

Evolution [167] to a state with low expected energy, ensuring only a limited number of eigenstates contributed significantly to the state produced by the ansatz. Root finding was then performed from points in parameter space with small random variations from the $\underline{\theta}_{imag}$ to map out the low energy subspace. This process is shown in figure 2.5, where root finding, preceded by Imaginary Time Evolution, is used to map out the lowest energy subspace of a 10 qubit spin chain, and root finding rapidly finds a state an order of magnitude closer to the ground state in energy error than Imaginary Time Evolution is able to converge to. This confirms that root finding has a significantly improved convergence rate when compared to Imaginary Time Evolution, where the latter is equivalent to natural gradient [168, 169] and is thus a second-order method that requires increased absolute quantum resources (samples) [168].

2.5.3 Finding excited states

Searching for excited states of a Hamiltonian is important in many practical applications and indeed variational quantum algorithms have been proposed for solving this class of problems. On the other hand, finding excited states using these orthogonality constrained VQE techniques [143, 170] can be difficult because we need to discover the parameters for and project out every state from the ground state up to the energy of the state we wish to find. CoVaR is agnostic to the energy of the state and acts to find states close to the one it is initialised into. The problem remains that CoVaR could converge to states that have been found in previous runs of the algorithm, and similar techniques of projecting out previously found states can be used. We can also potentially use classical techniques, such as Interval Analysis [160], to find all the roots within an area of parameter space, reducing the problem of converging to an already known eigenstate. In Fig. 2.6 I plot the probability of converging to the ground state of this spin chain with the overlap between the ground state and the initial state used for root finding. These initial states were obtained by performing Imaginary Time Evolution from random points in parameter space down to energies between -5.8 and the ground

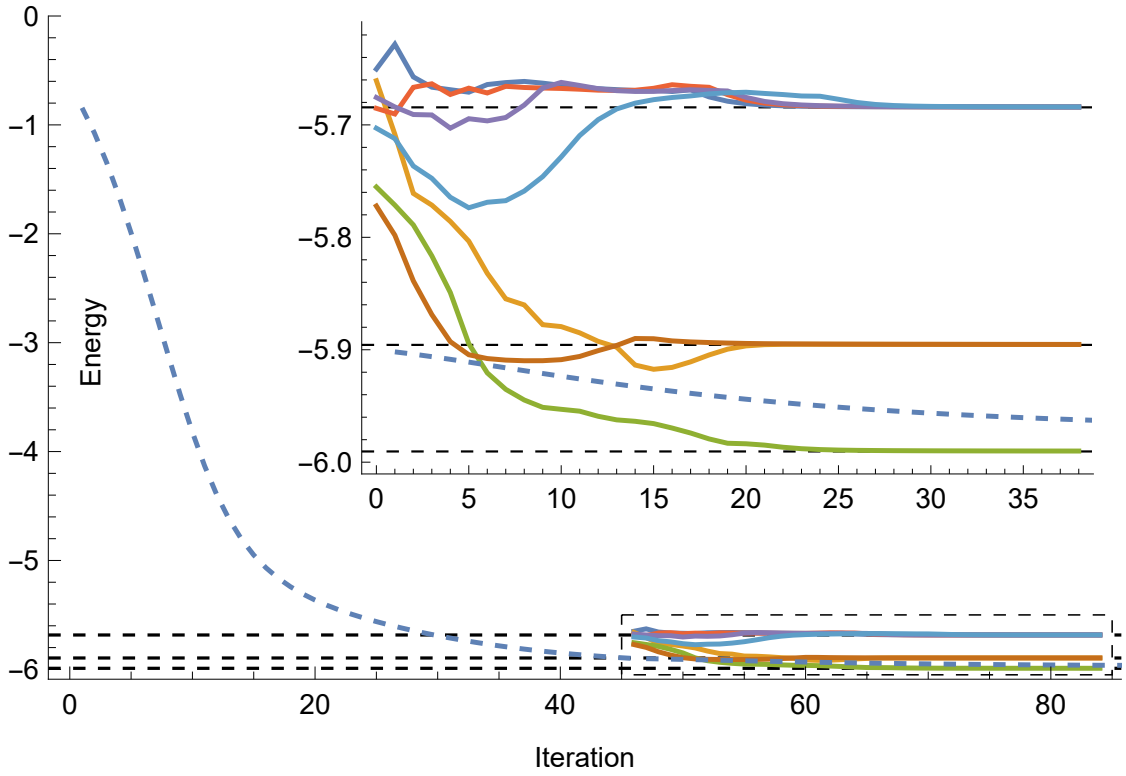


Figure 2.5: Using root finding to explore a low energy subspace of a 10 qubit spin chain via an ansatz circuit of 20 layers ($\nu = 610$). The initialisation to the low energy subspace is performed by Imaginary Time Evolution (Blue, Dashed) and is followed by runs of root finding to find the three lowest eigenstates (marked by dashed lines), to an accuracy of $\Delta E \leq 4 \times 10^{-4}$ in all cases. Imaginary Time Evolution was run for 250 iterations after the point where root finding started and converged to a state with $\Delta E = 0.012$. This investigation was done without shot noise.

state. The initial fidelity where root finding was started was recorded (Fig. 2.6 horizontal axis) and the fraction of runs that converged to the ground state is listed. Indeed, this fraction is proportional to the overlap of the initial state with the ground state and the runs that do not converge to the ground state converge to one of the other low-lying energy levels as per Fig. 2.5.

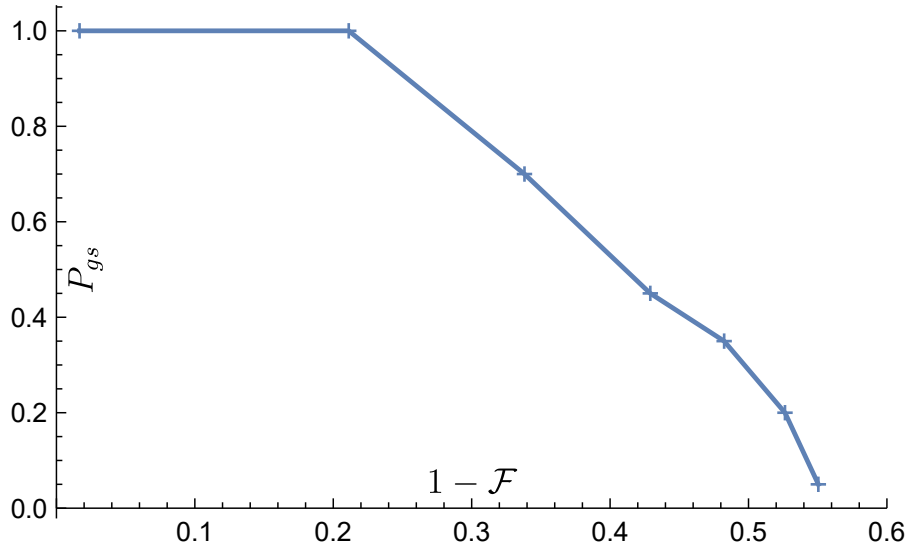


Figure 2.6: Probability of converging to the ground state of the spin chain, y-axis shows the proportion of 20 runs that converge to the ground state (as opposed to one of the excited states) as a function of starting infidelity from the ground state. Initialisation to these starting states is done the same way as in Fig. 2.5, by running Imaginary Time Evolution from a random initial state until reaching selected energies between -5.8 and the ground state. The plot shows a linear decay of success probability with ground state overlap, although we expect the rate of this decay to strongly depend on the distribution of the lowest-energy eigenvalues.

2.6 Comparison to Subspace Expansion

Subspace expansion [171] is a method for discovery of low-energy excited states starting from an estimated ground state $|\tilde{\psi}_G\rangle$. This is done by exploring directions in Hilbert space by applying low-weight operators to add excitations to the ground state using some operator pool, typically that of Pauli operators, to produce a new set of states $|\psi_k\rangle = \mathcal{P}_k |\tilde{\psi}_G\rangle$ and calculating the overlaps $H_{k,j} = \langle\psi_k| \mathcal{H} |\psi_j\rangle$ and $S_{k,j} = \langle\psi_k|\psi_j\rangle$ and diagonalising classically within this subspace. As opposed to CoVaR , subspace expansion cannot prepare the "good quality" representation of the eigenstates with a quantum computer, but is rather limited to estimating their energies with a classical computer.

The connection to CoVaR is expanded further in Appendix A.1 where we express covariance functions in terms of a set of quantum state overlaps $\langle O_k, \mathcal{H}\rangle_\psi = \langle\phi_{O_k}|\phi_{\mathcal{H}}\rangle$ where we can define the (unnormalised) vectors $|\phi_A\rangle := (A - \langle A\rangle)|\psi\rangle$. The correlation matrix in Eq. (2.2) is thus a positive-semidefinite overlap matrix

$[\mathbf{C}(\psi)]_{kl} = \langle \phi_{O_k} | \phi_{O_l} \rangle$. We can make a weak analogy to subspace expansion based on the following observation. Given covariance functions can be expressed as state overlaps, the method explores possible directions in Hilbert space via our operator pool that is beyond the capabilities of the ansatz and we gain information through a linearisation as the Jacobian at what parameters these overlaps are expected to vanish. Similarly, subspace expansion also uses operators additional to the ansatz to explore around the estimated ground state to extract low energy excited states. As such, we may be able to use existing heuristics from subspace expansion for selecting problem-specific operator pools.

Furthermore, we can also use ideas similar to those from this chapter to inform subspace expansion techniques, as the power to measure large numbers of observables allows us to find good expansion operators, I expand on this in Chapter 3.

2.7 Discussion and Conclusion

We have demonstrated that CoVaR can show significantly improved performance by many orders of magnitude in terms of accuracy and improvements in convergence speed compared to analogous variational algorithms due to its effective use of classical shadows. However, a main limitation is its vulnerability to random parameter initialisations, an issue that is similar but more severe than the regular barren plateau phenomenon found in VQAs whereby expected-value landscapes suffer from flat regions due to vanishing gradients [135]. For this reason we expect this limitation will mean that CoVaR will suffer from many of the general challenges of VQAs, e.g. finding problem specific ansätze for VQE problems and good initialisation conditions. As a result of these facts, we expect that the most beneficial use of root finding will be to obtain an accurate description of an eigenstate and its properties when good parameter initialisation is possible. Further work by other authors has already shown variational quantum simulation [172], along with adiabatic evolution [173] as uses of CoVaR where very good initialisations of PQCs can be found due to the continuous nature of time evolution. Recent work [41] has cast doubt on the field of variational algorithms in general, showing that all known examples

of provable absence of barren plateaus also enable classical simulability of the variational procedure after a quantum data collection step. Therefore, the regime in which variational algorithms on quantum computers may remain useful much more closely overlaps with the situation where good initialisations or 'warm starts' are available.

Even in this scenario, local traps may still exist. In fact, optimising VQAs has been shown to be NP-hard due to these persistent local minima [174]. It seems reasonable to expect Levenberg-Marquardt or related techniques [144, 149, 150] may help to mitigate the effects of these local minima near the solution, especially as when gradient descent is stuck in a local trap, those specific parameters may not be a trap for the covariances – especially as we randomly generate operator pools, which we expect to stochastically allow us to avoid traps.
The quantum resources, using classical shadows, required for a single iteration of our procedure is comparable to that of a standard gradient estimation in conventional VQE whereas our approach exhibits a significantly improved convergence speed and robustness against shot noise and possibly against experimental imperfections thanks to this large dataset.
There are a number of apparent extensions to our approach that we have not considered here. First, given the classical shadows are stored in a classical computer we can in principle determine multiple sets of update rules from them and apply the one that most decreases the variance or any other metric as opposed to our fully randomised scheme. Second, it would be worth exploring some specific use cases of CoVaR in more detail, such as finding highly excited states and investigating how effective the method may be at escaping from local traps found during gradient descent. I note that recompilation is another natural set of problems for CoVaR given any eigenstate of the problem Hamiltonian can be accepted as a solution, and it would be desirable to find other problems which can naturally be phrased as eigenstate-finding problems in this way. Thirdly, we have used classical shadows to extract a large amount of covariances in the case where

both the Hamiltonian and operator pool is constructed of local Pauli strings. It is an interesting direction for further work to attempt to use other randomised measurement channels to measure covariances with similar efficiency for non-local Hamiltonians or operator pools. As a matter of fact, related techniques leveraging simultaneous measurements of commuting observables are highly relevant given they allow the efficient reconstruction of a large number of not necessarily local Pauli strings as crucial in applications of quantum chemistry [175–178]. In fact, I exploit these techniques in Chapter 4 not for the purpose of measurement, but for the parallelisation of applying a sum of operator terms. Finally, it is worth noting that our update rule in Eq. (2.8) is invariant under global depolarising noise when the expected value $\langle \mathcal{H} \rangle$ is known to high precision, e.g., via well-established error mitigation techniques [146, 179–181]. As such, it is reasonable to expect that the approach is robust against certain levels of experimental imperfections, but we leave it to future work to evaluate the performance in current and near-future generation hardware and to confirm the practicality of the method.

3
Shadow Subspace Expansion

Contents

3.1 Foreword

This section is relevant to the work published in Physical Review A

- **G. Boyd**, Bálint Koczor, Zhenyu Cai

  High-Dimensional Subspace Expansion Using Classical Shadows [182]

Z.C. produced the original idea and B.K. performed shot noise propagation analysis. I developed the regularisation scheme, techniques for performing Shadow Subspace Expansion on subspaces with a particular symmetry and performed numerical simulations. All writing below is my own, in places using text originally authored by myself verbatim from the manuscript.

3.2 Introduction

Quantum computation often relies on the post-processing of measurement results from a quantum device to produce meaningful results, either for reducing the load on quantum hardware by offloading computation that can be performed classically [46, 119, 183], for mitigating errors in noisy computations [184], or for both [107–110]. Classical shadow techniques [185] can output a rich set of classical data that is amenable towards various post-processing techniques for efficiently probing many properties of the quantum state, offering the ability to "measure first, ask questions later". Its versatility has been employed for a wide range of important tasks in quantum information [113, 114, 186–188]. Ground state energy estimation is one of the most fundamental tasks in quantum simulation [189, 190] and is also more widely applicable in optimisation problems [191]. One typically aims to prepare a good approximation to the ground state through a state-preparation circuit, from which one can directly estimate its energy as the expected value of the problem Hamiltonian. However, this often incurs an estimation error due to constraints in the size of the state preparation circuit and/or noise in the circuit. In this work, we present a post-processing method using existing classical shadow measurement data to produce more accurate ground state energy estimation by performing subspace expansions [192–195] of unprecedented dimension, mitigating the constraints due to both the circuit size and the noise in the shadow data. Given our approach is completely performed in post processing, it does not require any additional quantum resources and circuit runs, and it is guaranteed to output a result that is at least as good as the original value, with the possibility of achieving orders of magnitude

performance improvement, especially in the cases with a highly mismatched starting state and/or a large amount of shadow data.

Our main contributions are a set of techniques for extracting relevant data from classical shadows and then performing numerically stable subspace expansions of high dimension (up to 10000) using noisy data. We investigate the performance of such high dimensional subspace expansions using simulated classical shadows of up to 14 qubits, as well as simulations of the procedure with gate noise, demonstrating the effectiveness of our scheme as an error mitigation method. We also provide an analysis of noise propagation in the method as well as a derivation of the variances of the observables due to the finite number of snapshots from the classical shadow.

In Section 4.3 I provide background on the two main components of the method, subspace expansion and classical shadows, Section 3.4 provides details on our shadow subspace expansion procedure and how I perform the numerical post-processing to ensure a well-conditioned eigenvalue problem. In Section 3.6 I present the results of numerical simulations of our method, investigating the effect of shot noise from classical shadows as well as simulated gate noise.

3.3 Background

Here I first introduce the basic concepts that will form the core of our approach, such as subspace expansion using quantum computers and classical shadows for extracting a large number of expected values.

3.3.1 Subspace Expansion

In many quantum computing applications, such as variational eigensolvers, the goal is to prepare a quantum state that minimizes a specific cost function. When this cost function is the energy of a state with respect to a given problem Hamiltonian H_{tot}, the objective is to prepare its ground state ρ_0. However, due to limitations in qubit resources, gate depth, or the presence of noise, it is often not possible to prepare the exact ground state. Instead, the best achievable approximation is some potentially noisy state, denoted as ρ. Quantum subspace expansion provides

a method to enhance the expectation values of observables obtained from ρ by using additional measurements and post-processing.

In the case when ρ is a pure state $|\psi\rangle\langle\psi|$, we can improve upon the direct energy estimation $\langle\psi|H|\psi\rangle$ by constructing a subspace spanned by the set of states $\{G_i|\psi\rangle \mid 1 \leq i \leq N_G\}$, which are states obtained by applying a set of *expansion basis* operators $\{G_i\}$ to $|\psi\rangle$. We will then classically find the state within this subspace that minimises the energy. Note that I is always an element of $\{G_i\}$ such that our starting state $|\psi\rangle$ is also a state within this subspace, thus the final state we obtain by definition has energy lower than or equal to that of $|\psi\rangle$. Any quantum state in this subspace can simply be written as

$$|\psi_{\vec{w}}\rangle = \frac{\sum_{i=1}^{N_G} w_i G_i |\psi\rangle}{\left\|\sum_{i=1}^{N_G} w_i G_i |\psi\rangle\right\|} = \frac{\Gamma_{\vec{w}}|\psi\rangle}{\|\Gamma_{\vec{w}}|\psi\rangle\|}$$

where $\Gamma_{\vec{w}} = \sum_{i=1}^{N_G} w_i G_i$ is the *expansion operator* which is parametrised by the weight vector $\vec{w}$.

The weight vector (and the corresponding state) of minimal energy within this subspace is given as

$$\vec{w}^* = \arg\min_{\vec{w}} \langle\psi_{\vec{w}}| H_{\text{tot}} |\psi_{\vec{w}}\rangle . \tag{3.1}$$

Applying the same argument to the case where the starting state ρ is a mixed state, the state within the subspace is

$$\rho_{\vec{w}} = \frac{\Gamma_{\vec{w}}\rho\Gamma_{\vec{w}}}{\text{Tr}(\Gamma_{\vec{w}}^2\rho)},$$

and the optimisation problem becomes

$$\vec{w}^* = \arg\min_{\vec{w}} \text{Tr}(H_{\text{tot}}\rho_{\vec{w}}). \tag{3.2}$$

In order to obtain the weights $\vec{w}$ that extremise the energy $\text{Tr}(H_{\text{tot}}\rho_{\vec{w}})$, we need to solve the generalised eigenvalue problem:

$$\boldsymbol{H}\vec{w} = E_{\vec{w}}\boldsymbol{S}\vec{w} \tag{3.3}$$

where

$$\boldsymbol{H}_{ij} = \mathrm{Tr}(G_i H_{\mathrm{tot}} G_j \rho) \tag{3.4}$$

$$\boldsymbol{S}_{ij} = \mathrm{Tr}(G_i G_j \rho). \tag{3.5}$$

If it is the ground state energy that we are interested in, then we obtain an improved ground state energy estimate $E_{\vec{w}^*}$ as the eigenvalue when we solve the equation above. If we are interested in the expectation value of some other observable O with respect to the ground state, then we can measure the following matrix elements

$$\boldsymbol{O}_{ij} = \mathrm{Tr}(G_i O G_j \rho)$$

and then reconstruct the improved observable expectation value using:

$$\begin{aligned}\mathrm{Tr}(O\rho_{\vec{w}^*}) &= \frac{1}{\mathrm{Tr}(\Gamma^2_{\vec{w}^*}\rho)} \sum_{i,j=1}^{N_G} w_i^* \,\mathrm{Tr}(G_i O G_j \rho) w_j^* \\ &= \frac{\vec{w}^{*\dagger} \boldsymbol{O} \vec{w}^*}{\mathrm{Tr}(\Gamma^2_{\vec{w}^*}\rho)}.\end{aligned}$$

3.3.2 Shadow Tomography

The expansion procedure requires us to measure the matrix elements $\boldsymbol{H}_{ij}$, $\boldsymbol{S}_{ij}$, and $\boldsymbol{O}_{ij}$, which involves $\mathcal{O}(NG^2)$ observables. Often, the Hamiltonian cannot be measured directly and must be decomposed into N_H sub-terms, resulting in a total of $\mathcal{O}(N_G^2 N_H)$ observables to measure. As a result, increasing the expansion space dimension N_G to improve subspace expansion performance can be costly. An efficient approach for measuring multiple observables is shadow tomography [89, 185]. To construct a classical shadow, we repeatedly sample a random unitary U_i from a distribution $\mathbb{U}$, apply it to the quantum state, and measure the output in the computational basis. This combined process is described by the *process channel* $\mathcal{M}$. For certain choices of $\mathbb{U}$, we can efficiently invert the process channel classically to obtain a classical snapshot $\rho_l = \tilde{\mathcal{M}}(U_i^\dagger |\boldsymbol{b}\rangle\langle\boldsymbol{b}| U_i)$, where $\tilde{\mathcal{M}}$ is the classical inversion. These snapshots are then used to estimate observables in post-processing.

In this chapter, I again focus on Pauli shadows, where $\mathbb{U}$ consists of all tensor products of single-qubit Clifford operators. Using this scheme, the sample complexity for

estimating M Pauli operators of weight l to within an error ϵ is $\mathcal{O}(3^l \log(M)/\epsilon^2)$ [185]. When both the operators G_i and the Hamiltonian terms are local, this allows us to efficiently compute many expectation values for high-dimensional subspace expansion. Importantly, the $\log(M)$ factor ensures that *all* M estimators achieve the specified precision ϵ. However, in our case, we are not interested in estimating each of the M observables individually but rather in using them for further post-processing to predict a single observable of interest, meaning this factor is not required.

3.4 Methods

We now outline the methodology for Shadow Subspace Expansion (SSE) and discuss practical considerations for performing large subspace expansions in post-processing using classical shadow data. While this approach is readily compatible with more advanced shadow techniques, such as shallow shadows [91–93] or fermionic shadows [94, 95], for simplicity, we will illustrate and explain our results using the local Pauli shadow example described earlier. This method is highly efficient for estimating low-weight Pauli observables simultaneously.

For this reason we construct the expansion operators $\Gamma_{\vec{w}^*}$ as linear combinations of local Pauli strings. The central assumption of subspace expansion is that the expansion basis will map the starting state ρ into another basis state in the subspace that has significant components orthogonal to the starting state, increasing the effective dimension of our subspace. For our special case in which the expansion basis is the set of local Pauli strings, this can be especially effective in many practical scenarios demonstrated later in our numerical experiments.

3.4.1 Shadow Subspace Expansion Procedure

We are considering the setting in which we want to estimate some property O_{tot} of the ground state ρ_0 of some Hamiltonian H_{tot}, with their Pauli decompositions being $O_{\text{tot}} = \sum_{k=1}^{N_O} O_k$ and $H_{\text{tot}} = \sum_{k=1}^{N_H} H_k$, respectively. This can be done by preparing ρ_0 and obtaining a set of its classical shadows of size N_s for the estimations of various properties through post-processing. However, due to noise in the circuit

and/or imperfect ansatz circuit, the state we prepare is some other ρ instead, and it is this state we obtain the shadow data for. We will now outline the way to obtain an improved estimate of $\mathrm{Tr}(O_{\mathrm{tot}}\rho_0)$ using the shadow data of ρ.

1. **Initial expansion basis:** Select an initial expansion basis $\{G_i\}$ of size N_G. One way is to use the set of all Pauli strings up to a certain weight, where the weight is determined by the amount of shadow data (the more shadow data, the higher the weight we can reach).

2. **Filtering expansion operators:** For each expansion basis operator G_i, we will perform a local subspace expansion of dimension 2 using the expansion basis $\{I, G_i\}$ and the corresponding improvement in energy is denoted as ΔE_i. This involves determining the expected values of the observables $\{G_i\}$, $\{H_k\}$ $\{G_iH_k\}$ and $\{G_iH_kG_i\}$ (this final group are just the H_k with an added phase as the G_i can be (anti)commuted through the H_k). These observables have weights up to $M_G + M_H$, and overall the total number of expectation values to be determined from the measurement data is $N_G + N_H + N_GN_H$. To filter out expansion basis operators that are unable to decrease the energy of the initial state, we will rank all operators in $\{G_i\}$ according to their improvement in energy ΔE_i and keep only the top K operators, which gives us the K-*significant* expansion basis. This process has a classical computational linear in N_G of $O(N_GN_H)$ due to the fact that ; thus, N_G can be chosen as potentially very large, e.g., $N_G \gtrsim 10^6$.

3. **Equation for full expansion:** We will perform subspace expansion using the K-significant expansion basis, which involves estimating the set of weight-$2M_G$ observables $\{G_iG_j\}$ to form the matrix $\boldsymbol{S}$, and weight-$2M_G + M_H$ observables $\{G_iH_kG_j\}$ to form the matrix $\boldsymbol{H}$, the matrix entries can be determined based on the observation that any product of local Pauli strings evaluates to a single Pauli string of higher locality. They require the estimations of K^2 and K^2N_H observables, respectively.

4. **Solving the eigenvalue problem:** To obtain the eigenspectrum in the subspace and the optimal weights $\vec{w}$. We detail our method for solving the noisy generalised eigenvalue problem in Section 3.4.2. This step has a classical computational complexity of $O(K^p)$ with a polynomial power $2 \leq p \leq 3$ and is thus the time bottleneck. This reflects the importance of the screening step for choosing the best K observables.

Step 4 will output the energy of the state found via subspace expansion. The following steps are needed for estimating other observables on the output state:

5. Use the shadows to estimate the set of weight-$2M_G+M_O$ observable $\{G_iO_kG_j\}$ of a size K^2N_O.

6. Use the optimal weight $\vec{w}^*$ to reconstruct the optimal observable

$$\begin{aligned}\mathrm{Tr}(O_{\mathrm{tot}}\rho_{\vec{w}^*}) &= \frac{\mathrm{Tr}(O_{\mathrm{tot}}\Gamma_{\vec{w}^*}\rho\Gamma_{\vec{w}^*})}{\mathrm{Tr}(\Gamma_{\vec{w}^*}\rho\Gamma_{\vec{w}^*})} \\ &= \frac{\sum_{i,j}\beta_k w_i^* w_j^* \,\mathrm{Tr}(G_iO_kG_j\rho)}{\mathrm{Tr}(\Gamma_{\vec{w}^*}^2\rho)}\end{aligned}$$

3.4.2 Regularisation of the Generalised Eigenvalue Problem

Classical shadows enable efficient estimation of a large number of expansion operators. However, many states produced by low-weight expansion operators exhibit significant overlap with each other, which can lead to highly singular matrices in the generalised eigenvalue equation. Shot noise propagation can be amplified by these small singular values in the overlap matrix. Thus, careful regularisation is necessary for our approach.

To numerically solve the generalised eigenvalue problem in Eq. (3.3), we require a well-conditioned $\boldsymbol{S}$ matrix, supported by the physical constraint that $\boldsymbol{S}$ be positive semi-definite. I address this by regularizing the problem, transforming the matrices $\boldsymbol{H}$ and $\boldsymbol{S}$ into a subspace where $\boldsymbol{S}$ has sufficiently large positive eigenvalues. This is similar to projecting onto the nearest positive semi-definite matrix in terms of Frobenius norm [196], but with the added step of dynamically setting a threshold to discard small positive eigenvalues, ensuring the eigenvalue problem remains

well-conditioned. Specifically, regularisation is done by restricting the problem to the subspace spanned by the top $\tilde{K}$ eigenvectors of $\boldsymbol{S}$, ordered by their eigenvalues. This represents the effective dimension of the matrix, containing information distinguishable from noise. These eigenvectors form a $K \times \tilde{K}$ transformation matrix $\boldsymbol{Q}$, which reduces the original problem to a dimension-$\tilde{K}$ generalised eigenvalue problem with the following transformed matrices:

$$\tilde{\boldsymbol{S}} = \boldsymbol{Q}^\dagger \boldsymbol{S} \boldsymbol{Q}, \qquad \text{and} \qquad \tilde{\boldsymbol{H}} = \boldsymbol{Q}^\dagger \boldsymbol{H} \boldsymbol{Q}, \tag{3.6}$$

which outputs the energy $E_0^{\tilde{K}}$. The optimal value of $\tilde{K}$ to be used can be determined from a series of such transformations by calculating the ground state energy E_0^l for each and stopping when the energies become unstable. This is determined by tracking the variance of the energy estimates within a moving window the and terminating when the moving variance reaches a minimum, indicating a series of gradually varying energies as the regularisation reaches the optimal value before the eigenvalue problem becomes unstable, resulting in an increase in the moving variance. For example, given a sequence of ground state energies E_0^l, we calculate the differences $\Delta_l = E_0^l - E_0^{l-1}$, and then the moving variance with a window size W:

$$V_l = \mathrm{Var}\{\Delta_l, \ldots \Delta_{l+W}\} \tag{3.7}$$

The minimum of this moving variance provides a good point to stop increasing l where a low energy has been found, but before the eigenvalue problem has become unstable. In our numerics in this work we use a window size $W = 7$.

3.5 Symmetry Considerations

It may be desirable for the subspace expansion to find the lowest energy state with a certain symmetry, that is, within a subspace of the Hilbert space which has the correct eigenvalue of a symmetry operator A. For example, we may want to restrict our solution to the space with the correct number of electrons in a quantum chemistry application, to ensure the ground state energy estimate is not corrupted by non-physical states with the wrong number of electrons. These techniques were

considered in the context of the LiH simulations in Fig. 3.4, but in that case the ground state of the Hamiltonian has the correct physical particle number, so the techniques described in this section were not used in the numerics.

If the state rho ρ has the correct value of this symmetry, it is straightforward to modify the subspace expansion to only choose operators that commute with the symmetry operator $[G_i, A] = 0$.

However, when our state ρ does not itself lie within the symmetry subspace we must instead use post-processing to mitigate for the component with the wrong symmetry. One way of achieving this is to perform an analogue of symmetry verification [197], and use existing techniques and perform an 'internal' subspace expansion to individually mitigate all the matrix elements required [110, 198].

Suppose the ideal state lies in the subspace defined by the projector

$$\Pi_{\mathbb{S}} = \frac{1}{|\mathbb{S}|} \sum_{S \in \mathbb{S}} S \tag{3.8}$$

Given a noisy state ρ that does not lie in this subspace, we can write the subspace projected expectation value of some observable O as [110]

$$\mathrm{Tr}(O\rho_{\mathrm{sym}}) = \frac{\mathrm{Tr}(O\Pi_{\mathbb{S}}\rho\Pi_{\mathbb{S}})}{\mathrm{Tr}(O\Pi_{\mathbb{S}}\rho)} = \frac{1}{|\mathbb{S}|} \frac{\sum_{S,S' \in \mathbb{S}} \mathrm{Tr}(SOS'\rho)}{\sum_{S \in \mathbb{S}} \mathrm{Tr}(S\rho)} \tag{3.9}$$

However, this requires increasing the weight of all observables measured by the weight of the symmetry operator(s) (including the largest weight $G_i H_k G_j$), and may produce a large overhead due to the exponentially scaling shot requirements for higher weight observables.

There is an alternative method which we describe for the case of a single symmetry operator A (although it is easily generalised), that only requires measuring observables $G_i A G_j$. We can additionally determine the elements of the matrix

$$\boldsymbol{A}_{ij} = \mathrm{Tr}(G_i A G_j \rho) \tag{3.10}$$

and before solving for the ground state energy, solve the generalised eigenvalue problem for the symmetry operator:

$$\boldsymbol{A}\vec{w} = A_{\vec{w}}\boldsymbol{S}\vec{w} \tag{3.11}$$

From solving this equation, we can identify a subspace within our subspace expansion with a particular eigenvalue of the symmetry operator A. That is, we can diagonalise the matrix $\boldsymbol{S}^{-1}\boldsymbol{A}$ in order to find a transformation from the full subspace to the subspace with the desired symmetry eigenvalue. This will produce a transformation similar to Eq. (3.6), to be performed to produce symmetry adapted matrices $\boldsymbol{H}_A, \boldsymbol{S}_A$ which will then be used for a better estimate of the lowest energy within the correct symmetry subspace.

3.6 Results

We will first examine the use of Shadow Subspace Expansion (SSE) for solving the ground state energy for a 14-qubit spin chain model. The spin chain Hamiltonian takes the form

$$\mathcal{H} = J\sum_{i=1}^{N} \vec{\sigma}_i \cdot \vec{\sigma}_{i+1} + c_i Z_i. \tag{3.12}$$

with $J = 0.1$ and $c_i \in [-1, 1]$ as uniformly random on-site energies. This Hamiltonian has been considered in studies of self-thermalization and many-body localization and has been identified as a promising target for quantum advantage [199].
For the sake of demonstration, we initialise using the Variational Quantum Eigensolver (VQE) [46] for 300 steps to an initial state with energy error $\Delta E = 0.235$ from the ground state, and then perform SSE.

3.6.1 Performance with increasing subspace size

In this section, we will investigate the performance of SSE as the size of the subspace K increases. For simplicity, we will first assume the shot noise that is added to all the observables that form the entries of $\boldsymbol{H}$ and $\boldsymbol{S}$ is drawn from a complex Gaussian distribution of mean 0 and standard deviation ϵ, and then examine the effects of accurately modelled shot noise from classical shadow data in Sections 3.6.2 and 3.6.3.

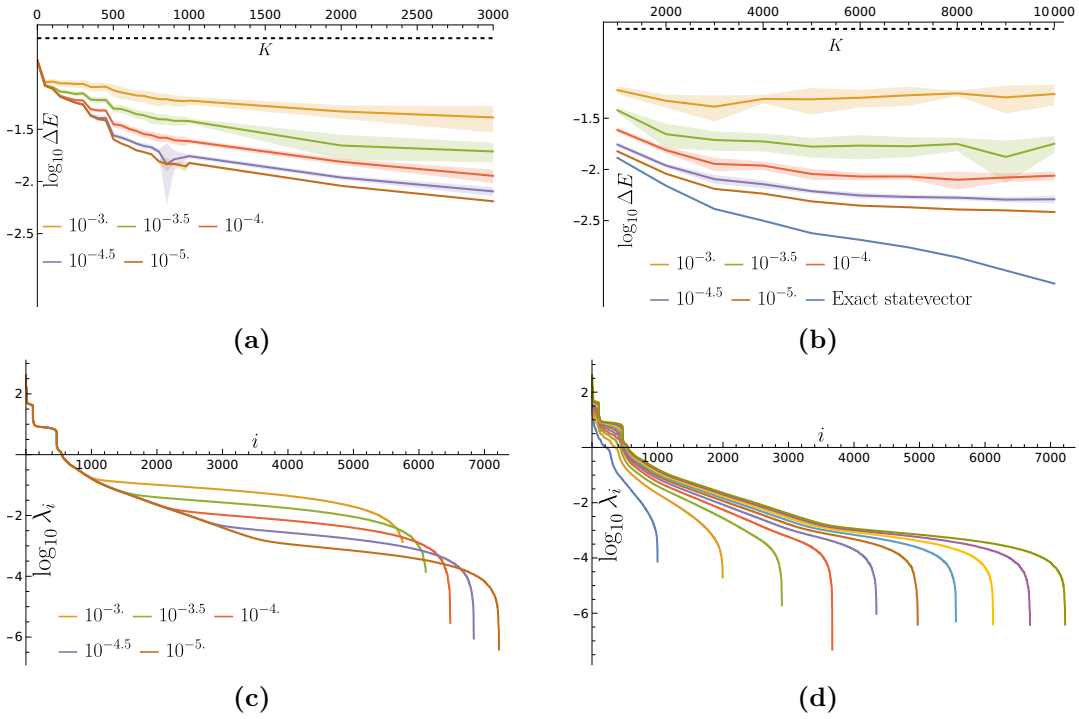


Figure 3.1: (a) Error of SSE for the spin chain problem with subspace dimension K, with shot noise level ϵ per matrix element (denoted by the legend) with the energy of the input state denoted by the black dotted line. Faded regions indicate standard deviation of SSE result as determined by results from 20 independent runs.(b) Errors as K increases demonstrating that increasing K beyond a threshold set by the noise level will not decrease the estimation error. (c) Impact of matrix element error on the eigenvalues of the matrix $\boldsymbol{S}$, for $K = 10000$ on the spin chain problem (with unphysical negative eigenvalues discarded), indicating that the source of the decrease in accuracy in energy is the removal of information from the low magnitude eigenvalues. The truncation dimension required is typically found to be well above the noise-induced levelling off on this plot. (d) Impact of increasing K on the eigenvalues of the matrix $\boldsymbol{S}$ for $\epsilon = 10^{-5}$ for $K = 1000, 2000 \ldots 10000$ increasing from left to right (with negative eigenvalues removed) indicating that the significant eigenvalues are increased as more information in included in the subspace expansion by increasing K, however this does not substantially raise lower magnitude eigenvalues above the noise floor.

In standard subspace expansion, the cost of measuring expectation values increases quadratically with the subspace size, limiting us to small subspaces, typically around $K \sim 10$. However, SSE does not face this limitation. As shown in Fig. 3.1a, we apply SSE with subspace sizes up to $K = 3000$ under varying levels of shot noise. Expanding beyond the $K \sim 10$ range results in a significant reduction in energy estimation errors. With moderate shot noise of $\epsilon \sim 10^{-3}$, we achieve nearly an order of magnitude reduction in error, and with low shot noise of $\epsilon \sim 10^{-5}$, we

see almost two orders of magnitude improvement.

In Fig. 3.1b, we look at even larger subspace sizes up to $K = 10^4$, which nears the maximum K available when using up to 3-local Paulis as our expansion basis operators (10690 for 14 qubits). In the noiseless case, the error in the estimated ground state energy decreases monotonically with the increase of K. With a finite level of shot noise per matrix element, the error in the ground state energy found by SSE hits an *error floor* as we increase the size of the subspace K. This error floor will decrease as we decrease the amount of shot noise, with a rough fit of $\Delta E_{\text{floor}} \propto \epsilon^{0.6}$ found in our numerics.

The phenomenon of the error floor shown in Fig. 3.1b whereby the energy error with a given level of noise saturates as K is increased can be explained by looking at the eigenvalues of $\boldsymbol{S}$. In Fig. 3.1c, we show the eigenvalue spectrum of $\boldsymbol{S}$ for $K = 10^4$ at varying levels of shot noise, demonstrating that the magnitude of some eigenvalues will be so small such that they are buried under the shot noise as the noise increases. The information about the subspace carried by these eigenvalues and their associated eigenvectors needs to be excluded in the subspace expansion to keep the generalised eigenvalue problem well-conditioned. Hence, for a given shot noise level, there is a point where increasing K will only add low-magnitude eigenvalues buried under the noise and will not assist in improving the ground state energy, reaching the error floor mentioned.

We also investigate how increasing K affects the eigenvalues of $\boldsymbol{S}$. In Fig. 3.1d, we plot these eigenvalues for various K at a noise level of $\epsilon = 10^{-5}$. For smaller K, a larger proportion of the eigenvalues rise above the noise level. However, as K increases, more operators that contribute little additional information to the subspace expansion are included, resulting in a growing number of eigenvalues are mainly noise. It is important to note that while the eigenvalues increase roughly linearly with K (see Appendix C), they decay exponentially with the index i—at least in the intermediate range between large eigenvalues and those dominated by noise. Consequently, the linear improvement in signal-to-noise ratio with increasing K raises only a exponentially small number of eigenvalues above the noise threshold,

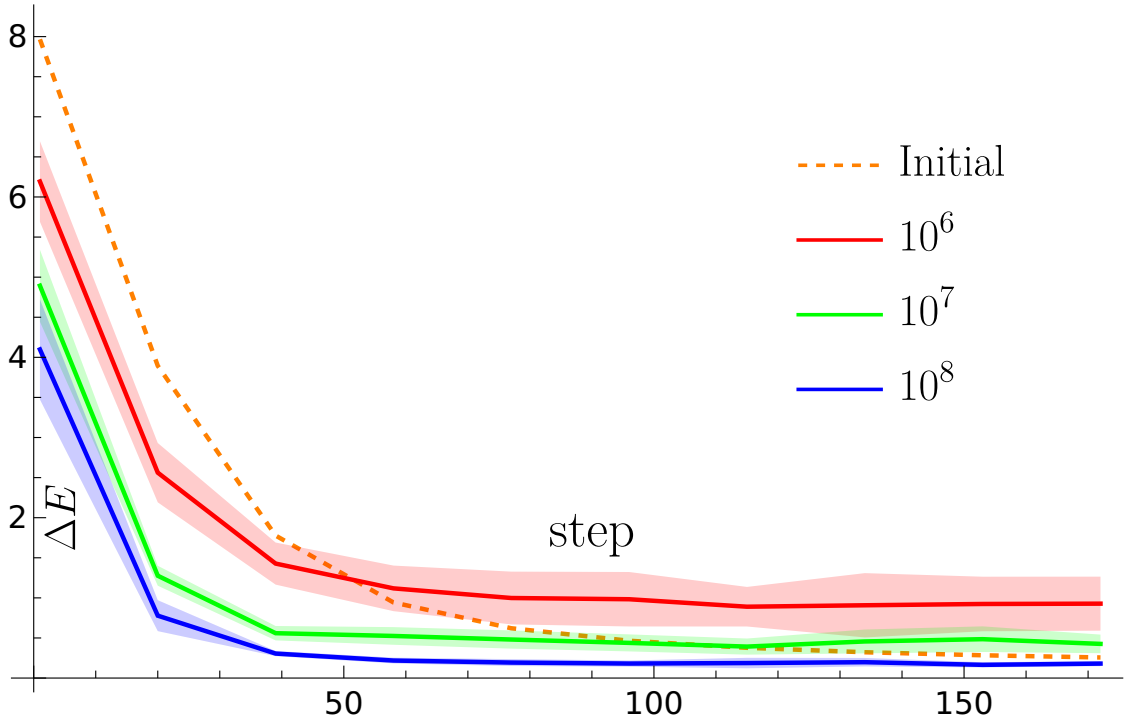


Figure 3.2: Energies of states over the course of a VQE optimisation of the spin chain model using simple gradient descent with line search, obtained using 10^6 snapshots. The orange dashed line corresponds to the energy of the state at that point in the optimisation, whereas the red, green and blue curves correspond to the energies achieved using SSE (with $K = 5000$) using accurate shadow variances for a shadow with the number of snapshots indicated in the legend. Faded regions indicate standard deviation of SSE result as determined by results from 20 independent runs.

resulting in essentially no improvement and the accuracy floor seen in Fig. 3.1b. It is worth noting that the value of K for which this noise floor is reached will also depend on the choice of operator pool used for the expansion. In Fig. 3.1, I use the standard SSE choice of local Pauli operators, but if we instead used a pool including non-local operators, it is likely that there would be more operators with large signal corresponding to the larger eigenvalues of $\boldsymbol{S}$, resulting in a larger value of K where the error floor is reached. Indeed we can see a small example of this phenomenon in Fig. 3.4, where the Hartree-Fock state of LiH cannot be improved by subspace expansion with local Pauli operators.

3.6.2 Simulations Using Shadow Variances

Until this point, we have been considering shot noise producing uniform Gaussian errors on the matrix elements of $\boldsymbol{S}, \boldsymbol{H}$, analogous to shot noise from measuring each of the elements independently. However, in the realistic setting where the expectation values are estimated from a fixed number of classical shadow snapshots, we do not expect the level of noise on each matrix element to be the same. In particular, when using local Clifford shadows, the variance of our estimators will have an extra factor of 3^w, where w is the weight of the observable, meaning that instead of a single value of ϵ shared between all matrix elements as in the numerics in Fig. 3.1, each matrix element corresponding to an observable of weight w has a new error $\epsilon \to \sqrt{3^w}\epsilon$. I therefore perform additional numerics where I accurately simulate the shot noise produced by the finite size of the classical shadow to examine its effect on the subspace expansion. In Fig. 3.2, I perform SSE on a series of states obtained from different steps along the VQE optimisation of the spin chain model, with the numbers of snapshots between 10^6 and 10^8, and compare the results to the energy of the initial state. I find substantial improvements for all numbers of snapshots for high energy initial states, as determined from direct estimation with 10^6 snapshots. For states closer to the ground state, below a certain number of snapshots, the effect of shot noise on SSE means that we can get energy estimates with higher variance than direct estimation. This effect can be mitigated by increasing the number of snapshots. In our experiment, with 10^8 snapshots, we will always obtain a better result compared to direct estimation.

It is worth noting that the original state (corresponding to the identity expansion operator) is in the full K-dimensional subspace. However, when we try to regularise the subspace via transformation and truncation as outlined in Section 3.4.2, the resultant $\tilde{K}$-dimensional subspace may not fully contain the original state anymore. This is the reason why we can reach an energy above direct estimation. This can be potentially solved by creating a new regularisation procedure explicitly keeping the original state in the output subspace, e.g. by regularising only the subspace outside the original state. In the present case, since the direct estimated

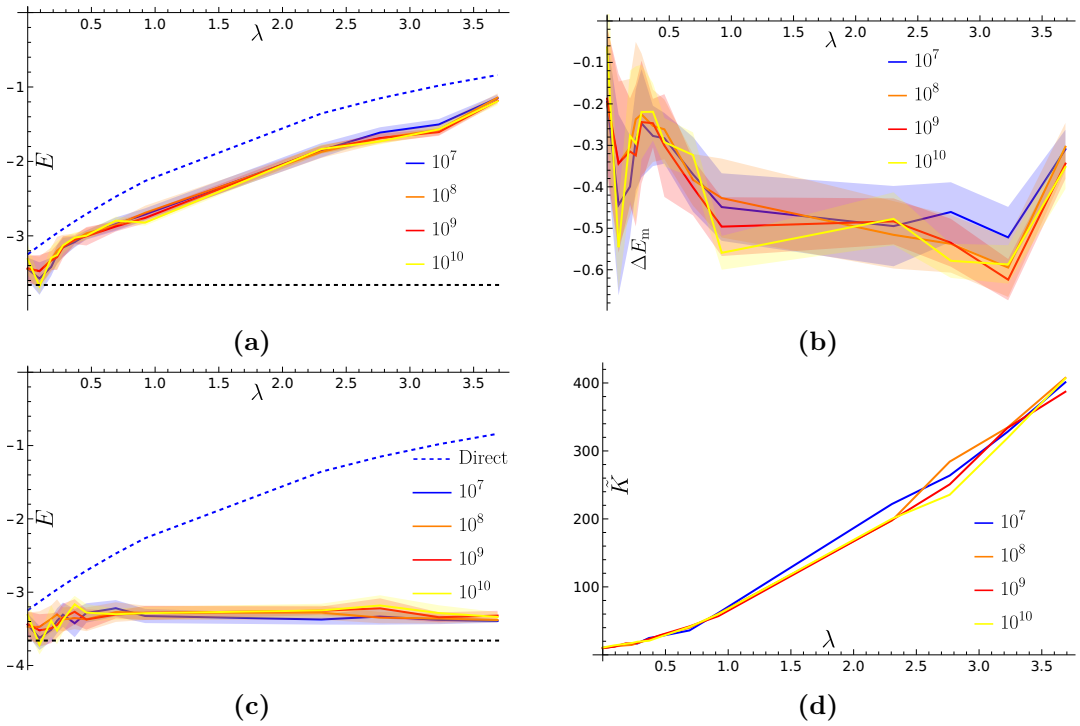


Figure 3.3: (a) Energies obtained via simulations with gate noise, comparing the direct measurement of energy with the use of SSE with varying level of gate noise, denoted by the circuit fault rate λ and shot noise from snapshot counts as given in the legend. SSE lines correspond to an average over 20 separately sampled instances of shot noise with outliers removed, the shaded regions are the standard deviations of these distributions. The effective dimension of the subspace $\tilde{K}$ is allowed to be 15 at maximum. (b) Energy improvements of the SSE mitigated observables over the direct estimation on the noisy state in Fig. 3.3a showing an energy improvement of ~ 0.5 over the whole noise range, approximately independent of shot count. (c) The same calculation, as Fig. 3.3a but with $\tilde{K}$ allowed to vary freely. The average values of $\tilde{K}$ found by the regulariser for the numerics in this plot are shown in Fig. 3.3d. (d) Average value of $\tilde{K}$ found for the circuit used to generate Fig. 3.3c with fault rate λ. The value of $\tilde{K}$ increases past the dimension of the Hilbert space in this small example, but remains well below the 4^n possible Pauli terms that can be used in the expansion for gate noise mitigation.

energy, being an entry in the $\boldsymbol{H}$, is always a known quantity, we will simply output the direct estimated energy if the SSE energy is higher. In this way, we can ensure our method will always provide a better or at least an equally good estimate compared to direct estimation.

Whilst we realise that finding a lower energy than direct estimation using SSE requires either a high energy starting state or a large number of snapshots, we note that there are multiple techniques that could alleviate the shot burden. For example, derandomisation [99] can be used after the identification of good

observables for the subspace expansion. We also note the existence of alternative classical shadow techniques that can achieve greater performance on some restricted classes of observables [97] that could reduce the overhead based on problem-specific knowledge. Recent works also allow for the reduction of mean-squared error of the estimators produced from classical shadows via a bias-variance trade-off which would be effective at reducing the error on the high-weight observables. This can be performed either through rescaling conventional shadow estimators [102], or using cumulant expansions for estimating k-RDMs of fermionic systems [103]. We also note that recent works provide a way to reduce the sampling overhead for high-weight observables by optimising the shadow inverse map [104–106].

3.6.3 Effects of Gate Noise

To investigate the effect of gate noise on our algorithm, we perform density matrix simulations on a 7-qubit spin chain, using a parametrised quantum circuit with 161 single-qubit gates and 60 two-qubit gates, adding single or two-qubit depolarising noise after each gate, with the noise strength for two-qubit gates being 5 times of that for single-qubit gates. We do this for a range of gate noise strengths leading to differing circuit fault rates λ, which are the expected number of faults per circuit execution, given by the sum of error probabilities over all gates $\lambda = \sum_i p_i$, and simulate shot noise as in Section 3.6.2 for a range of shot noise strengths denoted by the number of snapshots taken. We take $K = 500$ and in Fig. 3.3a show the performance of SSE on noisy circuits, producing an energy improvement over the direct noisy estimate of approximately 0.5 over the noise range, with little dependence on the shot count as shown in Fig. 3.3b, demonstrating the robustness of the method as a form of error mitigation, distinct from existing error mitigation methods that combine shadows and subspace expansion through reconstruction and expansion in terms of the density matrix ρ [111, 112].

Noisy simulations of quantum circuits are resource intensive so I only simulated up to 7-qubits, which corresponds to a Hilbert space dimension of $d = 2^7$ with 256 real parameters. The Hermitian and trace-preserving operator space that density

operators live in has the dimension of $d^2 - 1 \sim 15000$. The small system size runs the risk of having an expansion subspace with a dimension larger than the degree of freedom needed to simulate the full quantum system. Hence, we have avoided this by clamping the maximum possible value of $\tilde{K}$ to be 15 in Fig. 3.3a. In Fig. 3.3c we show the same simulations without any restrictions on $\tilde{K}$, and show the corresponding values of $\tilde{K}$ found in Fig. 3.3d, where they increase with the noise level. We see that indeed increasing the dimension of the subspace is an excellent way to mitigate gate noise in the circuit. In fact as shown in Fig. 3.3c, we are able to achieve an estimation accuracy that is largely independent of the circuit noise level if we put no restriction on the dimension of the expansion.

3.6.4 Quantum Chemistry and Excited States

Besides spin chain Hamiltonians, we also examine the performance of SSE on a quantum chemistry Hamiltonian. We use the 12-qubit LiH Hamiltonian in the STO-3G basis [200], and run VQE using a UCCSD ansatz [201]. We then perform SSE on the output VQE states along the optimisation trajectory. In Fig. 3.4, we show the VQE energies achieved by the orange line and the SSE energies achieved from the state at that step number using blue markers, with a shot noise of $\epsilon = 10^{-6}$. SSE is unable to improve on the Hartree-Fock state due to its restriction to local Pauli strings, but even after a small number of steps, two orders of magnitude improvement in the energy estimation error ΔE can be achieved over the directly estimated VQE energy. In this case, the Hamiltonian of LiH used had a ground state with the correct electron number, and so no accommodation needed to be made for the electron number symmetry of the Hamiltonian. However, if this were not the case and the Hamiltonian had a 'non-physical' ground state with a different number of electrons, the techniques in Section 3.5 would be required to ensure the solution was not corrupted with lower energy states of the wrong electron number. As the generalised eigenvalue problem (Eq. (3.3)) outputs an entire spectrum of energies within the subspace, it is interesting to compare the spectrum produced by SSE and the true spectrum of the Hamiltonian. In Fig. 3.5 we show the energy difference

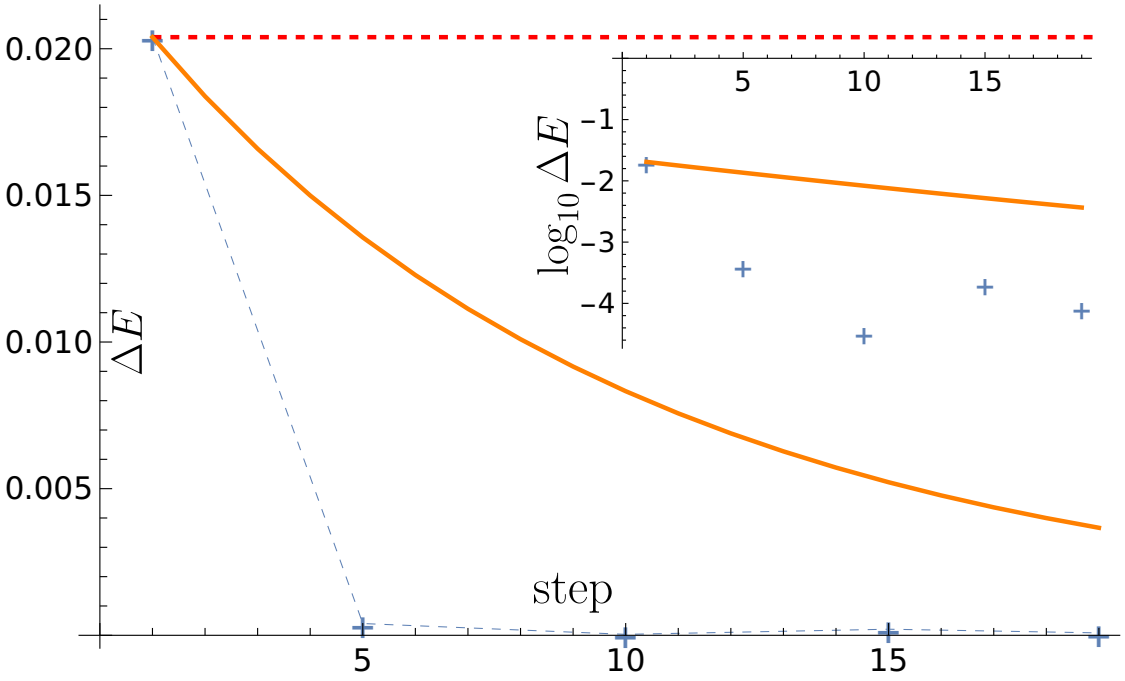


Figure 3.4: Energy estimation errors of direct estimation (orange curve) and SSE (blue markers) as a function of the number of VQE steps (orange, using UCCSD ansatz with a single layer) used to generate the initial expansion state, starting from the Hartree-Fock state for LiH. The Hartree-Fock energy is shown by the red dotted line, and SSE is unable to improve on this due to the long Pauli strings require but even after a small number of VQE steps, SSE is able to rapidly make improvements on the ground state energy. The inset shows the log of the energy error for SSE. Shot noise level $\epsilon = 10^{-6}$.

between the two spectra for LiH for a run of SSE with $K = 2000$ and without shot noise from the final state output by the variational algorithm in Fig. 3.4. The ground state energy difference found is approximately $\Delta E = 10^{-4}$, with the inset showing the two spectra individually (SSE in blue, true spectrum in orange). We can see that for the first ~ 80 eigenstates, the energies are in strong agreement with an energy difference of $\Delta E \lesssim 10^{-2}$. This falls off rapidly as we move to higher energy eigenstates. The agreement at the low energy level suggests the possibility of further extending our techniques to applications involving excited states.

3.7 Conclusion

In this chapter I have presented shadow subspace expansion (SSE) which leverages classical shadow data to perform high-dimensional subspace expansions for esti-

mating ground state energies of quantum systems. I numerically investigated our approach by finding the ground state of a 14-qubit spin chain as well as obtaining excited state energies of a LiH Hamiltonian as relevant in quantum chemistry.
I demonstrate that our method remains effective for subspace expansions of dimension up to the thousands – which is well beyond the size limitations of previous approaches. We observe more than one order of magnitude error reduction (depending on the initial state) when using observable expectation values under a typical level of shot noise $\propto 10^{-3}$, while indeed further error reduction can be achieved by further suppressing shot noise (increasing the number of snapshots or circuit repetitions). We specifically considered simple local Clifford shadows for estimating expansion operators as local Pauli strings and found that our method is robust against relatively small overlaps with the true ground state and, indeed, outperforms direct energy estimation (using the available expected values) in all examples. However, when the initial state is already very close to the true ground state, then we observe that an increased number of shadow snapshots are needed to obtain improvements beyond direct energy estimation, which is a major limitation of the method. Furthermore, we demonstrate that the approach is also robust against gate noise in the state-preparation circuits.
Based on the potential advantages outlined above, when one performs ground-state energy estimation using classical shadows, our approach can be additionally performed fully in post-processing at no quantum cost and can potentially further reduce errors in the energy estimates; Of course, in the worst case our approach yields no improvement in which case one still has access to direct energy estimate. Given SSE is particularly effective when the initial state has relatively small overlap with the true ground state a potentially powerful application could be the following: if we are given classical shadows of the ground state of a given Hamiltonian, we can use this as an initial state for SSE to probe the ground state energy of slightly modified Hamiltonians, potentially allowing us to approximately probe the entire nearby energy landscape from the shadow data of one single state. Furthermore, in our numerical simulations we have primarily focused on estimating ground state

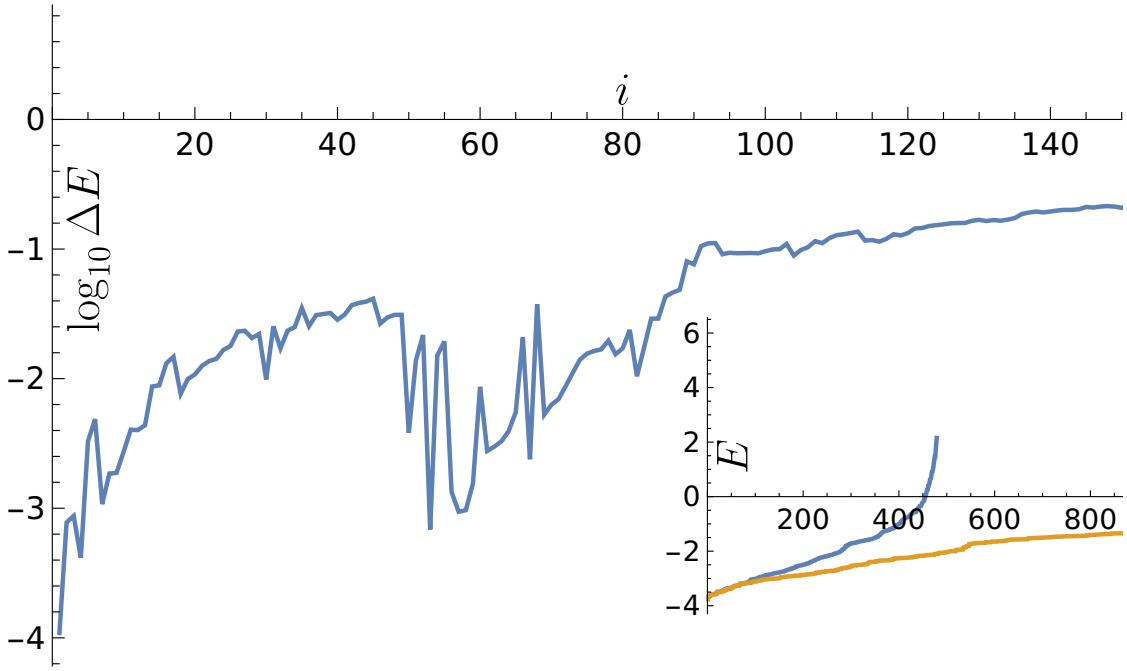


Figure 3.5: The energy difference between the spectrum produced by SSE and the true spectrum of LiH for the first 150 energy levels, indicating some agreement for the lowest-lying eigenstates but rapidly diverges. The inset plot shows the SSE spectrum (blue) and the true spectrum (orange), extending into higher energy levels.

energies, however, as we note in Section 3.4.1, our method can directly be used for estimating other observables from the ground states.

Besides performing expansion of a given state within a subspace, one can also use post-processing techniques to project the state into a given subspace using techniques like symmetry verification [197, 202], in which the symmetry can arise from the target problems or qubit encoding [192]. Ref. [109] have applied classical shadows to symmetry verification in the context of quantum error correction codes and similar ideas are also discussed in Ref. [110]. It will be interesting to see if classical shadows can be applied to more general techniques that incorporate both symmetry verification and subspace expansion [198] as discussed in Section 3.5.

Many further improvements are possible for SSE. The current version of SSE is still very susceptible to shot noise due to the finite number of shadow snapshots. This can be potentially alleviated by studying alternative shadow schemes and expansion bases. I could also explore use of alternative regularisation techniques for subspace expansions developed at around the same time this work was released [203].

4

Parallelisation of Linear Combination of Unitaries

Contents

4.1 Foreword

This section is relevant to the work in review

- **G. Boyd**

  Low-Overhead Parallelisation of LCU via Commuting Operators [204]

The work in this chapter is entirely my own. I developed the algorithm, provided an open source implementation, and performed numerics. As the text of the above work is entirely my own, this chapter will follow that manuscript with few modifications.

4.2 Introduction

The Linear Combination of Unitaries (LCU) method [205] is a powerful quantum subroutine used for the efficient block encoding of operators, that when combined with Quantum Singular Value Transformation (QSVT) [131, 132] leads to asymptotically optimal algorithms for Hamiltonian simulation [206] as well as dissipative ground state finding algorithms [207], and quantum linear algebra [208]. Despite the power of LCU, there are high overheads associated with its implementation, leading to recent work promoting the use of product formula methods for Hamiltonian simulation as a practical alternative [209, 210]. Here, I make a contribution to reducing the time overhead of LCU by presenting a general method for parallelising the algorithm for operators provided in the Pauli representation that is typically able to reduce the depth (T-depth in the fault-tolerant setting) of LCU by a factor of $O(n)$ (see Section 4.3.4) while only requiring an $O(\log n)$ factor increase in qubit count, resulting in a net reduction in the space-time volume cost of the logical algorithm by a factor of $O(n/\log n)$. A special case of the circuits I propose parallelisation methods for in this work is QROM circuits [211], used for loading classical data into quantum computers.

I also provide methods for parallelisation of the application of gates with complicated patterns of positive and negative controls from an ancilla register of size a at the cost of depth $O(\log a)$, which may be preferable when the size of the ancilla register is large (as opposed to the case assumed for LCU above where the register has size $O(\log n)$) which allows us to parallelise the above circuits with a constant factor qubit overhead, reducing the depth by a factor of $O(n/\log a)$.

Special cases of reducing the complexity of LCU exist, e.g. for the specific case of local fermionic Hamiltonians where tensor hypercontraction methods [212] and others [213–215] can be used to reduce the daunting $O(n^4)$ cost of block

encoding molecular Hamiltonians. Parallelisation methods also exist for performing Hamiltonian simulation for certain kinds of Hamiltonian [216].

Prior work also includes methods using one-hot or k-hot encodings of the coefficients or fan-out have been examined to reduce the depth and improve the errors in LCU by taking advantage of the parallelisability of geometrically local observables [217, 218].

In Section 4.3 I discuss the LCU method, as well as the basis for the parallelisation techniques I will use. Section 4.4 will discuss the method in more detail, including discussion of the implementation of these methods on fault-tolerant hardware.

4.3 Background

4.3.1 LCU

The LCU method constructs a linear combination of unitaries using the combination of PREPARE and SELECT subroutines. It involves constructing an operator A as a linear combination of a set of unitary operators U_j with corresponding coefficients c_j:

$$A = \sum_{j}^{L} c_j U_j \tag{4.1}$$

This is done by loading the coefficients into an ancilla register in the PREPARE step:

$$|0^a\rangle \to |\alpha\rangle = \frac{1}{\sqrt{\|c\|_1}} \sum_j \sqrt{c_j}\,|j\rangle \tag{4.2}$$

where the size of the ancilla register is $\log L$. I will restrict my analysis when discussing LCU to $L \sim \text{poly}(n)$ so that the number of ancilla in the coefficient register is $O(\log n)$.

Then the unitaries are applied controlled on the ancilla in the SELECT step:

$$|j\rangle\,|\psi\rangle \to |j\rangle\,U_j\,|\psi\rangle \tag{4.3}$$

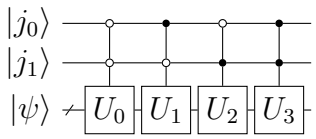


Figure 4.1: Example of the usual procedure for the SELECT step of LCU with two ancillas.

LCU is of particular use for implementing block encodings of non-unitary operators, such as Hamiltonians, and therefore provides access to performing QSVT on these block encodings to apply functions of the operator.

4.3.2 QROM

Quantum Read-Only Memory [211] is a technique for loading classical data into quantum memory, by allowing data to be accessed by an index in superposition:

$$\mathrm{QROM}_d \sum_{l=0}^{L-1} \alpha \, |l\rangle \, |0\rangle = \sum_{l=0}^{L-1} \alpha \, |l\rangle \, |d_l\rangle \tag{4.4}$$

This can be achieved using a circuit in the form of Fig. 4.1 where the unitaries applied are tensor products of Pauli X's encoding the binary strings of the data entries d_l to be loaded. QROM circuits are also used as a component of advanced schemes for block encoding quantum chemistry Hamiltonians [212], where it is noted that QROM is the dominant cost of the procedure, and therefore a prime target for runtime optimisation.

4.3.3 Parallelisation Techniques

Techniques for parallelising quantum algorithms have been known for some time [219]. The algorithm makes use of the primitives of fanout and gate teleportation using states prepared by measurement and feedforward.

The fanout operation [220] (Fig. 4.2) can be thought of as a basis-dependent "copy" operation that takes information in the computational basis from one register and spreads it across additional registers.

$$\sum_j \sqrt{c_j} \, |j\rangle \, |0\rangle \ldots |0\rangle \to \sum_j \sqrt{c_j} \, |j\rangle \, |j\rangle \ldots |j\rangle \tag{4.5}$$

Crucially, this allows us to perform commuting operations on a qubit register in parallel, by first performing a fanout to spread the information across multiple registers, and then performing the operations on each register individually. We then uncompute the fanout operation.

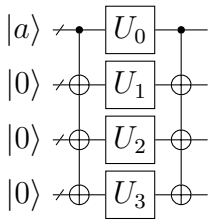


Figure 4.2: Fanout operation performing operations on the same input in parallel.

A log-depth fanout circuit can be constructed straightforwardly from a tree of CNOT gates, and a constant depth circuit can be constructed using teleportation with $O(k)$ ancillas, where the fanout acts on k qubits [221].

An additional tool I use is the use of measurement and feedforward (or adaptive circuits), where circuits contain measurements and operations that are controlled on the measurement results, resulting in deterministic measurement based circuits to prepare certain resource states [31, 221] and the use of gate teleportation with these states [222] to reduce circuit depth, I discuss this further in Section 4.4.1.

4.3.4 Partitioning into commuting groups

A third useful technique is the partitioning of operators into commuting groups of Pauli terms, these methods were initially developed for the optimisation of measurements in variational algorithms [223–226]. In this context, it is desirable to partition observables into commuting groups to determine which components can be measured simultaneously. This is done by calculating an approximate Minimum Clique Cover (MCC) of the graph $G = (V, E)$ where the vertices are the Pauli terms in the operator and edges are placed between commuting terms.

The amount of commuting terms found using this method will of course depend on the operator in question, but evidence from numerics on molecular electronic Hamiltonians with up to 50 thousand terms indicate that the ratio of total terms over number of commuting groups scales "at least linearly with the number of qubits" [226]. Jena *et al.* [227] also conjecture that this ratio is linear with respect to the lengths of the operators and provide numerical evidence for molecular Hamiltonians.

4.4 Parallelisation

In this section I will discuss my parallelisation of the SELECT stage of LCU, as the parallelisation of the PREPARE stage will significantly depend on the particular state preparation method chosen, but I note that a reduction in depth by a factor of $O(n)$ with $O(n \log n)$ qubits using fanout is achievable [228, 229]. In the usual application of SELECT (Fig. 4.1) the unitaries are applied in sequence, as is required by the complex controls on the ancilla register storing the coefficients. As we are applying a polynomial number of terms, the PREPARE ancilla register will only be of logarithmic size, allowing us to use fanout to parallelise the action of gates on this register while only paying a $n \log n$ qubit cost. In the rest of this section I will describe how I use this fanout of the ancilla register along with transformations on the system register to facilitate parallelisation of the SELECT stage. I also provide a diagram summarising the required preprocessing steps and circuits of the parallelisation described in this section in Appendix E.

To parallelise unitaries that are not already acting on disjoint sets of qubits on the system register, we can perform a unitary U to transform the action of the unitaries on the system register into action on disjoint qubits. The set of unitaries $\{U_i\}$ must satisfy two conditions in order for this to be possible:

- They must all commute, $[U_i, U_j] = 0$
- They must form a linearly independent set, $\prod_{\{i\}} U_i \neq U_j \in \{U_i\}$.

We do this by first partitioning the terms of the Hamiltonian into commuting groups as described in Section 4.3.4, and then use a greedy search algorithm to find linearly independent sets within each of those commuting groups.

For a Hamiltonian H, I define the filling factor $\mathcal{F}$ as the average size of the sets of terms that satisfy these conditions, after placing all the terms into such sets, normalised by the maximum applicable size (which is the number of qubits in the main register).

For the case where the unitaries in SELECT are Pauli operators P_i, we can partition the operators into linearly independent commuting sets, and then perform Clifford transformations to transform the terms in each set into single qubit Pauli operators as described in Section 4.3.4. We can then apply these controlled Pauli operators in parallel from the ancilla registers, applying up to n terms simultaneously, [1] and then perform the inverse Clifford transformation, as depicted in Fig. 4.3.

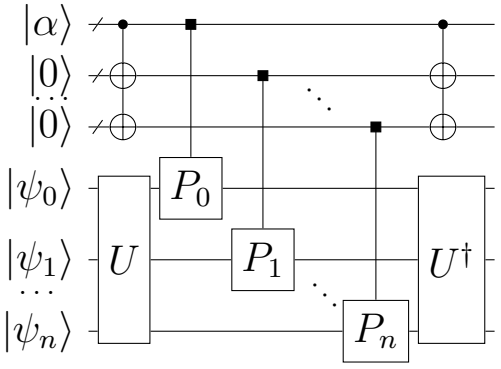


Figure 4.3: Applying Pauli operators in the SELECT step of LCU in parallel, by using fanout and a Clifford transformation U. The square controls indicate multiple controls on the ancilla register given the index of the Pauli operator.

The issue with this is that the linear depth saving from applying n terms in parallel is offset by the depth of the Clifford transformations, which is $O(n)$ in the worst case when using reversible compilation [230]. I therefore present an alternative compilation of Clifford circuits using adaptive circuits and teleportation with $3n$

[1] If the size of the commuting group is $m > n$, it can still be applied in one step by enlarging the system register to m qubits by appending qubits in $|0\rangle$ and performing a Clifford circuit on all m qubits (this also requires increasing the size of the fanout).

ancilla that is constant depth in Section 4.4.1.

The depth reduction achieved is determined by the filling factor $\mathcal{F}$. I conjecture that this is $O(n)$ in the case of practical molecular Hamiltonians and provide evidence from the literature based on the size of commuting groups in Hamiltonians for this in Section 4.3.4. In Fig. 4.4 I demonstrate the factor of improvement in T depth for a series of molecular Hamiltonians by performing the partitioning into parallelisable sets, with the slope of the graph indicating an approximately constant $\mathcal{F} \approx 0.5$ for molecular Hamiltonians in this regime. I also performed the partitioning for 'Hamiltonians' containing only Pauli X terms (as would be found in QROM circuits) up to 14 qubits, and find $\mathcal{F} \approx 1$ for this case (see Appendix D).

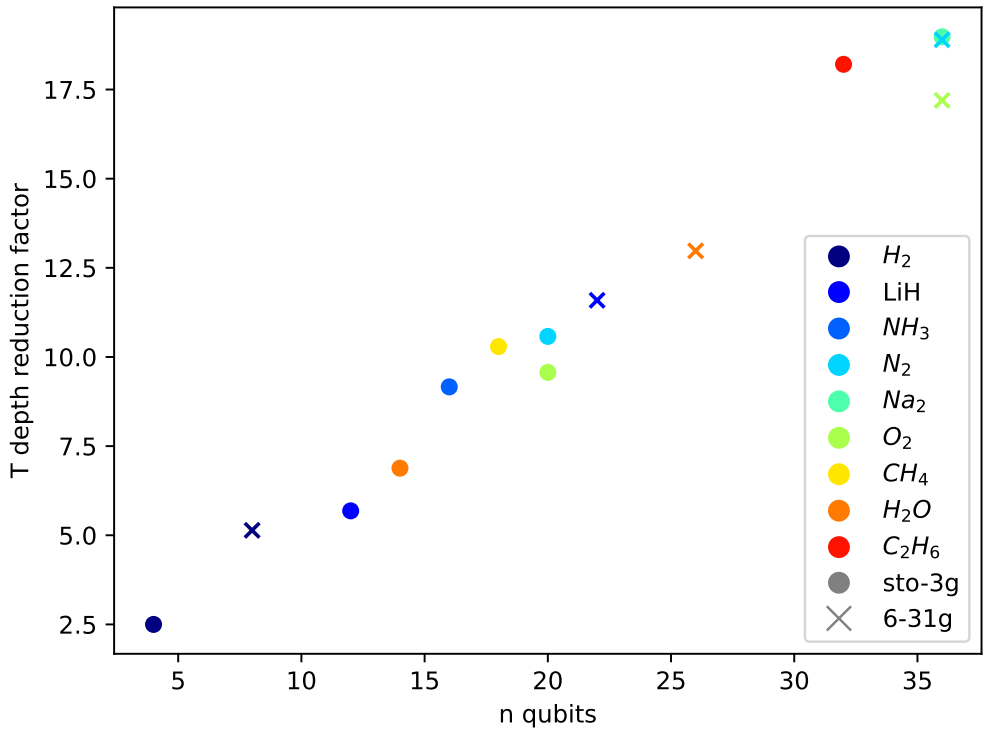


Figure 4.4: T depth reduction as determined by the average size of the parallelisable sets found in molecular Hamiltonians up to 36 qubits. The slope indicates an approximately constant $\mathcal{F} \approx 0.5$ for molecular Hamiltonians in this regime. Colour indicates the molecule and marker indicates the choice of basis.

Given this partitioning into parallelisable sets, the space-time saving is a factor

of $O\left(\frac{n\mathcal{F}}{\log n}\right)$. Although the saving will be very significant in the high-n regime, I find that even in the low-qubit regime, we can still achieve large speed-ups and a reduction in the overall space-time volume. For example, the 26-qubit Hamiltonian of H_2O in the 6-31G basis contains 13884 terms, which can be applied in 1070 parallelised steps by using 468 qubits, resulting in 13 times fewer layers of multi-controlled Pauli gates for a qubit increase of 11.7 times, which slightly reduces the space-time volume by a factor of ~ 1.1. See Section 4.4.4 for further discussion on how the space-time volume of the parallelised subroutine compares to the serial version when taking into account the space required by T factories.

4.4.1 Performing Clifford Circuits in Constant Depth

In order to reduce the depth of the required Clifford circuits, we can use techniques from [231] that apply them in constant depth using a constant number of resource stabiliser states on $O(n)$ ancilla. This method is based on the observation that any Clifford circuit is equivalent to circuits that contain a sequence of stages containing the same kind of gates, for example, Aaronson and Gottesman provided an 11 stage compilation H-C-P-C-P-C-H-P-C-P-C where -H-, -P-, and -C- stand for stages composed of only Hadamard, Phase, and CNOT gates, respectively [34]. More recently, techniques for implementing Clifford circuits using only 3 layers of two-qubit gates (the sequence CX-CZ-P-H-CZ-P-H) have been developed [232].

Considering only the non-trivial layers which contain 2-qubit gates, we can perform these layers by preparing states that are stabiliser states of Calderbeck-Shor-Steane (CSS) codes (up to single qubit rotations) [233] on $3n$ ancilla and performing Steane syndrome extraction circuits using these resource states as an input [231], followed by a correction consisting of single qubit Pauli gates (Fig. 4.5).

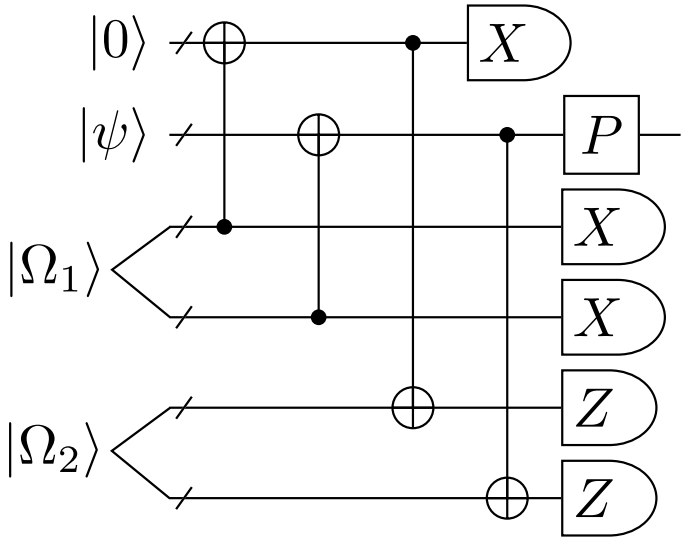


Figure 4.5: Circuit based on Steane syndrome extraction for performing a Clifford C-stage from [231], using the CSS states $|\Omega_{1,2}\rangle$, with a Pauli correction P depending on the measurement outcomes. This circuit uses $5n$ ancilla, but the same operation can be performed with $3n$ ancilla by performing the measurement on $|\Omega_1\rangle$ and then generating $|\Omega_2\rangle$ on the same ancilla.

These resource states can themselves be prepared in constant depth using adaptive circuits with the standard method to prepare stabiliser states of measuring the stabilisers and correcting incorrect outcomes [13], the required measurements of parities can be performed in constant depth using either circuits that are log-depth in the weight of the stabilisers, or in constant depth by noting that parity circuits are equivalent to fanout under a unitary transformation involving one layer of single qubit gates [234] and using results described in Section 4.3.3.
Therefore, the required Clifford transformations can be performed in constant depth, potentially allowing for a linear depth saving from parallelisation.

4.4.2 Replacing the Ancilla Fanout

It may be the case that we are not willing to pay the cost of fanning out the ancilla register n times, either because we are restricted in the number of available qubits, or, for QROM, it may be the case that the input register is of comparable (or larger) size than the output register. I therefore present a scheme for parallelising the action of the input register using a constant factor increase in ancilla qubits and additional constant-depth Clifford transformations.

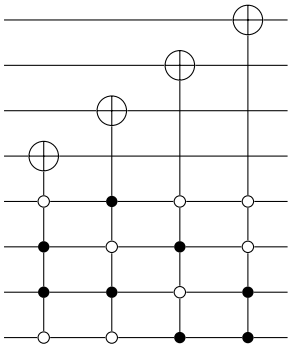


Figure 4.6: Example of a QROM circuit with a complex pattern of positive (filled) and negative (empty) controls, with the action on the output register already diagonalised.

We can use a Clifford transformation on the ancilla register to transform the pattern of controls into the form in Fig. 4.7 (provided the bit strings describing the controls form a linearly independent set, see Appendix D for a discussion on partitioning bit strings into linearly independent sets). This can be seen by the fact that conjugating the circuit in Fig. 4.6 by a CNOT gate acting between qubits i and j results in a row operation adding row i of the controls (thought of as a binary matrix) onto row j, we can therefore use Gauss-Jordan elimination to diagonalise the pattern of controls provided the set of bit strings are linearly independent. The Clifford circuit we need to apply is also only a single C-stage (Fig. 4.5) so has a depth of $\sim 3\times$ less than a full Clifford unitary.

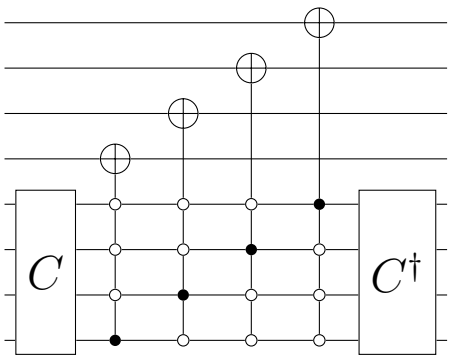


Figure 4.7: Use of a Clifford transformation to 'diagonalise' the action of the controls on the input register.

These controls cannot yet be trivially parallelised. However, each of the gates is now only activated from a state with Hamming weight 1, so by computing a flag containing `hamming_weight==1`, we can then perform the gates on the output

register by performing a Toffoli on a register that contains a fanout of the original ancilla register, and a register containing copies of the `hamming_weight==1` flag, as shown in Fig. 4.8.

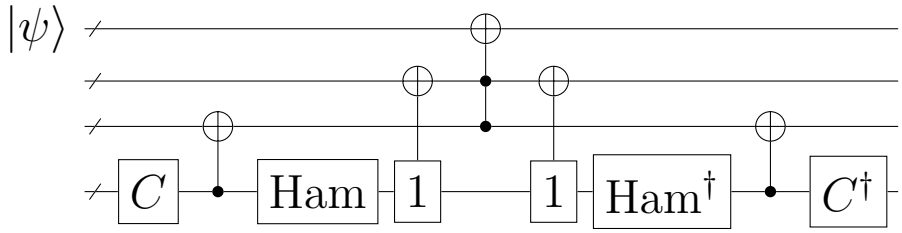


Figure 4.8: Circuit to compute and use the Hamming weight of the inputs to parallelise Fig. 4.7.

This Hamming weight circuit can be done in depth $\log a$, and a constant factor increase in ancilla qubits is required to perform this parallelisation.

4.4.3 Extension to Arbitrary Parallelism

After submitting this work to be published, I followed with further research on how to bypass the limitation of commuting groups, which restricted the maximum possible parallelism possible by the filling factor $\mathcal{F}$ of the Hamiltonian.
You can attempt to bypass the commutation requirement using the observation that for any Paulis $\{P_i\}$

$$[P_i \otimes P_i, P_j \otimes P_j] = 0 \tag{4.6}$$

so provided the state $\rho \otimes \rho$ a transformation can be performed to apply the $P_i \otimes P_i$ simultaneously to both copies by using the unitary

$$U = \sum_i |i\rangle \langle i| P_i \otimes P_i \tag{4.7}$$

However, after measuring the ancilla register, the two copies remain entangled unless we have some way to uncompute the action on the second register, something that seems just as hard as originally applying the LCU in the first place.
We can instead use some other set of operators $\{Q_i\}$, and for all the products to commute

$$[P_i \otimes Q_i, P_j \otimes Q_j] = 0 \tag{4.8}$$

we only need the commutation structure of $\{Q_i\}$ to be the same as that for $\{P_i\}$, meaning that if P_i, P_j (anti)commute then Q_i, Q_j must also (anti)commute, (although their non-zero (anti)commutators need not be the same).

This allows us to 'overlay' the Q_i over our original Hamiltonian as pictured in Fig. 4.9.

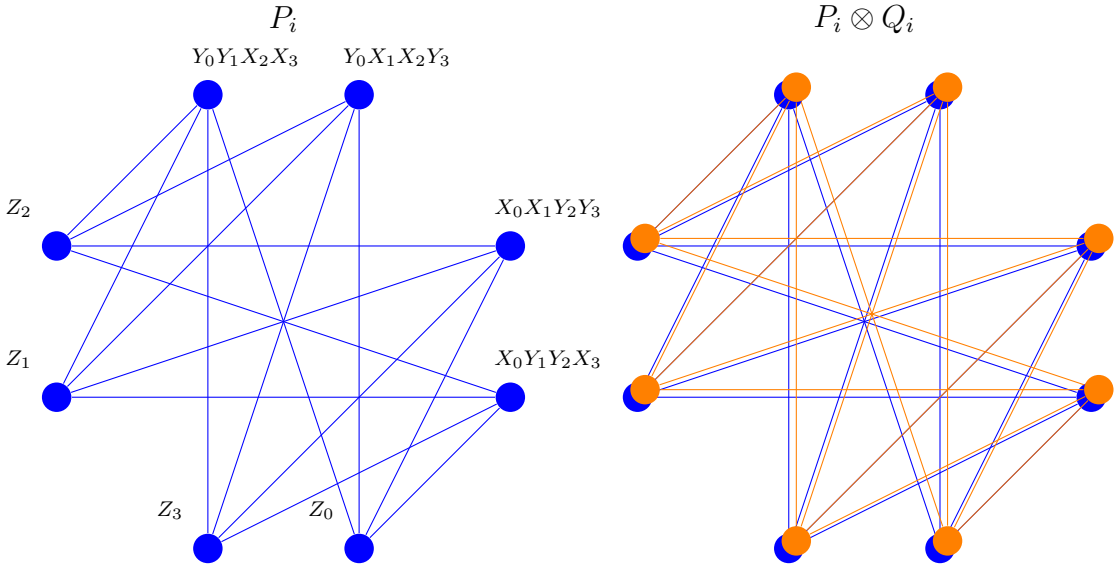


Figure 4.9: Non-trivial part of the anti-commutation graph of the H_2 Hamiltonian (blue), along with the result when the original terms are extended by new terms Q_i (orange). For the second graph, the orange and blue edges cancel, resulting in a fully disconnected graph, meaning all terms commute and can be performed in parallel.

Now it remains to choose $\{Q_i\}$ such that efficient uncomputation is possible, we have two useful freedoms in this choice:

- The Q_i may act on larger number of qubits than the original Hamiltonian
- We can choose the state they act on

This allows us to choose $\{Q_i\}$ that only consist tensor products of Pauli X and Z operators and have them act on the state $|0\rangle$, where the Z's act as the identity, so we only need to uncompute the X's, which we can do efficiently in parallel as the commutation requirement is already satisfied, and we can always add logarithmically

more qubits to our mapping $P_i \to Q_i$ to ensure the Q_i with Z's removed are linearly independent. This allows us to arbitrarily parallelise at the cost of doubling the number of terms we need to apply. Whereas before the maximum speed-up achievable was $n\mathcal{F}$, now we can achieve $m/2$ where m is the number of qubits we are willing to allocate.

An example mapping from $P_i \to Q_i$ is to take every single Pauli in P_i and perform the mapping (preserving commutation relations) onto two qubits

$$\begin{aligned} X &\to X \otimes X \\ Y &\to X \otimes Z \\ Z &\to Z \otimes I \end{aligned} \tag{4.9}$$

Mappings requiring fewer qubits are likely to be possible in specific cases depending on the Hamiltonian.

4.4.4 In fault-tolerant architectures

Until this point, I have only been discussing the parallelisation implemented in terms of logical operations. However, there are some further considerations that must be made when examining the implications of parallelisation in fault-tolerant architectures. I will discuss these in the context of the surface code, but similar considerations apply to other fault-tolerant settings.

In surface code architectures, the available operations are of the form Clifford+T where the T gates are non-transversal and require the use of additional techniques such as magic state distillation [235] to be implemented, and therefore have a higher cost than the Clifford gates, taking up the majority of the computational budget. The LCU parallelisation succeeds in reducing the T-depth of the algorithm without changing the T-count. However, whereas only a factor $O(\log n)$ qubit increase is required in the logical setting, in the fault-tolerant setting, the number of qubits used for magic state distillation must be increased to keep up with the increased

rate of magic state usage required for a speed-up for the parallelised algorithm. However, this additional cost in T factories is modest and also introduces a $\log n$ factor (albeit with a higher constant) in the naïve case where every multi-controlled gate is done separately with $\log n$ T cost. I also note that when the controls of the LCU/QROM circuits are constructed with unary iteration [211], which only results in a constant rate of T state usage regardless of register size, the increase in space for the T factories is further decreased. In Fig. 4.10, I use the Azure Quantum Resource Estimator [236] to compare the space-time volume of the parallel vs serial computation (assuming $\mathcal{F} = 1$) which initially has n qubits in the main register, $\log n$ ancilla qubits, and use of unary iteration for the application of controls. I find orders of magnitude savings in space-time volume for large n.

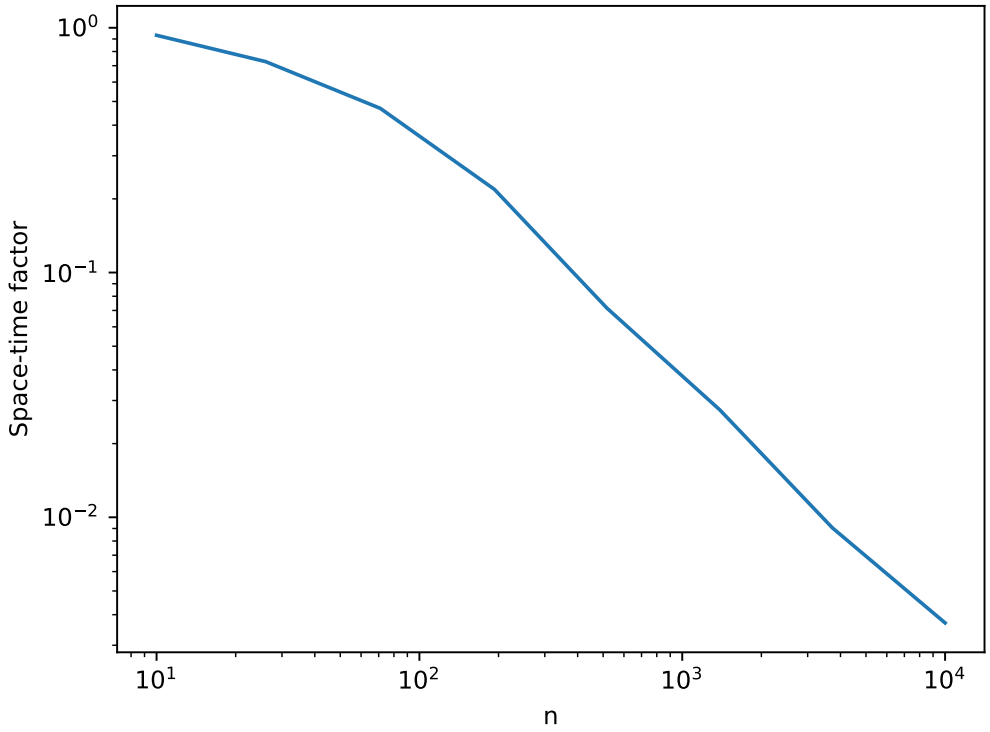


Figure 4.10: Scaling of the improvement in space-time volume of the parallelised algorithm over the serial version for $\mathcal{F} = 1$, when including the space required for additional T factories due to the increased rate that the parallel circuit is able to use them.

The increase in qubits required for T factories may also be mitigated by the following factors:

- In cases where more magic state factories are required for other parts of the computation than are required to keep up with the rate of the serial SELECT procedure, the parallelised procedure is more efficiently able to use magic states at the rate they are produced.
- Magic states can be prepared offline and stored in a quantum memory.
- Magic state preparation has been significantly optimised in recent years [237, 238], and optimisation is likely to continue, further bringing down the cost. Indeed, since the numerics for Fig. 4.10 were performed, approaches able to reduce the space-time overhead of producing magic states by an order of magnitude have been produced [239].

4.5 Discussion and Conclusion

In this work I have produced an effective and low-overhead scheme for reducing the depth of the SELECT subroutine of LCU or data loading via QROM by a factor $O(n\mathcal{F})$, where $\mathcal{F} \leq 1$ is the filling factor denoting the average size of the parallelisable sets of terms to be applied, whilst only increasing the required number of qubits by a factor $O(\log n)$. A key part of this scheme is a method for performing Clifford transformations in constant depth based on adaptive circuits and teleportation. Methods are also presented for the parallelisation of complex patterns of multi-controlled operations with only a constant factor qubit overhead. I provide a numerical study on the parallelisability of SELECT on molecular Hamiltonians up to 36 qubits, and find they all can be reduced in depth by a factor of approximately $n/2$. I also note the procedure is inherently scalable with the number of ancillas available, meaning that if only $O(m \log n)$ ancillas are available for the fanout (as opposed to $O(n \log n)$), then the algorithm will simply perform m operations in parallel. Alternatively, $m > n$ can be chosen, by adding $m - n$ qubits

to the system register initialised in $|0\rangle$, and performing the Clifford transformation on all m qubits. The procedure can also be extended by taking $P_i \to P_i \otimes Q_i$, allowing all terms to commute, and thereby arbitrary parallelisation, with the extra requirement of needing to uncompute the action of the Q_i which can also be done in a parallel fashion with a suitable choice of Q_i.

I believe that the use of parallelisation techniques in quantum algorithms is a fruitful direction for reducing the overheads of quantum computation, particularly when asymptotically optimal algorithms exist for problems such as Hamiltonian simulation, but the current runtime estimates for useful applications can be daunting. Works of this kind could also be of interest from the perspective of certain hardware platforms, as it means that scaling up hardware can reduce runtimes, allowing for offsetting of long gate times.

Further work includes accurate resource estimations of qubit counts and required wall-clock times in the fault-tolerant setting, the extension of the scheme to other families of unitaries, e.g. using matchgate circuits [240] and the generalisation of this scheme to alternative groupings of operators.

Code Availability

Code relevant to this chapter, including python code for preprocessing Hamiltonians and Q# code for the circuits described can be found at on github. The code makes use of the packages OpenFermion [200], Stim [241], and graph-tool [242].

5
Ongoing Work

Contents

5.1 Foreword

This chapter provides results from unpublished work on symmetries in adiabatic state preparation and entanglement guided variational recompilation. The work in this chapter is entirely my own.

In the first section, I explore how leveraging symmetries can improve the efficiency and accuracy of adiabatic state preparation. Motivated by the difficulty of choosing classical input states for quantum algorithms, I produce a toy example where of

two input states, the 'worse' higher energy state can exploit symmetry-avoided transitions to perform better than a lower energy state in terms of final ground state overlap. I then further discuss the use of symmetry in adiabatic state preparation. The second section investigates the use of entanglement measures to guide the variational recompilation of quantum circuits. By incorporating entanglement metrics, I seek to develop more effective strategies for circuit construction. I build on previous work for automatically discovering quantum circuits by incorporating knowledge about entanglement into the gate placement heuristic and find indications of substantial improvements in some regimes.

5.2 Symmetries in Adiabatic State Preparation

The adiabatic theorem in quantum mechanics is summarised by Born and Fock in their 1928 paper as follows: "If one assigns numbers to the states of a system corresponding to the energy levels, the adiabatic theorem asserts that if the system initially was in a state with a specific number, the probability of the system transitioning to a state with a different number during an adiabatic change is infinitely small" [243]. That is, if constructs a time-dependent Hamiltonian $H(s(t))$ where $H(s(0)) = H_0$ some initial Hamiltonian, and at a final time $H(s(T)) = H_f$ some final Hamiltonian, then if a system begins in an eigenstate of H_0 and evolves sufficiently slowly with respect to the energy gaps in $H(s(t))$, then at time T the system will be in the corresponding eigenstate of H_f. The adiabatic approximation of slow evolution compared to the size of the energy gaps will of course always be broken when energy levels are degenerate at some point in the evolution.

We can use this procedure to attempt to prepare interesting states for input into other quantum algorithms, starting with eigenstates of systems we can prepare classically. The motivation for this is the difficulty of preparing initial states for phase estimation, which require a large overlap with the true ground state of a system to be projected into this ground state with reasonable probability. Lee *et al.* discuss this problem and examine using Adiabatic State Preparation (ASP) for this task [244]. They find a correlation $T_{\text{ASP}} \sim \text{Poly}(1/S)$ where S is the overlap

between the initial state and the target state. Notably they demonstrate in their Fig. 2c, reproduced here as Fig. 5.1 that low-energy states from classical ansatz can be outperformed by higher energy states for state preparation on quantum computers. I therefore examine the use of symmetries within ASP, construct a toy problem which reproduces the phenomenon of a high energy state outperforming a low energy one in ASP due to its symmetries, and further discuss how symmetries might be used within ASP.

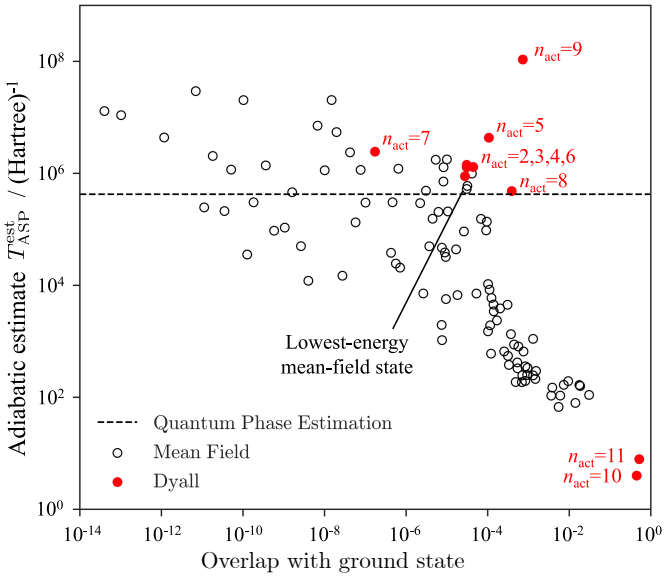


Figure 5.1: Estimate of ASP time against initial overlap for states drawn from two classes of classical ansätze for the reduced active space of a 24 qubit simulation of a [2Fe-2S] cluster. Horizontal line gives the time required to perform Quantum Phase Estimation. Figure 2c from [244]

5.2.1 Disallowed Transitions

A point sometimes overlooked in statements about adiabatic evolution is that transitions, even in the case of closing gaps, can be avoided if those transitions are disallowed due to symmetries. For an adiabatic evolution with Hamiltonian $H(s(t))$ which at all times commutes with a symmetry operator Y, $[H(s(t)), Y] = 0$ and has instantaneous eigenstates $|\phi_{E_n}^{\lambda_\alpha}\rangle$ satisfying

$$H(s(t))\,|\phi_{E_n}^{\lambda_\alpha}\rangle = E_n(s(t))\,|\phi_{E_n}^{\lambda_\alpha}\rangle \tag{5.1}$$

$$Y\,|\phi_{E_n}^{\lambda_\alpha}\rangle = \lambda_\alpha\,|\phi_{E_n}^{\lambda_\alpha}\rangle \tag{5.2}$$

It is easy to prove that the matrix elements of the Hamiltonian must satisfy the condition: [245]

$$\langle\phi_{E_m}^{\lambda_\beta}|H(s(t))|\phi_{E_n}^{\lambda_\alpha}\rangle\,(\lambda_\beta - \lambda_\alpha) = 0 \tag{5.3}$$

and also for the time derivative of the Hamiltonian

$$\langle\phi_{E_m}^{\lambda_\beta}|\dot{H}(s(t))|\phi_{E_n}^{\lambda_\alpha}\rangle\,(\lambda_\beta - \lambda_\alpha) = 0 \tag{5.4}$$

This guarantees that over the course of the Adiabatic State Preparation (ASP), no transitions can be driven between two states with different symmetries. We can use this fact to design the initial Hamiltonians/states for ASP.

5.2.2 Toy Example

Typically, classical states used as an input for quantum algorithms are chosen to be the 'best' available with classical techniques, often measured by their energy, with lower energy states being favourable due to the variational principle. Motivated by the demonstration of higher-energy states outperforming lower energy states in terms of required adiabatic evolution time in [244], here I produce a toy example of ASP where it is possible to use symmetries to make a higher energy initial state produce an output state closer to the desired ground state of the final Hamiltonian than when using a lower energy initial state. The example is producing the ground state of a 1D triple-well potential (within a harmonic trap) through ASP by starting from the eigenstates of the bounding harmonic potential, according to the schedule

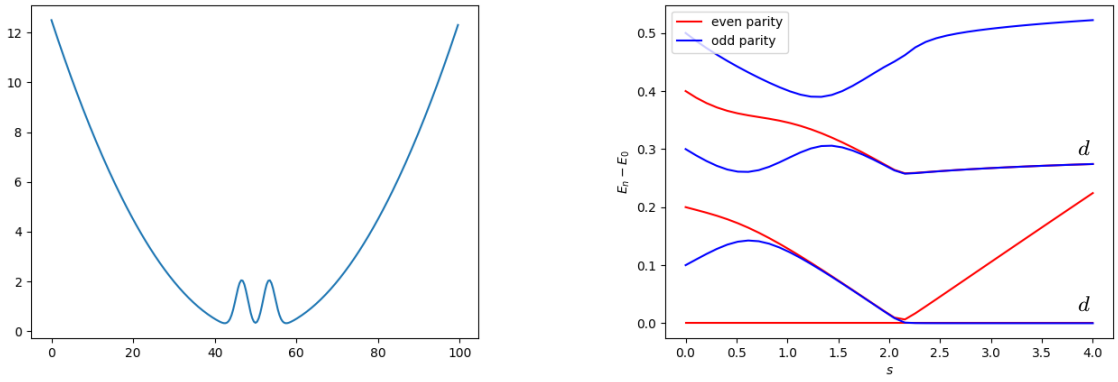
$$H(s) = H_{\text{harm}}\left(1 - \frac{s}{4}\right) + \frac{s}{4}H_{\text{triple well}} \tag{5.5}$$

with $s \in [0, 4]$ and the triple well potential is

$$H_{\text{triple well}} = C\left(x^2 + b\left(e^{-\frac{(x-\Delta)^2}{2d^2}} + e^{-\frac{(x+\Delta)^2}{2d^2}}\right)\right) \tag{5.6}$$

Applied in a box of size $L = 100$ with parameters $C = 0.005, b = 100, d = 1.5, \Delta = L/30$.

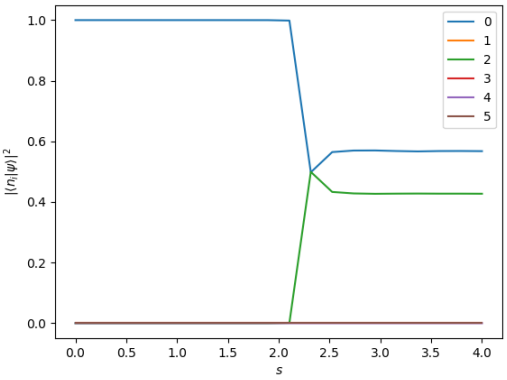
The final potential and instantaneous eigenvalues of the Hamiltonian are shown in Fig. 5.2 where an avoided crossing appears between two even parity states at $s \approx 2.1$. This crossing induces transitions when performing ASP starting from the even-parity initial ground state. However, starting from the odd-parity initial first excited state (or equivalently, an odd parity ground state of another Hamiltonian, e.g. a double-well) allows for easy preparation of the triple-well ground state. Fig. 5.3 shows the overlaps of the evolving state with the instantaneous eigenstates, whereas Fig. 5.4 shows the overlaps with the eigenstates of the final Hamiltonian. The energy differences of the initial states from the desired ground state are $\Delta E_{\text{even}} = 0.597$ and $\Delta E_{\text{odd}} = 1.077$. As the odd parity initial state is able to produce the desired final ground state, despite having a larger energy, this reproduces the phenomenon of a higher-energy initial state outperforming a lower energy one in terms of ability to produce a desired ground state using ASP.



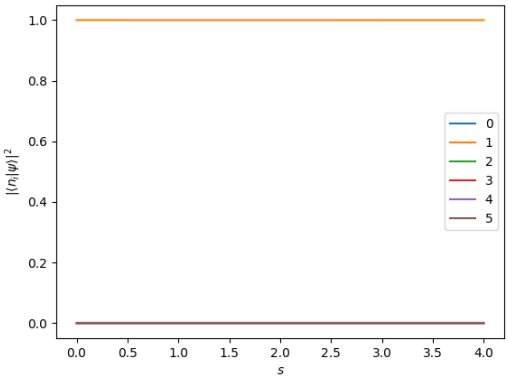
(a) Triple-well potential, the target potential for ASP.

(b) Spectrum of the Hamiltonian over the course of ASP, eigenstates are coloured by parity. Final states marked with d are a degenerate combination of an even and an odd state.

Figure 5.2: Target potential (Eq. (5.6)) and spectrum of the Hamiltonian over the course of the evolution, a phase transition can be seen at $s \approx 2.1$ where the barrier potentials become high enough to disfavour wave functions with amplitude in the central region.

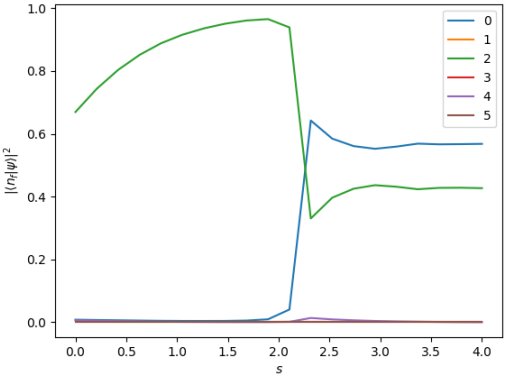


(a) ASP when $|\psi_{init}\rangle$ is the ground state of the harmonic potential

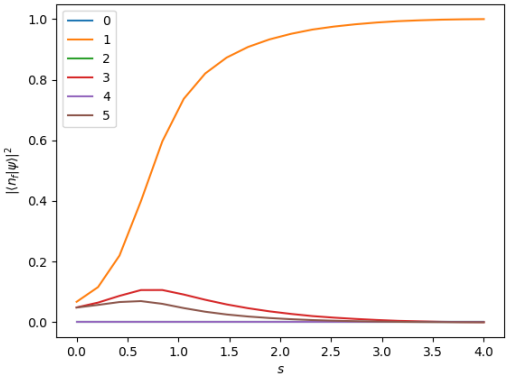


(b) ASP when $|\psi_{init}\rangle$ is the first excited state of the harmonic potential, the system remains in the instantaneous first excited state throughout the evolution, with negligible overlap with the other eigenstates.

Figure 5.3: Overlaps with the instantaneous eigenstates of the Hamiltonian over the course of the evolution



(a) ASP when $|\psi_{init}\rangle$ is the ground state of the harmonic potential, ground and 2nd excited states switch after the phase transition



(b) ASP when $|\psi_{init}\rangle$ is the first excited state of the harmonic potential

Figure 5.4: Overlaps with the eigenstates of the final Hamiltonian over the course of the evolution

5.2.3 Producing a Symmetry-respecting H_0

Electronic structure Hamiltonians where relativistic effects can be neglected do not contain operators relating to the spin of the constituent electrons of the molecule. This results in these problems having symmetry operators of the total spin S^2 and component of the spin in a particular direction S_z. Additionally, molecular

problems have point group symmetries relating to the spatial symmetries of the layout of the nuclei (in the Born-Oppenheimer approximation). These point group symmetries are taken into account by conventional quantum chemistry methods such as Hartree-Fock (HF) methods.

The question that remains is how to produce an initial Hamiltonian H_0 given a final Hamiltonian H_1 and symmetry operator(s) $\{S_i\}$ such that the H_0 commutes with all S_i. Here I discuss and implement a few ideas in this direction on a linear hydrogen chain H_4 Hamiltonian in the STO-3G basis using 8 qubits. For comparison, Fig. 5.5 shows the instantaneous spectrum of the 30 lowest eigenstates when performing ASP on this system, with a final Hamiltonian produced via orbitals found via HF, and using a non-symmetry respecting initial Hamiltonian that is just a sum of Z operators with correct signs to produce the HF computational basis state. That is, given a computational basis state with bits b_i, a simple Hamiltonian with this state as the ground state is

$$H_N = \sum_{i=0}^{n} (-1)^{b_i} Z_i \tag{5.7}$$

Due to the fact that the HF solution has a high overlap with the true solution, there is no gap that closes midway through the evolution.

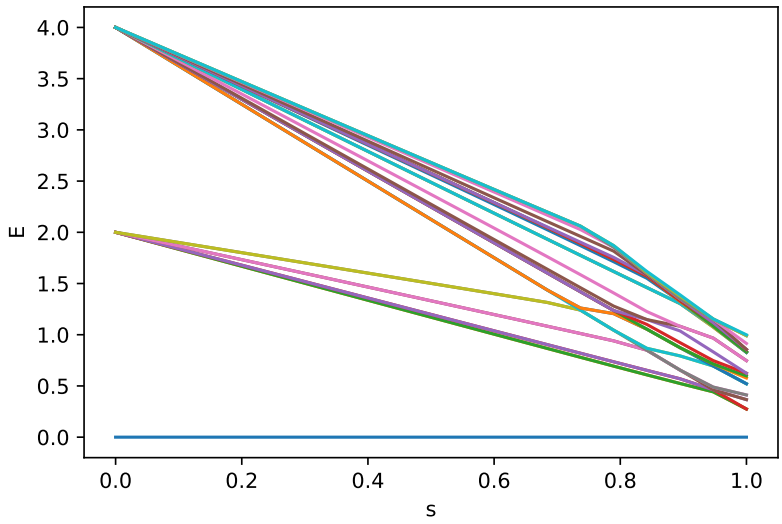


Figure 5.5: ASP spectrum using an initial Hamiltonian of the form H_N for H_4

Parent Hamiltonians

The idea of parent Hamiltonians [246, 247] revolves around the construction of a Hamiltonian from a given state, such that the state is an eigenstate of the produced Hamiltonian. This is guaranteed by using the covariances of operators with respect to that state, as covariances of any operator with a Hamiltonian will all be 0 for an eigenstate of that Hamiltonian. Given operators in a pool $O_k \in \mathcal{P}$, the covariances with respect to a state $|\psi\rangle$ can be calculated as:

$$\langle O_j, O_k \rangle_\psi = \langle \psi | O_j O_k | \psi \rangle - \langle \psi | O_j | \psi \rangle \langle \psi | O_k | \psi \rangle \tag{5.8}$$

A covariance matrix over the operator pool space can then by found $L_{ij} = \langle O_j, O_k \rangle_\psi$. The kernel of this matrix gives sums of operators for which the state $|\psi\rangle$ is an eigenstate. The similarities of this discussion with the eigenstate-finding approach of Chapter 2 is no coincidence. In a sense, using covariances to find states is the mirror image of using them to find parent Hamiltonians.

Usually, the operator pool $\mathcal{P}$ is chosen to be a local pool of operators, but perhaps it is possible to modify the pool such that it only uses operators that commute with the symmetry operators $\{S_i\}$. This could be done by finding operators in the kernel of the Liouvillain $[S_i, \cdot]$, and using these as the operator pool for parent Hamiltonian construction.

However, there are two major downsides to this, the first is that the basis of operators required to fully specify the above Liouvillian is the Krylov subspace of the symmetry operator, and is in general large. The second obstacle to using parent Hamiltonians is that the Hamiltonian produced has no guarantee to have $|\psi\rangle$ as the ground state, rather, it could be any excited state, producing difficulties for adiabatic methods that intend to transform from ground state to ground state.

Construction from Symmetry Operators

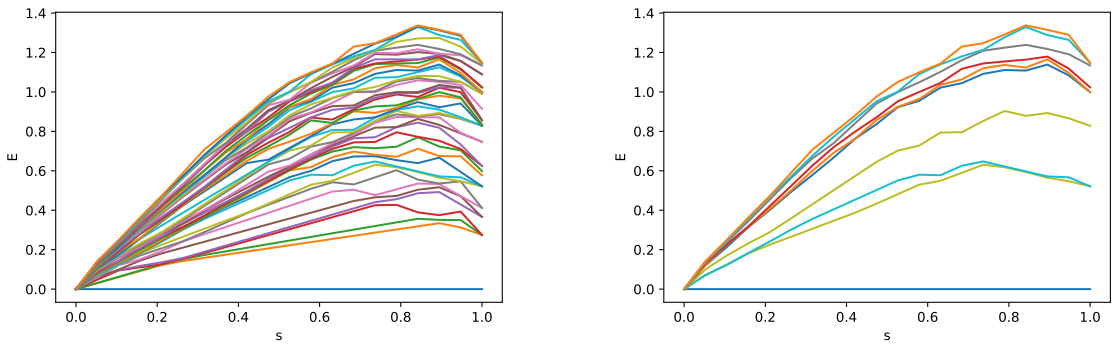
Another way to construct H_0 is to only use information about symmetries that you already know about your initial state. For example, suppose that we are given a state with symmetry operators A and B, that has eigenvalues a and b of these

operators respectively. It is possible to create a Hamiltonian H_0 that has the ground state only be states that have these symmetry eigenvalues.

An operator that satisfies this condition is:

$$H_0^s = (aB - bA)^2 \tag{5.9}$$

We can examine the instantaneous spectra over the evolution from H_0^s to the final Hamiltonian. This is shown in Fig. 5.6, where the left plot shows the full spectrum for the lowest 42 energy states, and the right where states of incorrect symmetry have been removed as these cannot be accessed by the adiabatic evolution. However, the issue with this method is that the initial degenerate subspace will in general be exponentially large, unless the Hamiltonian has an extensive number of symmetries (in which case it can often be classically easy to solve). Therefore, it is necessary to know a state within this subspace that will have a high overlap with the ground state that is deformed into the ground state of H_1.



(a) Instantaneous spectrum for lowest lying states from H_0^s to H_1

(b) Instantaneous spectrum for lowest lying states from H_0^s to H_1, states of incorrect symmetry removed.

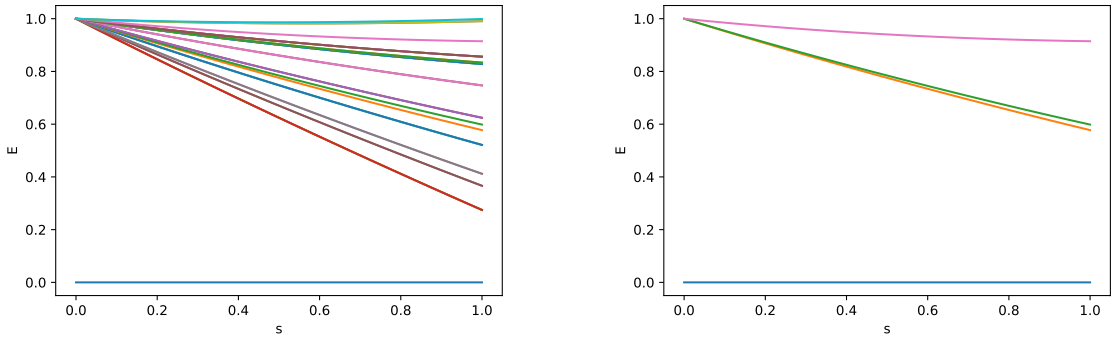
Figure 5.6: Instantaneous spectra for ASP with an initial Hamiltonian H_0^s constructed only from symmetry operators $A = S^2, B = S_z$, and their known eigenvalues in the ground state. (Note that when $a, b = 0$ H_0^s can be modified to use some constant a', b').

Projectors

Instead of only using information related to the symmetries of the initial state, we could instead use the fact that we have a shallow circuit to prepare the initial state,

and use rank-1 Hamiltonian simulation techniques [248] to enable us to choose $H_0^p = 1 - |\psi_0\rangle\langle\psi_0|$. There are results in the literature regarding the use of projectors as initial (or final, as ASP methods are reversible) states in adiabatic evolutions. The primary example is in Grover search, where the adiabatic method initially appears require an evolution time $O(N)$, but this can be reduced to the optimal $O(\sqrt{N})$ by choosing a scheduling function that produces a faster evolution away from the gap at $s = 1/2$ [249]. Additionally, bounds can be placed on the gap in general adiabatic evolutions where projectors are used [250], being proportional to the overlap between the initial state and the ground state of H_1. Furthermore, there is also previous work in the context of constraint satisfaction problems about the use of adiabatic search in a subspace [251], where an additional speed-up can be found due to the lower dimensionality of the relevant subspace. This is essentially the situation that adiabatically evolving from a projector in a symmetry defined subspace is, the method would essentially just be performing Grover search in the subspace with the correct spin symmetries.

Of the methods demonstrated here, it is clear that the projector based Hamiltonian performs best in terms of gap size to accessible states for this example. However, this method also requires a good initial overlap to the desired ground state. It is worth pointing out that the bound on this gap applies to the gap to the first excited state, and is therefore improved when the evolution is restricted to a subspace (as we can see in Fig. 5.7). Overall, it is unsurprisingly clear that only using a small amount of symmetry information is not enough to substantially improve ASP, but it is an interesting question of how these ideas can be incorporated into specific examples of ASP schemes for more realistic final Hamiltonians. Considering how symmetry constraints can be combined with information about the initial state would also be valuable, as the projector and symmetry operator Hamiltonian constructions in this section only use these pieces of information separately.



(a) Instantaneous spectrum for lowest lying states from H_0^p to H_1

(b) Instantaneous spectrum for lowest lying states from H_0^p to H_1, states of incorrect symmetry removed.

Figure 5.7: Instantaneous spectra for ASP with initial Hamiltonian H_0^p defined by the projector onto the HF state $H_0^p = 1 - |\psi_{\text{HF}}\rangle\langle\psi_{\text{HF}}|$

5.3 Guiding Recompilation with Entanglement Measures

Variational recompilation, as discussed previously in Section 2.5.1, is used to optimise quantum circuits for specific hardware. By adjusting circuit parameters, the method attempts to find adapted forms of idealised circuits to account for real-world quantum noise and hardware constraints. Variational recompilation is particularly valuable for near-term quantum devices, as it helps improve performance while navigating the limitations of NISQ hardware. In this section, I will further discuss recompilation methods, involving using some results on generalisation in learning unitaries along with my own ideas on leveraging entanglement measures to guide the automatic discovery of quantum circuits.

5.3.1 Generalisation in Learning Unitaries

Work from Caro *et al.* [82, 83] proved it is possible to learn a whole unitary by using training states drawn from so called *locally scrambled* ensembles. For our purposes, we will use the simple example of Pauli product states $\bigotimes |\psi_i\rangle$, where $|\psi_i\rangle$ are single qubit states that are randomly chosen eigenstates of one of the Pauli operators. If we are able to produce a unitary that has the correct action

on polynomially many of these states[1], it can be shown that this learned unitary will also have the correct action on Haar-random states, that is completely random states $U_{\text{Haar}}|0\rangle$ where U_{Haar} is a random uniformly sampled $SU(2^n)$ unitary matrix. This would be done by the circuit shown in Fig. 5.8.
The intuition for this is that one can predict the action of a unitary $U|\phi\rangle$ so long as the coefficients in $|\phi\rangle$ are sufficiently random and spread out across the 6^n Pauli product states, as only poly(n) training product states are required to approximately know $U|s\rangle$ for most of the 6^n states.

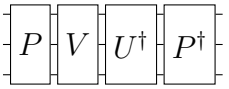


Figure 5.8: Single state recompilation of U using a state preparation unitary P and parametrised unitary V. When P is instead sampled from an ensemble of training states, this circuit can be used for full unitary compilation as in Fig. 5.9.

I reproduced this generalisation for a simple example of performing a full unitary compilation of a circuit for creating a GHZ state $|\text{GHZ}_n\rangle = \frac{1}{\sqrt{2}}(|0\rangle^{\otimes n} + |1\rangle^{\otimes n})$ from the $|0\rangle$ state on 5 qubits, using 15 training states, using either random computational basis states or random Pauli product states as shown and defined in Fig. 5.9. As expected, learning on computational basis states fails to capture the action on random states due to the lost phase information, whereas training on Pauli product states leads to generalisation.

5.3.2 Using Entanglement Measures for Recompilation

The goal in recompilation is essentially to undo the action of the unitary we wish to compile and return to the original (see Fig. 2.2). In the standard case where the original state is $|0\rangle$, or when it is any other product state, this task is essentially the same as removing the entanglement that the original unitary produced. Removing the entanglement, even if we do not produce the desired state directly, is also desirable as it allows us to be satisfied with any product state which we can then

[1] A number of training states scaling as $T \log T$, with T the number of parameters in our learned unitary. [83]

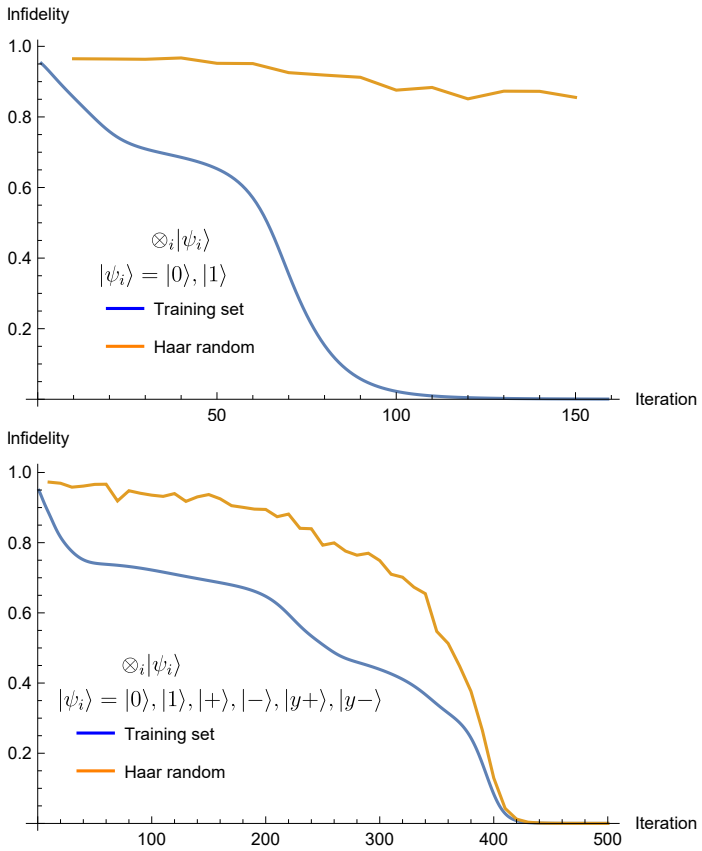


Figure 5.9: Learning a 5 qubit GHZ state circuit over the whole unitary using 15 random training states. The top plot shows training with computational basis states, which fails to generalise to random states. Whereas the bottom plot uses Pauli product states and succeeds in producing a low error on Haar random states after training.

easily transform to our target state by a layer of single qubit rotations. We can track the extent to which each qubit is entangled to the rest of the system using the proxy of the Rényi 2-entropy $S_2(\rho) = -\log \mathrm{tr}(\rho^2)$ [252] (although in practice I will sometimes just use the closely related quantity $\mathrm{tr}(\rho^2)$) and extract reduced density matrices (RDMs) of each qubit, to plot how the entanglement of each qubit evolves over a recompilation. Fig. 5.10 shows the recompilation of a 5-qubit circuit for creating a GHZ state, showing both the infidelity of the recompiled circuit and the amount of entanglement of each qubit with the rest, as determined by $1 - \mathrm{tr}(\rho_i^2)$ which is 1 when fully entangled and 0 when fully unentangled.

In this case, the optimisation is stuck, but we can see that two of the qubits are completely unentangled, and we essentially only need to apply one more two-qubit operation to the qubits denoted by the blue and orange curves, or change parameters

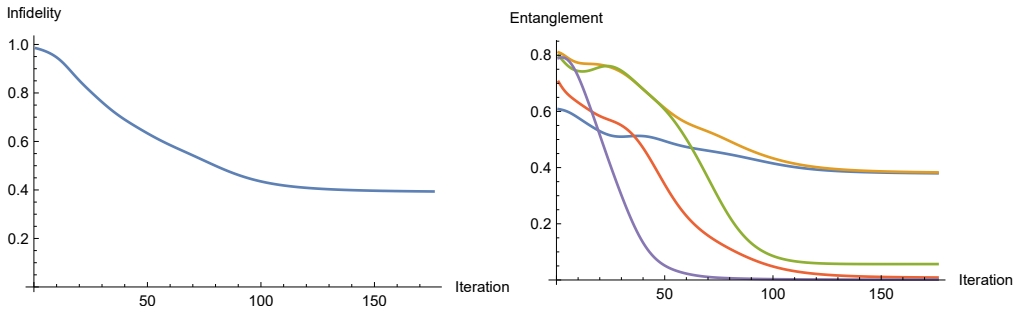


Figure 5.10: Single state compilation of a 5-qubit GHZ creation circuit, using a smaller ansatz than in Fig. 5.9. Also shown is a measure of the entanglement of each qubit with the rest as $1 - \mathrm{tr}(\rho_i^2)$. The optimisation reaches a local minimum when only two qubits remain substantially entangled.

of gates that cause entanglement between them in order to completely remove the entanglement remaining in the system.

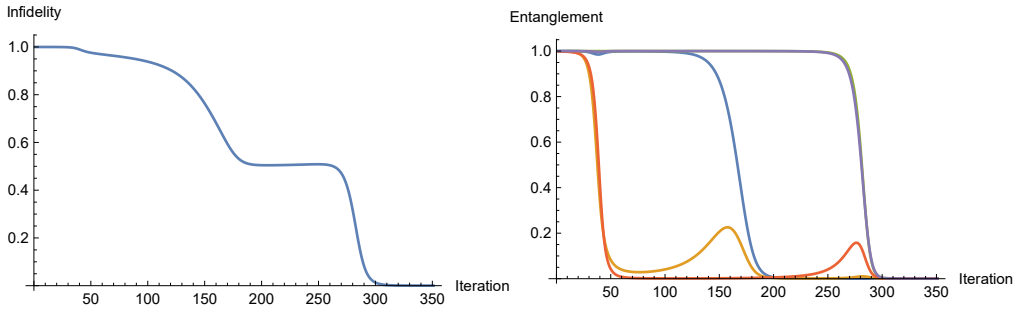


Figure 5.11: Single state compilation of a 5-qubit GHZ creation circuit, as in Fig. 5.10, but now using entanglement measures as a cost function $C = -\sum_i \mathrm{tr}(\rho_i^2)$. The flat region on the infidelity curve corresponds to a period where two qubits remain entangled, but due to the fixed ansatz structure, require an intermediary (at the time unentangled) qubit to interact with both, as shown by the orange 'hump' in the right plot.

Fig. 5.11 shows the action on just one state using the cost function of the sum $C = -\sum_i \mathrm{tr}(\rho_i^2)$, we can see that the flat regions are due to entanglement between spatially separated qubits with unentangled qubits between them. This phenomenon is a result of the fixed ansatz structure used in the recompiled unitary. The simplest way to remove entanglement may be to perform a two-qubit gate between qubits A and B, but as there are no gates connecting this pair in the fixed ansatz, a qubit C which has a low degree of entanglement must become an intermediary, resulting in the orange 'humps' in the figure. This effect means that using entanglement measures as a cost function directly with a fixed ansatz often results in challenging

optimisations prone to local minima.

This motivates an approach where we adaptively construct an ansatz, placing gates using heuristics guided by entanglement measures. To give context to the following discussion, in the following subsection I will give a brief overview of the work from Meister *et al.* [253] which adaptively constructs ansätze for recompilation problems. I will then adapt the gate placement algorithm to use entanglement information.

5.3.3 Recompilation with Automatically Generated Ansätze

Here I briefly discuss the work by Meister *et al.* [253] on automatically generating recompiled circuits over the course of a variational recompilation by randomly placing gates into a circuit in *moves*, which refer to the most basic possible modification of a circuit structure, i.e. the insertion of one additional gate at some position in the gate sequence.

Instead of using training states, this work evaluates the closeness of the result to the target unitary using the *Hilbert-Schmidt test* shown in Fig. 5.12 whereby the n qubits V acts on are entangled with another n qubits. The target state of all the output qubits is $|0\rangle$ which indicates the two unitaries are the same (up to a global phase).

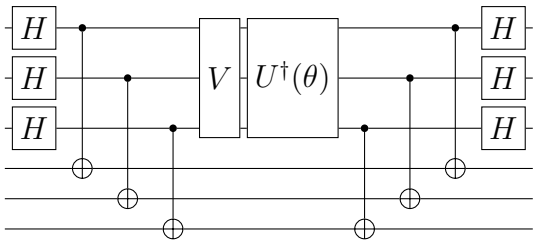


Figure 5.12: Hilbert-Schmidt test for recompiling a unitary V.

The scheme for generating circuits over the course of an optimisation is performed in a series of random steps: In a random step, a number of N_{moves} randomly chosen moves are applied to the circuit, and only then are the parameters of the gates added. This essentially means adding N_{moves} random gates to the circuit. Because many of the added gates are potentially not contributing to the reduction in energy, the algorithm will subsequently attempt to remove as many of the newly added gates as possible after each random step using heuristics such as removing gates

with small angles, or by exhaustively checking which gates can be removed without paying a large energy penalty. When drawing a random move, the moves to be applied are first sorted into two groups, one containing only moves with single-qubit gates, and the other containing only those with two-qubit gates. To draw a random move, they first choose one of the groups with equal probability, and then uniformly draw a move from the chosen group.

5.3.4 Entanglement Between Pairs of Qubits

In order to guide the progress of circuit construction, we need a more powerful measure of entanglement than the single qubit Rényi entropies used in Fig. 5.11. Imagine we have constructed an RDM ρ_{12} for two qubits from measurements collected from classical shadows [187]. (This same measurement data also allows us to calculate the single qubit RDMs ρ_1, ρ_2). $S(\rho_{12})$ does not tell us about the entanglement between qubits 1 and 2, but only about how they are entangled to the rest of the computer. In order to quantify this, we can use the faithful entanglement measure, *squashed entanglement* [254]

$$E_{12}^{sq} = \frac{1}{2} \min_{\rho_{123} \to \rho_{12}} (S(\rho_{13}) + S(\rho_{23}) - S(\rho_{123}) - S(\rho_3)) \tag{5.10}$$

Where 1 and 2 denote two individual qubit subsystems and 3 the rest of the system. Instead of minimising, if we just take ρ_{123} as the state of our whole quantum computer which is (ideally) pure, and therefore assume $S(\rho_{123}) = 0$. Then this becomes:

$$E_{12} = S(\rho_1) + S(\rho_2) - S(\rho_{12}) \tag{5.11}$$

Which is just the mutual information between qubits 1 and 2, $I(1:2)$
Mutual information being a good measure of entanglement comes from the overall state being a pure state, meaning we have no classical correlations and therefore entanglement is only source of correlation. In fact, despite our discussion so far focussing on entanglement, our recompilation task also needs to remove all correlations between qubits in the output state, so this measure of correlation will be sufficient for guiding gate placement when generating recompiled circuits.

5.3.5 Guiding gate placement with Correlation measures

We can easily calculate two qubit entropies, e.g. using classical shadow techniques, by calculating expectation values of all possible 2-local Paulis [187]:

$$E_{ij} = \frac{1}{4}(1 + \sum_{a,b\in\{x,y,z\}} P_i^a P_j^b) \tag{5.12}$$

Then we can use these measures to guide gate placement when constructing an ansatz. We guide the random search when placing two qubit gates by weighting with the mutual information between qubits, as lower values indicate that these qubits do not need to 'talk to each other' as much as ones with higher correlation. I therefore performed recompilation on n-qubit QFT circuits with all-to-all connectivity, using the training scheme described in Section 5.3.1 with random Pauli product states and averaging correlations over all the training states. This scheme is required over the Hilbert-Schmidt method used originally in [253], as the entangled second register of qubits makes it challenging to track the correlations in the output state.

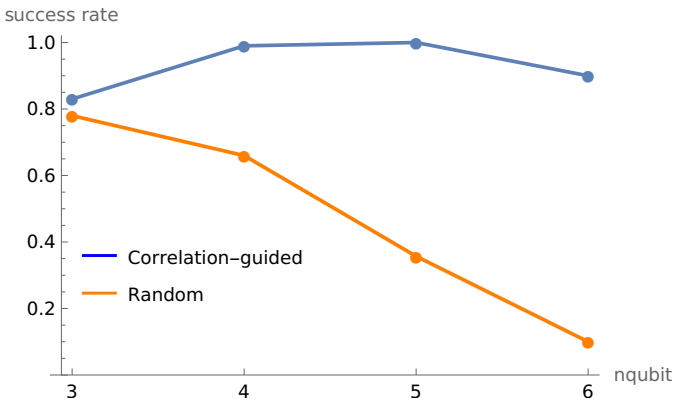


Figure 5.13: Comparison of success rate between random and correlation-guided gate placement (finding a valid recompilation to an error tolerance of 1%) over 20 runs on recompiling n qubit QFT circuits

Fig. 5.13 shows a comparison of success rate (finding a valid recompilation to an error tolerance of 1%) over 20 runs on this problem, using completely random gate placement, versus correlation-guided gate placement, showing how the search of architecture space is enhanced by using information from correlations. In this example, hyperparameters were changed from the values used in the code from [253]

to reduce the numbers of gates added per step to produce a harder task for the optimiser, allowing correlation-guiding to be more useful. In reality, it is possible to get good results for this number of qubits without using correlation-guiding but for larger numbers of qubits, the optimisation of the parameters within the circuits becomes sufficiently difficult that both methods struggle to find good solutions.
An interesting avenue for further work includes further investigation on how to use this for situations with limited connectivity between qubits. Here one would need some heuristic on how valuable it is to remove a correlation from the circuit directly (by inserting a new long range gate using a series of SWAP gates) versus just trying to retrain some other part of the circuit to remove that correlation. If taking an approach using SWAP gates, it would be worth keeping track of which qubits are in contact during each part of the ansatz, allowing us to place particular interactions between qubits where they require fewer extra SWAP gates. This approach could also enable us to perform a sequential optimisation of circuits, whereby we disentangle qubits individually to create a smaller, more manageable problem. Indeed, since I made this investigation, some related ideas have been employed for shallow circuit learning [117].

6
Conclusion

A question I have enjoyed asking quantum computing researchers in the past is "Which do you think will happen first, fusion power to the grid, or a fault-tolerant quantum computer performing a useful calculation?" Towards the start of my doctorate, this question often produced a laugh and sometimes a wince, more recently, it seems to be almost unanimously believed that quantum computers will win this (arbitrary) race. The rate of hardware advancement makes it quite possible to believe that this milestone will be reached in the coming decade. Theoretical and algorithmic developments also advance rapidly, with many important developments taking place while I have been involved in the field. I often have to remind myself that some results that I consider to be 'old wisdom' because I became aware of them shortly after joining the field, in fact only slightly pre-date the start of my doctorate.

In this thesis, I have detailed the contributions I have made to the field of quantum algorithms, ranging from near-term hybrid variational algorithms, to the future where a larger number of qubits than strictly necessary are available to speed up computations.

Firstly, in Chapter 2, I introduced the CoVaR method, which leverages classical shadows to efficiently optimise variational circuits through root finding to produce eigenstates providing an entirely new way to optimise such circuits. I demonstrated

the theoretical foundations of CoVaR and provided numerical simulations to validate its effectiveness. The results showed that CoVaR can significantly improve the convergence rate compared to traditional variational methods, and has consequently seen use in follow-up works [172, 173], although it suffers from many of the problems plaguing variational algorithms in general. As a variational algorithm, the potential usefulness of CoVaR in the future remains uncertain, and may be limited to use-cases where warm starts for the optimisation are available [41], such as short-time Hamiltonian simulation, or for recompilation/optimisation of small circuits for use in near-term hardware. Due to the enhanced difficulty of initialising CoVaR optimisations, this already strongly overlaps with the area of likely utility.

Secondly, I explored the Shadow Subspace Expansion technique, which aims to estimate ground state energies using subspace expansions of unprecedented dimension. I presented a detailed analysis of the method, including its mathematical formulation and practical implementation. My simulations indicated that SSE can be a useful post-processing tool for estimation and error mitigation, and provided classical shadows are already available, requires no extra quantum cost. This technique is able to advance the estimation of accurate ground state energies as an error-mitigation-like protocol which is able to improve on (not only accurately measure) energy estimates from an output state, at the cost of a large number of shots. For this reason, the usefulness of this approach is likely to be restricted to the following scenarios: there is already some other reason to measure a large number of shadow snapshots, which is somewhat restrictive as if you wish to extract information about some restricted set of observables it is usually more efficient to measure them directly; no further resources exist to improve the output state of whatever algorithm precedes SSE, if it is possible it is likely to be more efficient to just improve this state; it is possible to obtain 10^7+ shots, either by fast measurement times or multiple processors running in parallel.

Thirdly, Chapter 4 focused on the parallelisation of the Linear Combination of Unitaries method. I developed an algorithm to arbitrarily parallelise LCU operations for the common scenario of the unitaries being Pauli operators, producing large

run-time improvements with a constant factor qubit overhead, and reducing the space-time cost of an often expensive subroutine. Reductions in space-time volume of the computation will be generally useful in the fault-tolerant regime, and as constructions for providing magic states have improved [237, 239], this parallelisation has become more advantageous. Of course, extra qubits must be available to make use of these ideas, restricting their application to quantum computers decades away, and likely beyond the point where quantum advantage has already been achieved. This work moves beyond parallelisation of geometrically local observables, although it is likely to be less useful for fermionic operators, due to specialised schemes for block encoding quantum chemistry Hamiltonians e.g. Tensor Hypercontraction [212]. One application of this work with significant utility is in the application to QROM data loading circuits. Parallelisation is also likely to be essential for certain hardware platforms, with neutral atom and ion trap platforms having cycle times ~ 1000 greater than solid state platforms, although some of this discrepancy can be recovered by faster decoding of transversal gate operations available with greater connectivity [255] (a factor of the code distance d which will be ~ 30 for many fault-tolerant algorithms). Fortunately the connectivity of these systems also assists in the routing of these parallelised algorithms which is a challenge not discussed in this work.

Finally, I discussed some ongoing work, with preliminary results on symmetries in adiabatic state preparation and entanglement-guided variational recompilation. These findings suggest promising directions for future research, in improving state preparation techniques and generating efficient recompiled quantum circuits for examples of small circuits where the optimisation is tractable.

In conclusion, this thesis has made contributions to the field of quantum computing across the spectrum of quantum algorithms. The proposed techniques have the potential to enhance the performance and scalability of quantum computation, bringing us closer to realizing practical quantum advantage. It is my sincere hope that at some stage, a development that I have made is run on a quantum machine for the advancement of science.

Appendices

Contents

A Root Finding using Orthogonal constraints

We presented the general theory of our approach in Section 2.3 whereby we compute covariances with respect to an arbitrary operator pool $\mathcal{P}$ in order to search for eigenstates of an arbitrary Hamiltonian $\mathcal{H}$. Our CoVAR approach, while allowing large operator pool, restricts the problem Hamiltonian and the operator pool to local Pauli strings. While Pauli strings are orthonormal in operator space, their actions on quantum states are generally not orthogonal directions in Hilbert space. In the present section we explore another kind of operator pool $\mathcal{P}$ whereby the operators represent orthogonal directions in state space. Let us first start by interpreting covariances as overlaps in Hilbert spaces.

A.1 Interpretation as state overlaps

The covariances can be interpreted as overlaps between quantum states as

$$\langle A, B\rangle_\psi = \langle \phi_A | \phi_B \rangle, \tag{1}$$

where we can define the (unnormalised) vectors $|\phi_A\rangle := (A - \langle A\rangle)|\psi\rangle$ for any operator $A \in \mathbb{C}^{d\times d}$ with norm $\|\phi_A\|^2 = \mathrm{Var}[A]$.

this is, when we compute overlaps $\langle O_k, \mathcal{H}\rangle_\psi = \langle \phi_k | \phi_\mathcal{H}\rangle$ we actually decompose the quantum state $\mathcal{H}|\psi\rangle$ into a set of quantum states $|\phi_l\rangle$ that we obtain by acting

on $|\psi\rangle$ with elements of our operator pool. Given that $|\phi_{\mathcal{H}}\rangle$ must be the null vector when the eigenvalue equation is satisfied, it is necessary that any overlap with this vector must vanish.

While Pauli strings form an orthonormal basis of **operator space**, they have the disadvantage that in **Hilbert space** they result in non-orthogonal actions $\langle\phi_k|\phi_l\rangle \neq \delta_{kl}$, i.e., we decompose the vector $\mathcal{H}|\psi\rangle$ into a non-orthogonal basis. On the other hand, it is possible to define an operator pool that corresponds to orthogonal directions in Hilbert space.

Let us consider strings of X Pauli operators as $X_k \in \{\mathrm{Id}, X\}^{\otimes N}$ using the binary index $k \in \{0,1\}^N$. We define the orthogonal operator pool $\mathcal{P} := \{O_k := UX_kU^\dagger, k \in \{0,1\}^N\}$ via the operators, where the unitary quantum circuit U maps our reference $|\psi\rangle = U|0\rangle$ onto our quantum state. The quantum states $|\phi_k\rangle := O_k|\psi\rangle$ form an orthonormal system $\langle\phi_k|\phi_l\rangle = \delta_{kl}$ and thus the operators O_k map to orthogonal directions in Hilbert space.

The orthonormality in Hilbert space follows can be seen by

$$\langle\psi|O_kO_l|\psi\rangle = \langle 0|U^\dagger UX_kU^\dagger UX_lU^\dagger U|\underline{0}\rangle = \langle k|l\rangle,$$

where $|k\rangle$ are standard basis vectors with $\langle k|l\rangle = \delta_{kl}$.

The resulting covariances can actually be shown to be entries in a column vector of the Hamiltonian matrix. In particular, given the states $|\phi_k\rangle$ form an orthonormal basis they can be used to represent the Hamiltonian matrix as the covariances $f_k = \langle\psi|O_k\mathcal{H}|\psi\rangle = \langle\phi_k|\mathcal{H}|\phi_0\rangle = \mathrm{Col}_0(\mathcal{H})$, which are then actually elements of the first column vector of the problem Hamiltonian.

Importantly, while this operator pool has the advantage that the covariances represent independent, orthogonal directions in state space it is clear that we would generally need to compute all $2^N - 1$ of these orthogonal constraints, as elements of the first column of the Hamiltonian matrix, in order to be able to compute the variance and thus to verify that the quantum state $|\psi\rangle$ is an eigenstate of the Hamiltonian. In stark contrast for a decomposition of a Hamiltonian into an operator basis $\mathcal{H} = \sum_{k=1}^{r} h_a\mathcal{H}_a$ which typically grows polynomially with the system

size, it suffices to only compute the corresponding polynomial number of covariances. Although these operators, such as Pauli strings, are orthonormal in operator space, they do not correspond to orthogonal directions in Hilbert space.

A.2 Finding eigenstates via orthogonal operator pools

Let us now apply our orthogonal constraints to finding eigenstates by finding roots. An advantage of this scheme is that we can randomly sample the constraints according to an importance sampling, i.e., the constraints are picked with a probability proportional to their magnitude $p_k = |f_k|^2$. We can efficiently upper bound these probabilities in an experiment as $p_k = \langle \phi_k | \phi_{\mathcal{H}} \rangle \leq |\langle k | U^\dagger \mathcal{H} | \psi \rangle|^2$. In particular, we run the quantum circuit $U^\dagger \mathcal{H}_a U$ and measure samples in the standard basis, whereby we obtain the binary string k with probability $|\langle k | U^\dagger \mathcal{H}_a | \psi \rangle|^2$. It then follows that $p_k \leq \sum_a c_a |\langle k | U^\dagger \mathcal{H}_a | \psi \rangle|^2$. There is of course no guarantee that these probabilities have structure, however, when performing energy minimisation first, the probabilities are more likely to be peaked around the lower energy basis vectors $|\phi_k\rangle$. In case of finding eigenstates of a set of mutually commuting Hamiltonians $\mathcal{H}_a$ we can compute covariances as $\langle \phi_k | \mathcal{H}_a | \psi \rangle$ individually for all operators $\mathcal{H}_a$. If all such variances vanish then we are guaranteed that $\mathrm{Var}[\mathcal{H}_a] = 0$ for all a. In Figure A.1 we compare the performance of root finding for two choices of operator pool on a parameter rediscovery problem (3-local Pauli strings and the orthogonal operator pool), showing that both pools give very similar performance.

B Details of Numerical simulations of CoVaR

Data were obtained from statevector simulations performed by the open-source tools QuEST [256] and QuESTlink [257]. Shot noise was simulated by adding Gaussian noise of standard deviation $1/\sqrt{N_s}$ to computed matrix elements. Numerics for recompilation and the spin chain were both performed using a hardware efficient ansatz of the form in Fig. B.1.

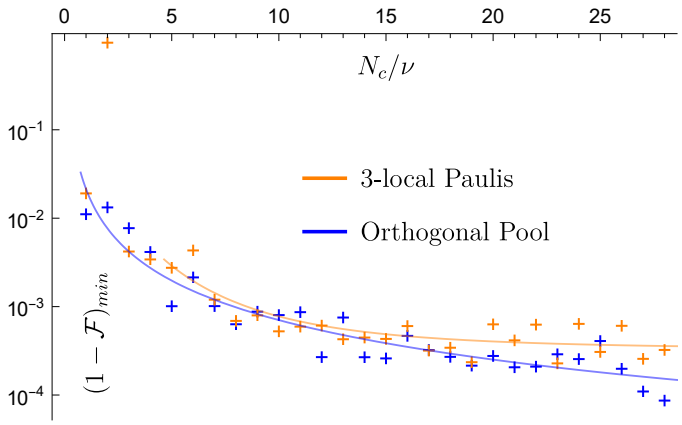


Figure A.1: Performance of recompiling the parameters of a 2 layer unitary on 14 qubits. Plot shows the minimum infidelity reached over 3 runs of 20 iterations of root finding with the ratio of constraints to ansatz parameters ν, for two different choices of operator pool. Blue for orthogonal pool and orange for the set of all 3-local Pauli strings. Fits shown are of the form $1 - \mathcal{F}_{min} = a\,(N_c/\nu)^{-b} + c$.

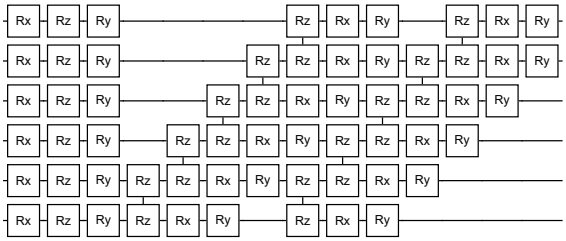


Figure B.1: One layer of the Hardware Efficient Ansatz used in numerics for 6 qubits

B.1 Computing the update rule classically

We need to first stack real and imaginary parts on top to solve for only real parameter updates. In particular, we want to solve the linear system of equations $\mathbf{J}\Delta\underline{\theta} = \underline{f}$. Here we estimate both $\mathbf{J}$ and $\underline{f}$ with a quantum computer and we want to compute the parameter update $\Delta\underline{\theta}$. If we merely compute the pseudoinverse of $\mathbf{J}$ and apply it as $\mathbf{J}^{-1}\underline{f}$ then the resulting vector is generally complex $\Delta\underline{\theta}$. However, we demand that our parameter-update is real as $\Delta\underline{\theta} \in \mathbb{R}$.

For this reason we enforce that the vector is real $\Delta\underline{\theta} \in \mathbb{R}$ via the following set of linear equations

$$\mathrm{Re}[\mathbf{J}]\Delta\underline{\theta} = \mathrm{Re}[\underline{f}], \qquad \text{and} \qquad \mathrm{Im}[\mathbf{J}]\Delta\underline{\theta} = \mathrm{Im}[\underline{f}].$$

We can simultaneously solve the above systems of equations by stacking real and imaginary parts $\tilde{\mathbf{J}} = (\mathrm{Re}[\mathbf{J}], \mathrm{Im}[\mathbf{J}])^T$ and $\tilde{\underline{f}} = (\mathrm{Re}[\underline{f}], \mathrm{Im}[\underline{f}])^T$ on top of each other

and this will guarantee that the solution is real $\tilde{\mathbf{J}}^{-1}\underline{\tilde{f}} \in \mathbb{R}$ since $\tilde{\mathbf{J}}$ and $\underline{\tilde{f}}$ are real matrices and vectors, respectively.

I invert the Jacobian via the usual damped, regularised inverse

$$\tilde{\mathbf{J}}^{-1} := [\tilde{\mathbf{J}}^{\intercal}\tilde{\mathbf{J}} + \lambda \mathrm{Id}]^{-1}\tilde{\mathbf{J}}^{\intercal}.$$

Here λ is a regularisation parameter that I dynamically set by choosing $\lambda = 0.0001 \times 2^i$ where i is incremented from 0 until the condition $\|\underline{f}(\underline{\theta}_t)\| < \|\underline{f}(\underline{\theta}_{t-1})\|$ is met, i.e., the norm of the covariance vector has decreased from the previous iteration. If the step $\Delta\underline{\theta}$ to be taken is too large for any individual parameter is too large $|\Delta\theta_i| > 1$, the update step is rescaled to normalise this value to 1, to prevent the algorithm from taking overly large steps. For the selection of constraints, N_c constraints are selected randomly from the chosen operator pool at every iteration. I implemented a line search algorithm whereby I compute the update rule $\Delta\underline{\theta}$ and compute the value of the vector norm $\|\underline{f}(\underline{\theta})\|$ at parameter values $\underline{\theta} = \underline{\theta}_t + \kappa\Delta\underline{\theta}$ in small increments in κ. However, it was observed empirically that a step close to the canonical choice $\kappa = -1$ was almost always chosen, so line search was not used in the numerics.

B.2 Effect of Constraint Number on Performance

Data for Fig. 2.2 was obtained by simulating a 14 qubit parameter rediscovery problem using 2 layers of the ansatz in Fig. B.1. The initial states were initialised close to the solution by randomly perturbing the solution parameters (an initial average fidelity of $\mathcal{F} = 46 \pm 7\%$). The fits in this figure are of the form $1 - \mathcal{F}_{min} = a\,(N_c/\nu)^{-b} + c$ and have constants $a = 0.0259, b = 1.68, c = 10^{-4}$ for the case with noise and $a = 5.62, b = 3.23, c = 0$ for the noise free case. The orange line in Fig. A.1 is identical to the orange line in Fig. 2.2 and the orthogonal operator pool plot has $a = 0.021, b = 1.48, c = 0$.

B.3 Recompilation

The numerics for recompilation in Fig. 2.4(c) were done on a 10 qubit parameter rediscovery problem for two layers of HEA ($\nu = 88$). Gradient descent was done with learning rate $\eta = 0.1$ for both VQE and V-VQE

B.4 Spin chain

For the spin chain simulations in Fig. 2.5, Imaginary Time Evolution was used from a random initialisation until an energy of $E = -5.9$ was reached, with parameters $\underline{\theta}_{imag}$. These parameters disturbed by $|\Delta\theta_k| \leq 0.05$ to produce 7 low energy states which root finding was then run for 40 iterations from. Imaginary Time Evolution was also continued from $\underline{\theta}_{imag}$ until convergence and reached an energy difference to the ground state of $\Delta E = 0.012$ compared to the 4×10^{-4} of CoVaR. The Hamiltonian used $J = 0.1$ and c_i chosen randomly between -1 and 1.

C Dependence of eigenvalues of Subspace Matrix $\boldsymbol{S}$ on K

Here we show the dependence of the largest eigenvalue of the matrix $\boldsymbol{S}$ on the subspace dimension K Fig. C.1, demonstrating that the inclusion of more subspace dimensions does increase the amount of information present about the most significant directions within the subspace. However, as this increase is linear, we are not able to significantly increase the number of eigenvalues above the noise floor shown in Fig. 3.1d by increasing K.

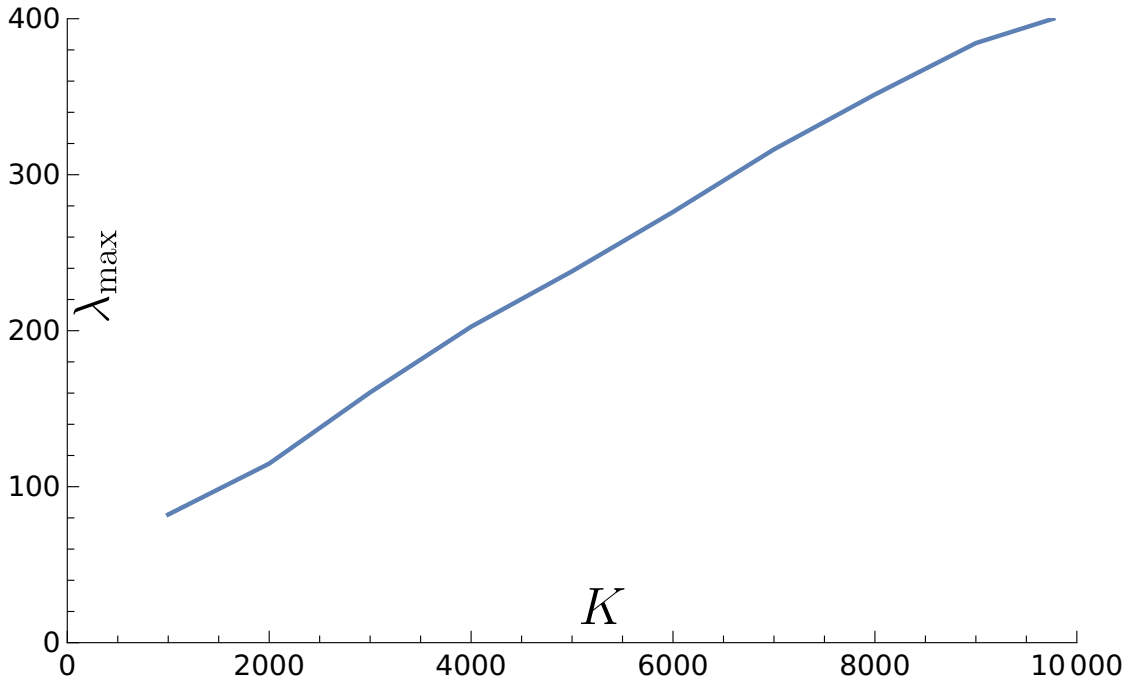


Figure C.1: Demonstration of the largest eigenvalue of $\boldsymbol{S}$ increasing linearly with K, as is expected for a matrix of overlaps $\boldsymbol{S}_{ij} = \langle \psi_i \rangle \psi_j$, where the states are mostly similar to each other (consisting of only small local changes). For example in the extreme case where we take all $G_i = I$, then the largest eigenvalue is K.

D Filling factor for QROM circuits

For the case of parallelising QROM circuits, the operators being applied are all tensor products of Pauli X operators, so already commute with each other. The linear independence condition becomes equivalent to the linear independence of binary vectors, which is easy to satisfy as demonstrated in Fig. D.1 where I plot $\mathcal{F}$ found by a greedy search over all 2^n bit strings for varying n.

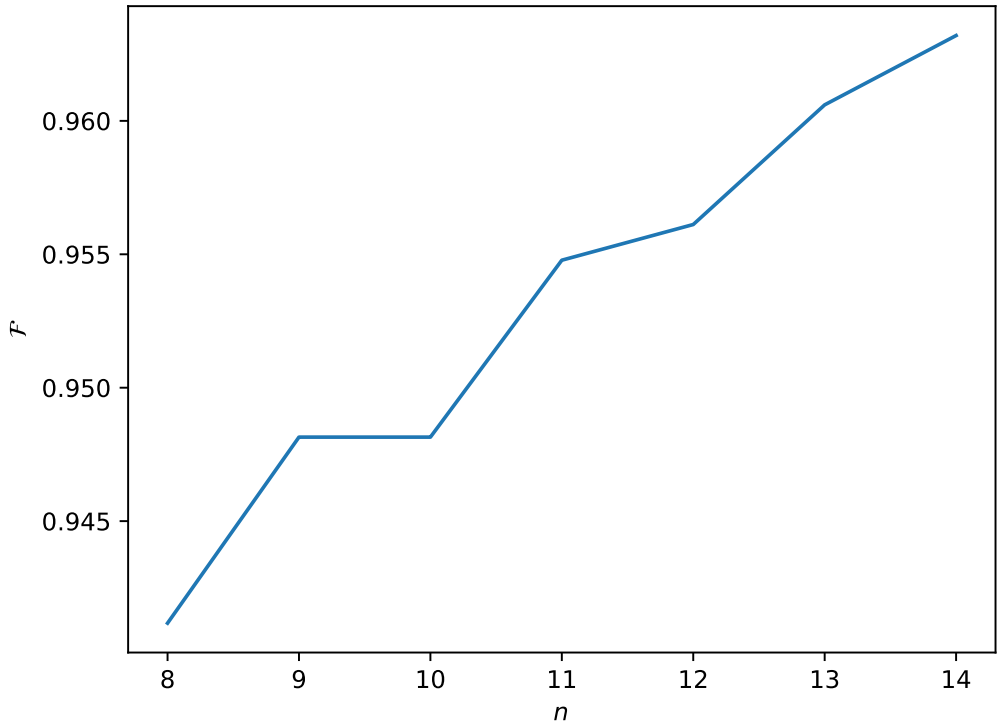


Figure D.1: Filling factor for parallelisation of QROM circuits, using Hamiltonians containing all tensor products of Pauli X's.

E LCU Parallel Implemenation Diagram

Fig. E.1 shows the overall scheme of the LCU parallelisation algorithm in the case where the ancilla fanout replacement of Section 4.4.2 is not used, following all the preprocessing and circuits required for the parallelisation of SELECT through identifying required commuting and linearly independent sets of Paulis, their associated Clifford transformations, and the compilation of those into constant depth circuits.

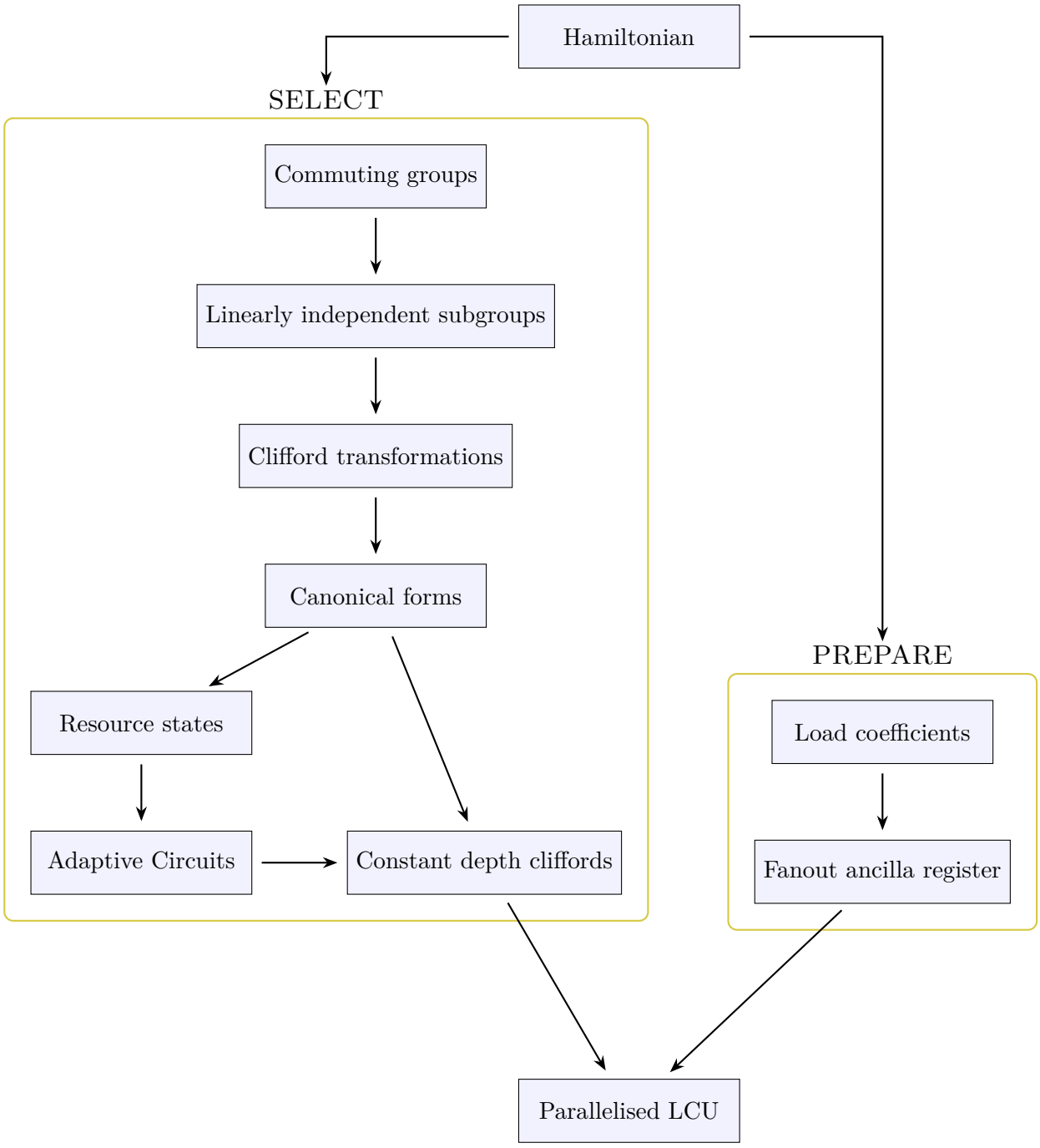


Figure E.1: Algorithm diagram for LCU parallelisation, showing the preprocessing steps required for SELECT as well as the primitives used for paralellisation on both PREPARE and SELECT registers.

References

[1] Richard P. Feynman. “Simulating Physics with Computers”. In: *International Journal of Theoretical Physics* 21 (June 1982), pp. 467–488.

[2] John Preskill. “Quantum Computing in the NISQ Era and Beyond”. In: *Quantum* 2 (Aug. 2018), p. 79.

[3] Paul Adrien Maurice Dirac and Ralph Howard Fowler. “Quantum Mechanics of Many-Electron Systems”. In: *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* 123 (Apr. 1929), pp. 714–733.

[4] Paul Benioff. “The Computer as a Physical System: A Microscopic Quantum Mechanical Hamiltonian Model of Computers as Represented by Turing Machines”. In: *Journal of Statistical Physics* 22 (May 1980), pp. 563–591.

[5] David Deutsch and Roger Penrose. “Quantum Theory, the Church–Turing Principle and the Universal Quantum Computer”. In: *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* 400 (July 1985), pp. 97–117.

[6] David Deutsch and Richard Jozsa. “Rapid Solution of Problems by Quantum Computation”. In: *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences* 439 (Dec. 1992), pp. 553–558.

[7] Peter W. Shor. “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”. In: *SIAM Journal on Computing* 26 (Oct. 1997), pp. 1484–1509.

[8] Lov K. Grover. “A Fast Quantum Mechanical Algorithm for Database Search”. In: *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing - STOC ’96*. Philadelphia, Pennsylvania, United States: ACM Press, 1996, pp. 212–219.

[9] H D Zeh. “On the Interpretation of Measurement in Quantum Theory”. In: *Frontiers of Physics* (1970), p. 8.

[10] Maximilian Schlosshauer. “Quantum Decoherence”. In: *Physics Reports* 831 (Oct. 2019), pp. 1–57.

[11] W. G. Unruh. “Maintaining Coherence in Quantum Computers”. In: *Physical Review A* 51 (Feb. 1995), pp. 992–997.

[12] Peter W. Shor. “Scheme for Reducing Decoherence in Quantum Computer Memory”. In: *Physical Review A* 52 (Oct. 1995), R2493–R2496.

[13] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. 10th anniversary ed. Cambridge ; New York: Cambridge University Press, 2010.

[14] A. Yu. Kitaev. "Fault-Tolerant Quantum Computation by Anyons". In: *Annals of Physics* 303 (Jan. 2003), pp. 2–30.

[15] S. B. Bravyi and A. Yu Kitaev. *Quantum Codes on a Lattice with Boundary*. Nov. 1998. arXiv: quant-ph/9811052.

[16] Jonathan A. Jones, Michele Mosca, and Rasmus H. Hansen. "Implementation of a Quantum Search Algorithm on a Quantum Computer". In: *Nature* 393 (May 1998), pp. 344–346.

[17] Thomas Monz et al. "14-Qubit Entanglement: Creation and Coherence". In: *Physical Review Letters* 106 (Mar. 2011), p. 130506. arXiv: 1009.6126 [quant-ph].

[18] R. Barends et al. "Superconducting Quantum Circuits at the Surface Code Threshold for Fault Tolerance". In: *Nature* 508 (Apr. 2014), pp. 500–503.

[19] Frank Arute, Kunal Arya, Ryan Babbush, et al. "Quantum Supremacy Using a Programmable Superconducting Processor". In: *Nature* 574 (Oct. 2019), pp. 505–510.

[20] Han-Sen Zhong et al. "Phase-Programmable Gaussian Boson Sampling Using Stimulated Squeezed Light". In: *Physical Review Letters* 127 (Oct. 2021), p. 180502.

[21] Yulin Wu et al. "Strong Quantum Computational Advantage Using a Superconducting Quantum Processor". In: *Physical Review Letters* 127 (Oct. 2021), p. 180501.

[22] Sepehr Ebadi et al. "Quantum Phases of Matter on a 256-Atom Programmable Quantum Simulator". In: *Nature* 595 (July 2021), pp. 227–232.

[23] Rajeev Acharya et al. "Suppressing Quantum Errors by Scaling a Surface Code Logical Qubit". In: *Nature* 614 (Feb. 2023), pp. 676–681.

[24] M. P. da Silva et al. *Demonstration of Logical Qubits and Repeated Error Correction with Better-than-Physical Error Rates*. Apr. 2024. arXiv: 2404.02280.

[25] Dolev Bluvstein et al. "Logical Quantum Processor Based on Reconfigurable Atom Arrays". In: *Nature* 626 (Feb. 2024), pp. 58–65.

[26] Noah Berthusen et al. *Experiments with the 4D Surface Code on a QCCD Quantum Computer*. Aug. 2024. arXiv: 2408.08865 [quant-ph].

[27] Riddhi S. Gupta et al. "Encoding a Magic State with beyond Break-Even Fidelity". In: *Nature* 625 (Jan. 2024), pp. 259–263.

[28] C. Ryan-Anderson et al. "High-Fidelity Teleportation of a Logical Qubit Using Transversal Gates and Lattice Surgery". In: *Science* 385 (Sept. 2024), pp. 1327–1331.

[29] Colin P. Williams. "Quantum Gates". In: *Explorations in Quantum Computing*. Ed. by Colin P. Williams. Texts in Computer Science. London: Springer, 2011, pp. 51–122.

[30] Harry Buhrman et al. *State Preparation by Shallow Circuits Using Feed Forward*. July 2023. arXiv: 2307.14840 [quant-ph].

[31] Michael Foss-Feig et al. *Experimental Demonstration of the Advantage of Adaptive Quantum Circuits*. Feb. 2023. arXiv: 2302.03029 [cond-mat, physics:quant-ph].

[32] Kevin C. Smith et al. *Constant-Depth Preparation of Matrix Product States with Adaptive Quantum Circuits*. Apr. 2024. arXiv: 2404.16083 [quant-ph].

[33] Charles H. Bennett et al. "Teleporting an Unknown Quantum State via Dual Classical and Einstein-Podolsky-Rosen Channels". In: *Physical Review Letters* 70 (Mar. 1993), pp. 1895–1899.

[34] Scott Aaronson and Daniel Gottesman. "Improved Simulation of Stabilizer Circuits". In: *Physical Review A* 70 (Nov. 2004), p. 052328.

[35] Bryan Eastin and Emanuel Knill. "Restrictions on Transversal Encoded Quantum Gate Sets". In: *Physical Review Letters* 102 (Mar. 2009), p. 110502.

[36] Sergey Bravyi and Alexei Kitaev. "Universal Quantum Computation with Ideal Clifford Gates and Noisy Ancillas". In: *Physical Review A* 71 (Feb. 2005), p. 022316.

[37] John Preskill. "Quantum Computing in the NISQ Era and Beyond". In: *Quantum* 2 (Aug. 2018), p. 79.

[38] M. Cerezo et al. "Cost Function Dependent Barren Plateaus in Shallow Parametrized Quantum Circuits". In: *Nature Communications* 12 (Dec. 2021), p. 1791. arXiv: 2001.00550.

[39] Carlos Ortiz Marrero, Mária Kieferová, and Nathan Wiebe. "Entanglement-Induced Barren Plateaus". In: *PRX Quantum* 2 (Oct. 2021), p. 040316.

[40] Jarrod R. McClean et al. "Barren Plateaus in Quantum Neural Network Training Landscapes". In: *Nature Communications* 9 (Nov. 2018), pp. 1–6.

[41] M. Cerezo et al. *Does Provable Absence of Barren Plateaus Imply Classical Simulability? Or, Why We Need to Rethink Variational Quantum Computing*. Mar. 2024. arXiv: 2312.09121.

[42] Samson Wang et al. "Noise-Induced Barren Plateaus in Variational Quantum Algorithms". In: *Nature Communications* 12 (Dec. 2021), p. 6961. arXiv: 2007.14384 [quant-ph].

[43] Manuel S. Rudolph et al. *Classical Surrogate Simulation of Quantum Systems with LOWESA*. Aug. 2023. arXiv: 2308.09109.

[44] Matthew L. Goh et al. *Lie-Algebraic Classical Simulations for Variational Quantum Computing*. Aug. 2023. arXiv: 2308.01432.

[45] Laura Clinton et al. "Phase Estimation of Local Hamiltonians on NISQ Hardware". In: *arXiv:2110.13584 [quant-ph]* (Oct. 2021). arXiv: 2110.13584 [quant-ph].

[46] Alberto Peruzzo et al. "A Variational Eigenvalue Solver on a Photonic Quantum Processor". In: *Nature Communications* 5 (July 2014), p. 4213.

[47] Guang Hao Low and Isaac L. Chuang. "Hamiltonian Simulation by Qubitization". In: *Quantum* 3 (July 2019), p. 163.

[48] Ophelia Crawford et al. "Efficient Quantum Measurement of Pauli Operators in the Presence of Finite Sampling Error". In: *Quantum* 5 (Jan. 2021), p. 385. arXiv: 1908.06942.

[49] Andreas Elben et al. "The Randomized Measurement Toolbox". In: *arXiv:2203.11374 [quant-ph]* (Mar. 2022). arXiv: 2203.11374 [quant-ph].

[50] Raja Selvarajan, Manas Sajjan, and Sabre Kais. "Variational Quantum Circuits to Prepare Low Energy Symmetry States". In: *arXiv:2112.12857 [quant-ph]* (Dec. 2021). arXiv: 2112.12857 [quant-ph].

[51] Johannes Jakob Meyer et al. *Exploiting Symmetry in Variational Quantum Machine Learning*. May 2022. arXiv: 2205.06217 [quant-ph].

[52] Harry Buhrman et al. "Quantum Fingerprinting". In: *Physical Review Letters* 87 (Sept. 2001), p. 167902.

[53] Yuxuan Du et al. "An Efficient Measure for the Expressivity of Variational Quantum Algorithms". In: *arXiv:2104.09961 [quant-ph]* (Apr. 2021). arXiv: 2104.09961 [quant-ph].

[54] John Napp. "Quantifying the Barren Plateau Phenomenon for a Model of Unstructured Variational Ans\ätze". In: *arXiv:2203.06174 [quant-ph]* (Mar. 2022). arXiv: 2203.06174 [quant-ph].

[55] Zoë Holmes et al. "Connecting Ansatz Expressibility to Gradient Magnitudes and Barren Plateaus". In: *arXiv:2101.02138 [quant-ph, stat]* (Jan. 2021). arXiv: 2101.02138 [quant-ph, stat].

[56] Abhinav Kandala et al. "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets". In: *Nature* 549 (2017), p. 242.

[57] Andrew G. Taube and Rodney J. Bartlett. "New Perspectives on Unitary Coupled-Cluster Theory". In: *International Journal of Quantum Chemistry* 106 (2006), pp. 3393–3401.

[58] Dave Wecker, Matthew B. Hastings, and Matthias Troyer. "Progress towards Practical Quantum Variational Algorithms". In: *Physical Review A* 92 (Oct. 2015), p. 042303.

[59] Dominic W. Berry et al. "Efficient Quantum Algorithms for Simulating Sparse Hamiltonians". In: *Communications in Mathematical Physics* 270 (Mar. 2007), pp. 359–371.

[60] Roeland Wiersema et al. "Exploring Entanglement and Optimization within the Hamiltonian Variational Ansatz". In: *PRX Quantum* 1 (Dec. 2020), p. 020319. arXiv: 2008.02941.

[61] Martin Larocca et al. "Theory of Overparametrization in Quantum Neural Networks". In: *arXiv:2109.11676 [quant-ph, stat]* (Sept. 2021). arXiv: 2109.11676 [quant-ph, stat].

[62] Jonathan Romero et al. "Strategies for Quantum Computing Molecular Energies Using the Unitary Coupled Cluster Ansatz". In: *Quantum Science and Technology* 4 (Oct. 2018), p. 014008.

[63] Maria Schuld et al. "Evaluating Analytic Gradients on Quantum Hardware". In: *Physical Review A* 99 (Mar. 2019), p. 032331.

[64] David Wierichs et al. “General Parameter-Shift Rules for Quantum Gradients”. In: *Quantum* 6 (Mar. 2022), p. 677.

[65] Andrea Mari, Thomas R. Bromley, and Nathan Killoran. “Estimating the Gradient and Higher-Order Derivatives on Quantum Hardware”. In: *Physical Review A* 103 (Jan. 2021), p. 012405. arXiv: 2008.06517 [quant-ph].

[66] James Stokes et al. “Quantum Natural Gradient”. In: *Quantum* 4 (May 2020), p. 269. arXiv: 1909.02108.

[67] Sam McArdle et al. “Variational Ansatz-Based Quantum Simulation of Imaginary Time Evolution”. In: *npj Quantum Information* 5 (Dec. 2019), p. 75. arXiv: 1804.03023.

[68] Bálint Koczor and Simon C. Benjamin. “Quantum Natural Gradient Generalised to Non-Unitary Circuits”. In: *arXiv:1912.08660 [quant-ph]* (Dec. 2020). arXiv: 1912.08660 [quant-ph].

[69] Jacob Biamonte. “Universal Variational Quantum Computation”. In: *Physical Review A* 103 (Mar. 2021), p. L030401. arXiv: 1903.04500 [quant-ph].

[70] Markus Reiher et al. “Elucidating Reaction Mechanisms on Quantum Computers”. In: *Proceedings of the National Academy of Sciences* 114 (July 2017), pp. 7555–7560.

[71] P. Jordan and E. Wigner. “Über das Paulische Äquivalenzverbot”. In: *Zeitschrift für Physik* 47 (Sept. 1928), pp. 631–651.

[72] Sergey B. Bravyi and Alexei Yu. Kitaev. “Fermionic Quantum Computation”. In: *Annals of Physics* 298 (May 2002), pp. 210–226.

[73] Charles Derby et al. “Compact Fermion to Qubit Mappings”. In: *Physical Review B* 104 (July 2021), p. 035118.

[74] William Kirby et al. “Second-Quantized Fermionic Operators with Polylogarithmic Qubit and Gate Complexity”. In: *arXiv:2109.14465 [quant-ph]* (Mar. 2022). arXiv: 2109.14465 [quant-ph].

[75] Suguru Endo et al. “Variational Quantum Algorithms for Discovering Hamiltonian Spectra”. In: *Physical Review A* 99 (June 2019), p. 062304. arXiv: 1806.05707.

[76] Rahul Nandkishore and David A. Huse. “Many-Body Localization and Thermalization in Quantum Statistical Mechanics”. In: *Annual Review of Condensed Matter Physics* 6 (2015), pp. 15–38.

[77] Sumeet Khatri et al. “Quantum-Assisted Quantum Compiling”. In: *Quantum* 3 (May 2019), p. 140. arXiv: 1807.00800.

[78] Sukin Sim et al. “Adaptive Pruning-Based Optimization of Parameterized Quantum Circuits”. In: 6 (Mar. 2021), p. 025019.

[79] M. Bilkis et al. “A Semi-Agnostic Ansatz with Variable Structure for Quantum Machine Learning”. In: *arXiv:2103.06712 [quant-ph, stat]* (Mar. 2021). arXiv: 2103.06712 [quant-ph, stat].

[80] Zhimin He et al. “Variational Quantum Compiling with Double Q-learning”. In: *New Journal of Physics* 23 (Mar. 2021), p. 033002. arXiv: 2103.11611.

[81] Tyson Jones and Simon C. Benjamin. "Quantum Compilation and Circuit Optimisation via Energy Dissipation". In: *arXiv:1811.03147 [quant-ph]* (May 2020). arXiv: 1811.03147 [quant-ph].

[82] Matthias C. Caro et al. "Out-of-Distribution Generalization for Learning Quantum Dynamics". In: *arXiv:2204.10268 [quant-ph, stat]* (Apr. 2022). arXiv: 2204.10268 [quant-ph, stat].

[83] Matthias C. Caro et al. "Generalization in Quantum Machine Learning from Few Training Data". In: *arXiv:2111.05292 [quant-ph, stat]* (Nov. 2021). arXiv: 2111.05292 [quant-ph, stat].

[84] Edward Farhi and Hartmut Neven. *Classification with Quantum Neural Networks on Near Term Processors*. Aug. 2018. arXiv: 1802.06002 [quant-ph].

[85] K. Mitarai et al. "Quantum Circuit Learning". In: *Physical Review A* 98 (Sept. 2018), p. 032309.

[86] Long Hin Li, Dan-Bo Zhang, and Z. D. Wang. "Quantum Kernels with Gaussian State Encoding for Machine Learning". In: *Physics Letters A* 436 (June 2022), p. 128088.

[87] Manuela Weigold et al. "Expanding Data Encoding Patterns For Quantum Algorithms". In: *2021 IEEE 18th International Conference on Software Architecture Companion (ICSA-C)*. Stuttgart, Germany: IEEE, Mar. 2021, pp. 95–101.

[88] Hiroshi Yano et al. "Efficient Discrete Feature Encoding for Variational Quantum Classifier". In: *IEEE Transactions on Quantum Engineering* 2 (2021), pp. 1–14. arXiv: 2005.14382 [quant-ph].

[89] Scott Aaronson. "Shadow Tomography of Quantum States". In: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2018. New York, NY, USA: Association for Computing Machinery, June 2018, pp. 325–338.

[90] Hsin-Yuan Huang, Richard Kueng, and John Preskill. "Predicting Many Properties of a Quantum System from Very Few Measurements". In: *Nature Physics* 16 (Oct. 2020), pp. 1050–1057.

[91] Christian Bertoni et al. *Shallow Shadows: Expectation Estimation Using Low-Depth Random Clifford Circuits*. arXiv:2209.12924 [cond-mat, physics:quant-ph]. Sept. 2022.

[92] Matteo Ippoliti et al. "Operator Relaxation and the Optimal Depth of Classical Shadows". In: *Physical Review Letters* 130 (June 2023), p. 230403.

[93] Hong-Ye Hu, Soonwon Choi, and Yi-Zhuang You. "Classical Shadow Tomography with Locally Scrambled Quantum Dynamics". In: *Physical Review Research* 5 (Apr. 2023), p. 023027.

[94] Kianna Wan et al. "Matchgate Shadows for Fermionic Quantum Simulation". In: *Commun. Math. Phys.* 404 (Dec. 2023), pp. 629–700.

[95] Bujiao Wu and Dax Enshan Koh. "Error-Mitigated Fermionic Classical Shadows on Noisy Quantum Devices". In: *npj Quantum Information* 10 (Apr. 2024), pp. 1–8.

[96] Richard Jozsa and Akimasa Miyake. *Matchgates and Classical Simulation of Quantum Circuits*. Nov. 2008. arXiv: 0804.4050.

[97] Matteo Ippoliti. *Classical Shadows Based on Locally-Entangled Measurements*. arXiv:2305.10723 [cond-mat, physics:quant-ph]. Nov. 2023.

[98] Maxwell West et al. *Real Classical Shadows*. Oct. 2024. arXiv: 2410.23481 [quant-ph].

[99] Hsin-Yuan Huang, Richard Kueng, and John Preskill. "Efficient Estimation of Pauli Observables by Derandomization". In: *Phys. Rev. Lett.* 127 (July 2021), p. 030503.

[100] Charles Hadfield et al. "Measurements of Quantum Hamiltonians with Locally-Biased Classical Shadows". In: *Communications in Mathematical Physics* 391 (May 2022), pp. 951–967.

[101] Charles Hadfield. *Adaptive Pauli Shadows for Energy Estimation*. May 2021. arXiv: 2105.12207 [quant-ph].

[102] Zhenyu Cai et al. *Biased Estimator Channels for Classical Shadows*. arXiv:2402.09511 [quant-ph]. Feb. 2024.

[103] Nayuta Takemori et al. *Balancing Error Budget for Fermionic K-RDM Estimation*. Dec. 2023. arXiv: 2312.17452 [cond-mat, physics:quant-ph].

[104] Andrea Caprotti, Joshua Morris, and Borivoje Dakić. *Optimising Quantum Tomography via Shadow Inversion*. May 2024. arXiv: 2402.06727 [quant-ph].

[105] Joonas Malmi et al. "Enhanced Observable Estimation through Classical Optimization of Informationally Over-Complete Measurement Data – beyond Classical Shadows". In: *Physical Review A* 109 (June 2024), p. 062412. arXiv: 2401.18049 [quant-ph].

[106] Laurin E. Fischer et al. "Dual Frame Optimization for Informationally Complete Quantum Measurements". In: *Physical Review A* 109 (June 2024), p. 062415. arXiv: 2401.18071 [quant-ph].

[107] Senrui Chen et al. "Robust Shadow Estimation". In: *PRX Quantum* 2 (Sept. 2021), p. 030348.

[108] Dax Enshan Koh and Sabee Grewal. "Classical Shadows With Noise". In: *Quantum* 6 (Aug. 2022), p. 776.

[109] Hong-Ye Hu et al. *Logical Shadow Tomography: Efficient Estimation of Error-Mitigated Observables*. arXiv:2203.07263 [quant-ph]. Mar. 2022.

[110] Hamza Jnane et al. "Quantum Error Mitigated Classical Shadows". In: *PRX Quantum* 5 (Feb. 2024), p. 010324.

[111] Nobuyuki Yoshioka et al. "Generalized Quantum Subspace Expansion". In: *Phys. Rev. Lett.* 129 (July 2022), p. 020502.

[112] Bo Yang et al. *Dual-GSE: Resource-efficient Generalized Quantum Subspace Expansion*. Sept. 2023. arXiv: 2309.14171 [quant-ph].

[113] Hans Hon Sang Chan et al. *Algorithmic Shadow Spectroscopy*. arXiv:2212.11036 [quant-ph]. Aug. 2023.

[114] Amira Abbas et al. "On Quantum Backpropagation, Information Reuse, and Cheating Measurement Collapse". In: *Advances in Neural Information Processing Systems* 36 (Dec. 2023), pp. 44792–44819.

[115] Robbie King et al. *Triply Efficient Shadow Tomography*. arXiv:2404.19211 [quant-ph]. Apr. 2024.

[116] Hsin-Yuan Huang, Sitan Chen, and John Preskill. *Learning to Predict Arbitrary Quantum Processes*. Oct. 2022. arXiv: 2210.14894 [quant-ph].

[117] Hsin-Yuan Huang et al. "Learning Shallow Quantum Circuits". In: *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*. June 2024, pp. 1343–1351. arXiv: 2401.10095 [quant-ph].

[118] Alexander M. Dalzell et al. *Quantum Algorithms: A Survey of Applications and End-to-End Complexities*. Oct. 2023. arXiv: 2310.03011 [quant-ph].

[119] P.W. Shor. "Algorithms for Quantum Computation: Discrete Logarithms and Factoring". In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. Nov. 1994, pp. 124–134.

[120] Chris Cade et al. "Improved Hardness Results for the Guided Local Hamiltonian Problem". In: *LIPIcs, Volume 261, ICALP 2023* 261 (2023), 32:1–32:19. arXiv: 2207.10250 [quant-ph].

[121] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. 10th. New York, NY, USA: Cambridge University Press, 2011.

[122] Thomas E O'Brien, Brian Tarasinski, and Barbara M Terhal. "Quantum Phase Estimation of Multiple Eigenvalues for Small-Scale (Noisy) Experiments". In: *New Journal of Physics* 21 (Feb. 2019), p. 023022.

[123] Laura Clinton et al. *Quantum Phase Estimation without Controlled Unitaries*. Oct. 2024. arXiv: 2410.21517 [quant-ph].

[124] Gilles Brassard et al. *Quantum Amplitude Amplification and Estimation*. May 2000. arXiv: quant-ph/0005055.

[125] Dominic W. Berry et al. "Exponential Improvement in Precision for Simulating Sparse Hamiltonians". In: *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*. May 2014, pp. 283–292. arXiv: 1312.1414 [quant-ph].

[126] Theodore J. Yoder, Guang Hao Low, and Isaac L. Chuang. *Fixed-Point Quantum Search with an Optimal Number of Queries*. Nov. 2014. arXiv: 1409.3305.

[127] Patrick Rall and Bryce Fuller. "Amplitude Estimation from Quantum Signal Processing". In: (July 2022).

[128] Aram W. Harrow and Annie Y. Wei. *Adaptive Quantum Simulated Annealing for Bayesian Inference and Estimating Partition Functions*. Feb. 2020. arXiv: 1907.09965.

[129] Arjan Cornelissen and Yassine Hamoudi. *A Sublinear-Time Quantum Algorithm for Approximating Partition Functions*. Jan. 2023. arXiv: 2207.08643.

[130] Sophia Simon et al. *Amplified Amplitude Estimation: Exploiting Prior Knowledge to Improve Estimates of Expectation Values*. Mar. 2024. arXiv: 2402.14791 [quant-ph].

[131] John M. Martyn et al. “Grand Unification of Quantum Algorithms”. In: *PRX Quantum* 2 (Dec. 2021), p. 040203.

[132] András Gilyén et al. “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* (June 2019), pp. 193–204. arXiv: 1806.01838.

[133] Gregory Boyd and Bálint Koczor. “Training Variational Quantum Circuits with CoVaR: Covariance Root Finding with Classical Shadows”. In: *Phys. Rev. X* 12 (Nov. 2022), p. 041022.

[134] Lennart Bittel and Martin Kliesch. “Training Variational Quantum Algorithms Is NP-Hard”. In: *Phys. Rev. Lett.* 127 (12 Sept. 2021), p. 120502.

[135] Jarrod R McClean et al. “Barren plateaus in quantum neural network training landscapes”. In: *Nature Communications* 9 (2018), pp. 1–6.

[136] Barnaby van Straaten and Bálint Koczor. “Measurement Cost of Metric-Aware Variational Quantum Algorithms”. In: *PRX Quantum* 2 (3 Aug. 2021), p. 030324.

[137] Alessandro Ferraro, Stefano Olivares, and Matteo GA Paris. “Gaussian states in continuous variable quantum information”. In: *arXiv preprint quant-ph/0503237* (2005).

[138] Avishy Carmi and Eliahu Cohen. “On the significance of the quantum mechanical covariance matrix”. In: *Entropy* 20 (2018), p. 500.

[139] Vinay Tripathi, Chandrashekar Radhakrishnan, and Tim Byrnes. “Covariance matrix entanglement criterion for an arbitrary set of operators”. In: *New Journal of Physics* 22 (2020), p. 073055.

[140] Markus Reiher et al. “Elucidating Reaction Mechanisms on Quantum Computers”. In: *PNAS* (July 2017), p. 201619152.

[141] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. “A quantum approximate optimization algorithm”. In: *arXiv preprint arXiv:1411.4028* (2014).

[142] Peter D. Johnson et al. “QVECTOR: An Algorithm for Device-Tailored Quantum Error Correction”. In: *arXiv:1711.02249 [quant-ph]* (Nov. 2017). arXiv: 1711.02249 [quant-ph].

[143] Tyson Jones and Simon C. Benjamin. “Robust Quantum Compilation and Circuit Optimisation via Energy Minimisation”. In: *Quantum* 6 (Jan. 2022), p. 628.

[144] John E Dennis Jr and Robert B Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. SIAM, 1996.

[145] William H Press et al. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[146] Suguru Endo et al. “Hybrid Quantum-Classical Algorithms and Quantum Error Mitigation”. In: *J. Phys. Soc. Jpn.* 90 (Feb. 2021), p. 032001.

[147] M. Cerezo et al. “Variational Quantum Algorithms”. In: *Nature Reviews Physics* 3 (Sept. 2021), pp. 625–644.

[148] Kishor Bharti et al. “Noisy intermediate-scale quantum algorithms”. In: *Rev. Mod. Phys.* 94 (1 Feb. 2022), p. 015004.

[149] Hirotada Okawa et al. “The W4 method: a new multi-dimensional root-finding scheme for nonlinear systems of equations”. In: *arXiv preprint arXiv:1809.04495* (2018).

[150] L Pasquini and D Trigiante. “A globally convergent method for simultaneously finding polynomial roots”. In: *mathematics of computation* 44 (1985), pp. 135–149.

[151] Ming Gong et al. “Quantum Walks on a Programmable Two-Dimensional 62-Qubit Superconducting Processor”. In: *Science* 372 (May 2021), pp. 948–952.

[152] El Houcine Bergou et al. “A Stochastic Levenberg–Marquardt Method Using Random Models with Complexity Results”. In: *SIAM/ASA Journal on Uncertainty Quantification* 10 (2022), pp. 507–536.

[153] Shan Sung Liew, Mohamed Khalil-Hani, and Rabia Bakhteri. “An Optimized Second Order Stochastic Learning Algorithm for Neural Network Training”. In: *Neurocomputing* 186 (Apr. 2016), pp. 74–89.

[154] Ryan Sweke et al. “Stochastic gradient descent for hybrid quantum-classical optimization”. In: *Quantum* 4 (2020), p. 314.

[155] Marco Cerezo et al. “Variational quantum algorithms”. In: *arXiv preprint arXiv:2012.09265* (2020).

[156] Guido Pagano et al. “Quantum approximate optimization of the long-range Ising model with a trapped-ion quantum simulator”. In: *PNAS* 117 (2020), pp. 25396–25401.

[157] Matthew P Harrigan et al. “Quantum approximate optimization of non-planar graph problems on a planar superconducting processor”. In: *Nature Physics* 17 (2021), pp. 332–336.

[158] Maria Schuld et al. “Evaluating analytic gradients on quantum hardware”. In: *Phys. Rev. A* 99 (3 Mar. 2019), p. 032331.

[159] Hsin-Yuan Huang, Richard Kueng, and John Preskill. “Efficient Estimation of Pauli Observables by Derandomization”. In: *Physical Review Letters* 127 (July 2021), p. 030503.

[160] Ramon E. Moore, R. Baker Kearfott, and Michael J. Cloud. *Introduction to Interval Analysis*. Society for Industrial and Applied Mathematics, Jan. 2009.

[161] Norbert Schuch and Jens Siewert. “Programmable Networks for Quantum Algorithms”. In: *Physical Review Letters* 91 (July 2003), p. 027902.

[162] Lorenzo Moro et al. “Quantum Compiling by Deep Reinforcement Learning”. In: *Communications Physics* 4 (Aug. 2021), pp. 1–8.

[163] Andrew Lucas. “Ising Formulations of Many NP Problems”. In: *Frontiers in Physics* 2 (2014).

[164] C. Kokail et al. “Self-Verifying Variational Quantum Simulation of Lattice Models”. In: *Nature* 569 (May 2019), pp. 355–360.

[165] David J. Luitz, Nicolas Laflorencie, and Fabien Alet. “Many-body localization edge in the random-field Heisenberg chain”. In: *Physical Review B* 91 (Feb. 2015), p. 081103.

[166] Andrew M Childs et al. “Toward the first quantum simulation with quantum speedup”. In: *PNAS* 115 (2018), pp. 9456–9461.

[167] Sam McArdle et al. “Variational ansatz-based quantum simulation of imaginary time evolution”. In: *npj Quantum Information* 5 (2019), pp. 1–6.

[168] James Stokes et al. “Quantum natural gradient”. In: *Quantum* 4 (2020), p. 269.

[169] Naoki Yamamoto. “On the natural gradient for variational quantum eigensolver”. In: *arXiv preprint arXiv:1909.05074* (2019).

[170] Oscar Higgott, Daochen Wang, and Stephen Brierley. “Variational Quantum Computation of Excited States”. In: *Quantum* 3 (July 2019), p. 156.

[171] Jarrod R. McClean et al. “Hybrid Quantum-Classical Hierarchy for Mitigation of Decoherence and Determination of Excited States”. In: *Physical Review A* 95 (Apr. 2017), p. 042308.

[172] Matthew L. Goh and Bálint Koczor. *Direct Estimation of the Density of States for Fermionic Systems*. July 2024. arXiv: `2407.03414 [cond-mat, physics:quant-ph]`.

[173] Wooseop Hwang and Bálint Koczor. *Preparing Ground and Excited States Using Adiabatic CoVaR*. Oct. 2024. arXiv: `2409.16194 [quant-ph]`.

[174] Lennart Bittel and Martin Kliesch. “Training Variational Quantum Algorithms Is NP-Hard”. In: *Physical Review Letters* 127 (Sept. 2021), p. 120502.

[175] Ophelia Crawford et al. “Efficient quantum measurement of Pauli operators in the presence of finite sampling error”. In: *Quantum* 5 (2021), p. 385.

[176] Tzu-Ching Yen, Vladyslav Verteletskyi, and Artur F Izmaylov. “Measuring all compatible operators in one series of single-qubit measurements using unitary transformations”. In: *Journal of chemical theory and computation* 16 (2020), pp. 2400–2409.

[177] Andrew Jena, Scott Genin, and Michele Mosca. “Pauli partitioning with respect to gate sets”. In: *arXiv preprint arXiv:1907.07859* (2019).

[178] Pranav Gokhale et al. “O(N3) Measurement Cost for Variational Quantum Eigensolver on Molecular Hamiltonians”. In: *IEEE Transactions on Quantum Engineering* 1 (2020), pp. 1–24.

[179] Bálint Koczor. “Exponential Error Suppression for Near-Term Quantum Devices”. In: *Phys. Rev. X* 11 (3 Sept. 2021), p. 031057.

[180] Bálint Koczor. “The dominant eigenvector of a noisy quantum state”. In: *New Journal of Physics* 23 (2021), p. 123047.

[181] William J Huggins et al. “Virtual distillation for quantum error mitigation”. In: *Physical Review X* 11 (2021), p. 041036.

[182] Gregory Boyd, Bálint Koczor, and Zhenyu Cai. “High-Dimensional Subspace Expansion Using Classical Shadows”. In: *Physical Review A* 111 (Feb. 2025), p. 022423.

[183] Dmitry Grinko et al. “Iterative Quantum Amplitude Estimation”. In: *npj Quantum Inf* 7 (Mar. 2021), pp. 1–6.

[184] Zhenyu Cai et al. “Quantum Error Mitigation”. In: *Rev. Mod. Phys.* 95 (Dec. 2023), p. 045005.

[185] Hsin-Yuan Huang, Richard Kueng, and John Preskill. "Predicting Many Properties of a Quantum System from Very Few Measurements". In: *Nature Physics* 16 (Oct. 2020), pp. 1050–1057.

[186] Kouhei Nakaji et al. "Measurement Optimization of Variational Quantum Simulation by Classical Shadow and Derandomization". In: *Quantum* 7 (May 2023), p. 995.

[187] Tiff Brydges et al. "Probing Rényi Entanglement Entropy via Randomized Measurements". In: *Science* 364 (Apr. 2019), pp. 260–263.

[188] G.I. Struchalin et al. "Experimental Estimation of Quantum State Properties from Classical Shadows". In: *PRX Quantum* 2 (Jan. 2021), p. 010307.

[189] Sam McArdle et al. "Quantum Computational Chemistry". In: *Reviews of Modern Physics* 92 (Mar. 2020), p. 015003.

[190] Andrew J. Daley et al. "Practical Quantum Advantage in Quantum Simulation". In: *Nature* 607 (July 2022), pp. 667–676.

[191] Amira Abbas et al. *Quantum Optimization: Potential, Challenges, and the Path Forward*. arXiv:2312.02279 [quant-ph]. Dec. 2023.

[192] Jarrod R. McClean et al. "Decoding Quantum Errors with Subspace Expansions". In: *Nature Communications* 11 (Jan. 2020), p. 636.

[193] Jarrod R. McClean et al. "Hybrid Quantum-Classical Hierarchy for Mitigation of Decoherence and Determination of Excited States". In: *Phys. Rev. A* 95 (Apr. 2017), p. 042308.

[194] Tyler Takeshita et al. "Increasing the Representation Accuracy of Quantum Simulations of Chemistry without Extra Quantum Resources". In: *Physical Review X* 10 (Jan. 2020), p. 011004.

[195] Jeffrey Cohn, Mario Motta, and Robert M. Parrish. *Quantum Filter Diagonalization with Double-Factorized Hamiltonians*. arXiv:2104.08957 [quant-ph]. Apr. 2021.

[196] Nicholas J. Higham. "Computing a Nearest Symmetric Positive Semidefinite Matrix". In: *Linear Algebra and its Applications* 103 (May 1988), pp. 103–118.

[197] X. Bonet-Monroig et al. "Low-Cost Error Mitigation by Symmetry Verification". In: *Phys. Rev. A* 98 (Dec. 2018), p. 062339.

[198] Zhenyu Cai. "Quantum Error Mitigation Using Symmetry Expansion". In: *Quantum* 5 (Sept. 2021), p. 548.

[199] Andrew M. Childs et al. "Toward the First Quantum Simulation with Quantum Speedup". In: *Proceedings of the National Academy of Sciences* 115 (Sept. 2018), pp. 9456–9461.

[200] Jarrod R. McClean et al. "OpenFermion: The Electronic Structure Package for Quantum Computers". In: *Quantum Science and Technology* 5 (June 2020), p. 034014.

[201] Joonho Lee et al. "Generalized Unitary Coupled Cluster Wave Functions for Quantum Computation". In: *J. Chem. Theory Comput.* 15 (Jan. 2019), pp. 311–324.

[202] Sam McArdle, Xiao Yuan, and Simon Benjamin. "Error-Mitigated Digital Quantum Simulation". In: *Phys. Rev. Lett.* 122 (May 2019), p. 180501.

[203] Tom O'Leary et al. *Partitioned Quantum Subspace Expansion*. Mar. 2024. arXiv: 2403.08868 [physics, physics:quant-ph].

[204] Gregory Boyd. *Low-Overhead Parallelisation of LCU via Commuting Operators*. Aug. 2024. arXiv: 2312.00696 [quant-ph].

[205] Andrew M. Childs and Nathan Wiebe. *Hamiltonian Simulation Using Linear Combinations of Unitary Operations*. Nov. 2012.

[206] Guang Hao Low and Isaac L. Chuang. "Hamiltonian Simulation by Qubitization". In: *Quantum* 3 (July 2019), p. 163. arXiv: 1610.06546 [quant-ph].

[207] Chi-Fang Chen, Michael J. Kastoryano, and András Gilyén. *An Efficient and Exact Noncommutative Quantum Gibbs Sampler*. Nov. 2023. arXiv: 2311.09207 [cond-mat, physics:math-ph, physics:quant-ph].

[208] Andrew M. Childs, Robin Kothari, and Rolando D. Somma. "Quantum Algorithm for Systems of Linear Equations with Exponentially Improved Dependence on Precision". In: *SIAM Journal on Computing* 46 (Jan. 2017), pp. 1920–1950. arXiv: 1511.02306 [quant-ph].

[209] Gumaro Rendon, Jacob Watkins, and Nathan Wiebe. *Improved Error Scaling for Trotter Simulations through Extrapolation*. Dec. 2022. arXiv: 2212.14144 [quant-ph].

[210] Pei Zeng et al. *Simple and High-Precision Hamiltonian Simulation by Compensating Trotter Error with Linear Combination of Unitary Operations*. Dec. 2022. arXiv: 2212.04566 [quant-ph].

[211] Ryan Babbush et al. "Encoding Electronic Spectra in Quantum Circuits with Linear T Complexity". In: *Physical Review X* 8 (Oct. 2018), p. 041015. arXiv: 1805.03662 [cond-mat, physics:physics, physics:quant-ph].

[212] Joonho Lee et al. "Even More Efficient Quantum Computations of Chemistry through Tensor Hypercontraction". In: *PRX Quantum* 2 (July 2021), p. 030305. arXiv: 2011.03494 [physics, physics:quant-ph].

[213] Kianna Wan. "Exponentially Faster Implementations of Select(H) for Fermionic Hamiltonians". In: *Quantum* 5 (Jan. 2021), p. 380. arXiv: 2004.04170 [quant-ph].

[214] Vera von Burg et al. "Quantum Computing Enhanced Computational Catalysis". In: *Physical Review Research* 3 (July 2021), p. 033055. arXiv: 2007.14460 [physics, physics:quant-ph].

[215] Dominic W. Berry et al. "Qubitization of Arbitrary Basis Quantum Chemistry Leveraging Sparsity and Low Rank Factorization". In: *Quantum* 3 (Dec. 2019), p. 208.

[216] Zhicheng Zhang, Qisheng Wang, and Mingsheng Ying. *Parallel Quantum Algorithm for Hamiltonian Simulation*. Jan. 2023. arXiv: 2105.11889 [quant-ph].

[217] Sina Zeytinoğlu and Sho Sugiura. *Error-Robust Quantum Signal Processing Using Rydberg Atoms*. Dec. 2022. arXiv: 2201.04665 [quant-ph].

[218] Nobuyuki Yoshioka et al. *Hunting for Quantum-Classical Crossover in Condensed Matter Problems*. Oct. 2023. arXiv: 2210.14109 [cond-mat, physics:quant-ph].

[219] Cristopher Moore and Martin Nilsson. *Parallel Quantum Computation and Quantum Codes*. Aug. 1998. arXiv: quant-ph/9808027.

[220] Peter Hoyer and Robert Spalek. "Quantum Circuits with Unbounded Fan-out". In: *Theory of Computing* 1 (2005), pp. 81–103. arXiv: quant-ph/0208043.

[221] Harry Buhrman et al. *State Preparation by Shallow Circuits Using Feed Forward*. July 2023. arXiv: 2307.14840 [quant-ph].

[222] Daniel Gottesman and Isaac L. Chuang. "Quantum Teleportation Is a Universal Computational Primitive". In: *Nature* 402 (Nov. 1999), pp. 390–393. arXiv: quant-ph/9908010.

[223] Ophelia Crawford et al. "Efficient Quantum Measurement of Pauli Operators in the Presence of Finite Sampling Error". In: *Quantum* 5 (Jan. 2021), p. 385. arXiv: 1908.06942 [quant-ph].

[224] Pranav Gokhale et al. *Minimizing State Preparations in Variational Quantum Eigensolver by Partitioning into Commuting Families*. July 2019. arXiv: 1907.13623 [quant-ph].

[225] Vladyslav Verteletskyi, Tzu-Ching Yen, and Artur F. Izmaylov. "Measurement Optimization in the Variational Quantum Eigensolver Using a Minimum Clique Cover". In: *The Journal of Chemical Physics* 152 (Mar. 2020), p. 124114. arXiv: 1907.03358 [physics, physics:quant-ph].

[226] Tzu-Ching Yen, Vladyslav Verteletskyi, and Artur F. Izmaylov. "Measuring All Compatible Operators in One Series of Single-Qubit Measurements Using Unitary Transformations". In: *Journal of Chemical Theory and Computation* 16 (Apr. 2020), pp. 2400–2409.

[227] Andrew Jena, Scott Genin, and Michele Mosca. *Pauli Partitioning with Respect to Gate Sets*. July 2019. arXiv: 1907.07859 [quant-ph].

[228] Pei Yuan and Shengyu Zhang. "Optimal (Controlled) Quantum State Preparation and Improved Unitary Synthesis by Quantum Circuits with Any Number of Ancillary Qubits". In: *Quantum* 7 (Mar. 2023), p. 956. arXiv: 2202.11302 [quant-ph].

[229] Xiao-Ming Zhang, Tongyang Li, and Xiao Yuan. "Quantum State Preparation with Optimal Circuit Depth: Implementations and Applications". In: *Physical Review Letters* 129 (Nov. 2022), p. 230504. arXiv: 2201.11495 [quant-ph].

[230] Dmitri Maslov and Ben Zindorf. "Depth Optimization of CZ, CNOT, and Clifford Circuits". In: *IEEE Transactions on Quantum Engineering* 3 (2022), pp. 1–8.

[231] Yi-Cong Zheng et al. *Depth Reduction for Quantum Clifford Circuits through Pauli Measurements*. May 2018. arXiv: 1805.12082 [quant-ph].

[232] Timothy Proctor and Kevin Young. *A Simple Asymptotically Optimal Clifford Circuit Compilation Algorithm*. Oct. 2023. arXiv: 2310.10882 [quant-ph].

[233] Kai Chen and Hoi-Kwong Lo. *Multi-Partite Quantum Cryptographic Protocols with Noisy GHZ States*. Apr. 2008. arXiv: quant-ph/0404133.

[234] Yasuhiro Takahashi and Seiichiro Tani. *Collapse of the Hierarchy of Constant-Depth Exact Quantum Circuits*. Sept. 2012. arXiv: 1112.6063 [quant-ph].

[235] Sergey Bravyi and Alexei Kitaev. "Universal Quantum Computation with Ideal Clifford Gates and Noisy Ancillas". In: *Physical Review A* 71 (Feb. 2005), p. 022316.

[236] Michael E. Beverland et al. *Assessing Requirements to Scale to Practical Quantum Advantage*. Nov. 2022. arXiv: 2211.07629 [quant-ph].

[237] Daniel Litinski. "Magic State Distillation: Not as Costly as You Think". In: *Quantum* 3 (Dec. 2019), p. 205.

[238] Wai-Keong Mok et al. *Rigorous Noise Reduction with Quantum Autoencoders*. Aug. 2023. arXiv: 2308.16153 [quant-ph].

[239] Craig Gidney, Noah Shutty, and Cody Jones. *Magic State Cultivation: Growing T States as Cheap as CNOT Gates*. Sept. 2024. arXiv: 2409.17595 [quant-ph].

[240] Richard Jozsa and Akimasa Miyake. "Matchgates and Classical Simulation of Quantum Circuits". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 464 (Dec. 2008), pp. 3089–3106. arXiv: 0804.4050 [quant-ph].

[241] Craig Gidney. "Stim: a fast stabilizer circuit simulator". In: *Quantum* 5 (July 2021), p. 497.

[242] Tiago P. Peixoto. "The graph-tool python library". In: *figshare* (2014).

[243] M. Born and V. Fock. "Beweis des Adiabatensatzes". In: *Zeitschrift für Physik* 51 (Mar. 1928), pp. 165–180.

[244] Seunghoon Lee et al. *Is There Evidence for Exponential Quantum Advantage in Quantum Chemistry?* Nov. 2022. arXiv: 2208.02199 [physics, physics:quant-ph].

[245] Min Zhuang et al. "Symmetry-Protected Quantum Adiabatic Evolution in Spontaneous Symmetry-Breaking Transitions". In: *Annalen der Physik* 532 (Apr. 2020), p. 1900471. arXiv: 1810.05805 [cond-mat, physics:quant-ph].

[246] Eli Chertkov and Bryan K. Clark. "Computational Inverse Method for Constructing Spaces of Quantum Models from Wave Functions". In: *Physical Review X* 8 (July 2018), p. 031029.

[247] Xiao-Liang Qi and Daniel Ranard. "Determining a Local Hamiltonian from a Single Eigenstate". In: *Quantum* 3 (July 2019), p. 159. arXiv: 1712.01850 [cond-mat, physics:quant-ph].

[248] Arthur G. Rattew and Bálint Koczor. *Preparing Arbitrary Continuous Functions in Quantum Registers With Logarithmic Complexity*. May 2022. arXiv: arXiv:2205.00519.

[249] Jérémie Roland and Nicolas J. Cerf. "Quantum Search by Local Adiabatic Evolution". In: *Physical Review A* 65 (Mar. 2002), p. 042308.

[250] Avatar Tulsi. "Adiabatic Quantum Computation with a 1D Projector Hamiltonian". In: *Physical Review A* 80 (Nov. 2009), p. 052328. arXiv: 0806.0385 [quant-ph].

[251] Jérémie Roland and Nicolas J. Cerf. "Adiabatic Quantum Search Algorithm for Structured Problems". In: *Physical Review A* 68 (Dec. 2003), p. 062312. arXiv: quant-ph/0304039.

[252] Yu-Xin Wang et al. "Entanglement Rényi α-Entropy". In: *Physical Review A* 93 (Feb. 2016), p. 022324.

[253] Richard Meister, Cica Gustiani, and Simon C. Benjamin. *Exploring Ab Initio Machine Synthesis of Quantum Circuits*. June 2022. arXiv: 2206.11245 [quant-ph].

[254] Matthias Christandl and Andreas Winter. ""Squashed Entanglement": An Additive Entanglement Measure". In: *Journal of Mathematical Physics* 45 (Mar. 2004), pp. 829–840.

[255] Hengyun Zhou et al. *Algorithmic Fault Tolerance for Fast Quantum Computing*. June 2024. arXiv: 2406.17653 [quant-ph].

[256] Tyson Jones et al. "QuEST and high performance simulation of quantum computers". In: *Sci. Rep.* 9 (2019), p. 10736.

[257] Tyson Jones and Simon C. Benjamin. "QuESTlink – Mathematica Embiggened by a Hardware-Optimised Quantum Emulator". In: *Quantum Sci. Technol.* (2020).