Graph-Theoretic Techniques for Optimizing NISQ Algorithms

by

Andrew Jena

A thesis
presented to the University Of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Combinatorics and Optimization (Quantum Information)

Waterloo, Ontario, Canada, 2024

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Artur Izmaylov Professor, Dept. of Physical and Environmental Sciences, University of Toronto Scarborough
Supervisor:	Michele Mosca Professor, Dept. of Combinatorics and Optimization, University of Waterloo
Internal Member:	David Gosset Associate Professor, Dept. of Combinatorics and Optimization, University of Waterloo
Internal Member:	Jon Yard Associate Professor, Dept. of Combinatorics and Optimization, University of Waterloo
Internal-External Member:	Christine Muschik Associate Professor, Dept. of Physics and Astronomy, University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

- Chapter 1 is an introduction to the rest of the thesis with no novel material, with background material properly cited.

- Chapter 2 is background work and definitions, which have been collected from various sources cited in the text.

- Chapter 3 is the result of a collaboration between myself, Lane Gunderman, and Luca Dellantonio, and is based on a preprint article on which I am a first author [1]. While doing background reading, it was discovered that our reduction techniques are similar to those used by Bravyi et al.'s qubit tapering algorithm [2]. This was unknown to us while researching and writing the preprint, and I have made every effort here to give credit to the ideas which first appeared in their article. The figures which depict the overview of the algorithm were helpfully constructed by Luca, and the proof of optimality we present is largely based on Lane's independent work [3], so he contributed mightily to the specifics of the proof. I have contributed equally or more to the rest of the material presented in Chapter 3.

- Chapter 4 is the result of a collaboration between myself, Ariel Shlosberg, Priyanka Mukhopadhyay, Jan Haase, Felix Leditzky, and Luca Dellantonio, and is based on a published article on which I am a first author [4]. The machine learning algorithm and simulations, as well as the description of the algorithm, is solely the work of Ariel. It is included here for the sake of completeness, as I believe it is extremely important to be able to compare and contrast the bucket filling and machine learning versions of AEQuO. The formulas we use for our Bayesian estimates of the covariance matrix were worked out in detail by Luca, who also again supplied the figures. The covariance formula, the bucket filling and quantum simulation code, and many of the numerical results were contributed in large or sole part by me. I have contributed equally or more to the rest of the material presented in Chapter 4.

- Chapter 5 is the result of a collaboration between myself and James Davies, and is based on an as-of-yet unpublished work, tentatively titled "Preparing classes of graph states," on which I will be a first author. The discussion here delves more into algorithms for traversing the quotient graph representation of stabilizer states than the source material, for which I am solely responsible. James and I discussed all of the proofs presented in this chapter, though he is largely to thank for formalizing some of the more theoretical ideas, especially concerning graph perturbations. I would also be remiss not to mention that many of the ideas were sparked by conversations between myself, Lane Gunderman, and Luca Dellantonio.

- Chapter 6 is a conclusion written in my own words, which contains no novel results necessitating citations or contributions. The futures ideas which I present are the result of ongoing conversations with many of the collaborators listed above.

Abstract

Entering the NISQ era, the search for useful yet simple quantum algorithms is perhaps of more importance now than it may ever be in the future. In place of quantum walks, the quantum Fourier transform, and asymptotic results about far-term advantages of quantum computation, the real-world applications of today involve nitty-gritty details and optimizations which make the most use of our limited resources. These priorities pervade the research presented in this thesis, which focuses on combinatorial techniques for optimizing NISQ algorithms.

With variational algorithms directing the discussion, we investigate methods for reducing Hamiltonians, reducing measurement errors, and reducing entangling gates. All three of these reductions bring us ever closer to demonstrating the utility of quantum devices, by improving some of the major bottlenecks of the NISQ era, and all of them do so while rarely ever leaving the combinatorial framework provided by stabilizer states. The qubit tapering approach to Hamiltonian simplification which we present was developed independently of the work by Bravyi et al., who discovered how to reduce qubit counts by parallelizing the computation of the ground state [1, 2]. The measurement scheme we describe, AEQuO, is built upon years of research and dozens of articles detailing, comparing, and contrasting a plethora of schemes [4]. The circuit optimization technique we introduce answers a question posed by Adcock et al., and our ideas and proofs are fundamentally grounded in the literature of isotropic systems and the graph-based results which have followed from it [5, 6].

Acknowledgements

There are a tremendous number of people who have made this journey possible, and each one of them deserves more acknowledgment than I could put into words. My friends and family, and especially my wife Madeleine, who put up with and supported me through long nights of research, are owed a great thanks.

I am extremely grateful to my advisor, Michele, who has been a great mentor and always allowed and encouraged me to branch out and explore my many interests. Without his approval, I never would have managed to work with so many great researchers from so many disparate fields.

On the note of great researchers, I am thankful to all of my collaborators throughout my years at Waterloo. Coming into the field of quantum computing from a pure math background, I was left with many holes in my knowledge of physics and chemistry, which I was only able to patch with the help of those talented individuals.

I would especially like to thank Scott Genin, who agreed to mentor me for my first quantum chemistry project during my masters. Without him, I may have never found my way to researching the VQE, which has led to the results presented in this thesis.

Of special note as well are two of my collaborators and friends, Lane Gunderman and Luca Dellantonio, with whom I have shared a group chat for the last few years because of how constantly we have wanted to share ideas with one another. It may very well be true that every result presented in this thesis has been discussed at one point or another between all of us.

Dedication

I dedicate this thesis to my wife, Madeleine, and our daughter on the way.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction and Motivation

Too late to be the first to break RSA encryption, yet too early to be certain which evolving quantum technology will be scalable, it is perhaps more difficult today for modern researchers in quantum algorithms to find their niche. During my masters research, I was introduced to the variational quantum eigensolver, which pairs a quantum device with a classical optimizer in an effort to provide near-term quantum advantages. As a combinatorialist at heart, I found my niche when I first learned that one bottleneck of this algorithm was the number of measurements required, and I wondered whether a graph-theoretic viewpoint could help in designing a more efficient measurement scheme. This positive experience led me to consider further applications of combinatorial techniques to quantum algorithm design.

In Chapter 2, we discuss relevant prior work, and some concurrent results from the many researchers who were simultaneously answering similar questions and related topics.

In Chapter 3, we begin by attempting to reduce the complexity of the input to the problem. We will show how to leverage Pauli symmetries to provide an efficient algorithm for parallelizing the calculation of the ground state of a local Hamiltonian. This work builds upon the idea of tapering qubits [2], which provides a method for splitting the computation of eigenstates of a Hamiltonian into the two eigenspaces of a commuting Pauli operator. Here, we flesh out this idea and discuss when the exponential classical cost can be avoided, as well as how to extend these results to non-Pauli symmetries.

In Chapter 4, we introduce AEQuO, a comprehensive measurement scheme for local Hamiltonians which estimates the measurement error based on prior results and adaptively chooses future measurements to minimize the error. AEQuO is novel in how it uses

Bayesian estimate of the variance and covariance of pairs of Pauli operators to allocate future measurements. The ability to change between qubitwise and general commutation and the ability to make use of overlapping measurements provides flexibility to those wishing to implement the algorithm on disparate quantum devices. Similarly, the Bayesian estimate of the total measurement error was more exact than the error obtained by repeating the algorithm many times for all of the Hamiltonians we tested, which leads to a more efficient method for comparing and contrasting future measurement schemes.

In Chapter 5, we investigate the problem of minimizing entangling gates in Clifford circuits. In doing so, we delve into the graph state formalism and develop an A* search algorithm for optimizing Clifford circuits. We provide novel insights into classes of graph states that can be constructed using asymptotically fewer entangling gates than the worst case graphs which require $\Omega(n^2/\log n)$ gates, and we use an outstanding conjecture of Jim Geelen to expand our class to cover a large family of graphs.

Finally, in Chapter 6, we conclude by providing insight into open questions and directions for future work, with nodes towards ongoing research and interesting avenues for other researchers.

Chapter 2

Background and Definitions

The overwhelming majority of techniques used to prove the results presented in this thesis make use of the fact that the q-dimensional Pauli group over n qubits, ignoring phases, is isomorphic to $(\mathbb{Z}/q\mathbb{Z})^{2n}$. The symplectic representation of phase-less Pauli operators will allow us to treat Pauli operators as vectors over a finite field, and the fact that Clifford group is the normalizer of the Pauli group will allow us to view Clifford circuits, acting by conjugation, as a permutation of these vectors. This provides the foundation for the fact that such circuits can be efficiently simulated is well-known as the Gottesman-Knill Theorem [7]. As such, it only seems right to begin our background and definitions here.

2.1 Paulis and Cliffords

For a thorough review of modern physics, quantum mechanics, and all such relevant background topics, nothing I could say, as a mathematician, could compete with the reviews presented in the standard textbooks in this field [8, 9, 10]. The following definitions will suffice for our purposes.

Definition 1 (Quantum State) Taking inspiration from John Watrous [10], since we will not concern ourselves with infinite-dimensional states, it suffices to define quantum states as vectors over a finite complex Euclidean space. Since we will restrict ourselves to unitary transformations, it will also suffice to assume states are normalized.

Formally, a q-dimensional *quantum state* is a normalized vector in $\mathbb{C}^q$.

$$|\phi\rangle = \sum_{i=0}^{q-1} \phi_i \, |i\rangle = \begin{pmatrix} \phi_0 \\ \vdots \\ \phi_{q-1} \end{pmatrix},$$

where $\phi_i \in \mathbb{C}$ and $\sum_{i=0}^{q-1} |\phi_i|^2 = 1$. □

Definition 2 (Unitary Operator) A *unitary operator* over $\mathbb{C}^q$ is an isometric linear operator which maps $\mathbb{C}^q \to \mathbb{C}^q$, which means that the operator preserves the 2-norm of the vector. This is equivalent to the statement that

$$U \text{ is unitary} \iff UU^\dagger = U^\dagger U = \mathbb{1}_q,$$

where $U^\dagger$ is the complex conjugate of the transpose of U, also known as the Hermitian conjugate or the adjoint. Moreover, since quantum algorithms map quantum states to one another, every quantum algorithm can be written as a unitary operator. □

Definition 3 (Hermitian Operator) A *Hermitian operator* over $\mathbb{C}^q$ is a linear operator which satisfies

$$H \text{ is Hermitian} \iff H = H^\dagger.$$

For the purposes of quantum algorithms, it is important to mention that the eigenvalues of Hermitian operators are real. It is likewise important that $e^{iH}\left(e^{iH}\right)^\dagger = \left(e^{iH}\right)^\dagger e^{iH} = e^{-iH}e^{iH} = e^{0_q} = \mathbb{1}_q$, which means that e^{iH} is unitary. In fact, every unitary operator, U, can be written as $U = e^{iH}$ for some Hermitian operator, H. □

Definition 4 (Qubit) In real-world quantum systems, multiple quantum states can be prepared and acted upon independently, or entangled and acted upon jointly. We will refer to each of these substates as a *qubit*, and we will assume that, in a given system, all of the qubits are of the same dimension, 2. Therefore, an n-qubit state lives in $\mathbb{C}^{2^n}$. □

Definition 5 (k-Local Hamiltonian [11]) Working with $2^n \times 2^n$ complex matrices classically is infeasible, since reading each entry in the corresponding matrix would require an exponential number of operations. Instead, we will often assume that a physical system of interest, with a corresponding Hermitian Hamiltonian, can be represented efficiently. To ensure this, we assume that our Hamiltonian can be written as a sum of poly(n) Hermitian terms, where each term acts non-trivially on at most k qubits. Such a Hamiltonian is called *k-local.* □

Definition 6 (Pauli Group [12]) Fortunately, there is a helpful basis over which the $2^n \times 2^n$ Hermitian operators form a real vector space. This basis is a group under multiplication called the *Pauli group*, which we will denote by $\mathcal{P}_2^n$. Each element of $\mathcal{P}_2^n$ is a tensor product of the following matrices together with an overall phase of ±1 or $\pm i$:

$$\mathtt{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \mathtt{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad \mathtt{Y} = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad \mathtt{Z} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \qquad \square$$

The Pauli group forms a basis for the set of Hermitian operators, and its elements satisfy the following commutation relations:

$$\begin{array}{llll}
\mathtt{I} \cdot \mathtt{I} = \mathtt{I} \cdot \mathtt{I} & \mathtt{I} \cdot \mathtt{X} = \mathtt{X} \cdot \mathtt{I} & \mathtt{I} \cdot \mathtt{Y} = \mathtt{Y} \cdot \mathtt{I} & \mathtt{I} \cdot \mathtt{Z} = \mathtt{Z} \cdot \mathtt{I} \\
\mathtt{X} \cdot \mathtt{I} = \mathtt{I} \cdot \mathtt{X} & \mathtt{X} \cdot \mathtt{X} = \mathtt{X} \cdot \mathtt{X} & \mathtt{X} \cdot \mathtt{Y} = -\mathtt{Y} \cdot \mathtt{X} & \mathtt{X} \cdot \mathtt{Z} = -\mathtt{Z} \cdot \mathtt{X} \\
\mathtt{Y} \cdot \mathtt{I} = \mathtt{I} \cdot \mathtt{Y} & \mathtt{Y} \cdot \mathtt{X} = -\mathtt{X} \cdot \mathtt{Y} & \mathtt{Y} \cdot \mathtt{Y} = \mathtt{Y} \cdot \mathtt{Y} & \mathtt{Y} \cdot \mathtt{Z} = -\mathtt{Z} \cdot \mathtt{Y} \\
\mathtt{Z} \cdot \mathtt{I} = \mathtt{I} \cdot \mathtt{Z} & \mathtt{Z} \cdot \mathtt{X} = -\mathtt{X} \cdot \mathtt{Z} & \mathtt{Z} \cdot \mathtt{Y} = -\mathtt{Y} \cdot \mathtt{Z} & \mathtt{Z} \cdot \mathtt{Z} = \mathtt{Z} \cdot \mathtt{Z}.
\end{array}$$

Since multiplication by a constant distributes over tensor products, it's possible to efficiently determine whether two elements of the Pauli group commute based on the parity of the number of locations on which their terms "disagree." In fact, this table of relations and this notion of commutation give rise to a notation for Pauli operators as vectors over $(\mathbb{Z}/2\mathbb{Z})^n$.

Definition 7 (Symplectic Formalism [12]) Ignoring phases, we can map qubit Paulis to the following binary symplectic vectors in $(\mathbb{Z}/2\mathbb{Z})^2$

$$\begin{aligned}
\mathtt{I} \propto \mathtt{X}^0\mathtt{Z}^0 &\mapsto \left(\mathsf{0} \,\middle|\, \mathsf{0} \right) \\
\mathtt{X} \propto \mathtt{X}^1\mathtt{Z}^0 &\mapsto \left(\mathsf{1} \,\middle|\, \mathsf{0} \right) \\
\mathtt{Y} \propto \mathtt{X}^1\mathtt{Z}^1 &\mapsto \left(\mathsf{1} \,\middle|\, \mathsf{1} \right) \\
\mathtt{Z} \propto \mathtt{X}^0\mathtt{Z}^1 &\mapsto \left(\mathsf{0} \,\middle|\, \mathsf{1} \right).
\end{aligned}$$

The X-part will always be on the left and the Z-part will always be on the right. We can similarly map tensors of Paulis to vectors in $(\mathbb{Z}/2\mathbb{Z})^{2n}$ in the natural way

$$\mathtt{I} \otimes \mathtt{X} \otimes \mathtt{Y} \otimes \mathtt{Z} \mapsto \left(\mathsf{0110} \,\middle|\, \mathsf{0011} \right). \qquad \square$$

Definition 8 (Symplectic Inner Product [12]) In the language of these binary symplectic vectors, the *symplectic inner product*, denoted by $\odot$, tells us whether two Pauli operators, P and Q, commute by the formula $PQ = (-1)^{P \odot Q} QP$. This function is defined as

$$\left(x_1, \ldots, x_n \,\middle|\, z_1, \ldots, z_n \right) \odot \left(x'_1, \ldots, x'_n \,\middle|\, z'_1, \ldots, z'_n \right) = \sum_{i=1}^{n} (x_i z'_i - z_i x'_i) \pmod 2.$$

When P and Q are one-qubit operators, it's easy to verify that this formula reproduces the table of commutation relations. □

The above definition uses subtraction rather than addition inside the summation, which is an odd choice when working modulo 2. However, this choice is actually important for a generalization of the Pauli operators to prime dimensions, at which point subtraction and addition are not equivalent operations. We will only briefly touch on generalizations of our results to prime fields, but suffice it to say that most results which can be proved for qubit Paulis and Cliffords can likewise be proved for generalized Paulis and generalized Cliffords.

In switching to the symplectic formalism, we have chosen to ignore overall phases. These phases are not physically realizable (there is no measurement which can determine the overall phase of a quantum state), but this does not mean they are useless. In fact, keeping track of overall phases becomes crucial when translating many of these results to practical purposes, since the Hamiltonian $P + Q$ is not equivalent to $P - Q$, where P and Q are two Paulis. We will make note of the situations in which this nuanced difference becomes relevant.

Definition 9 (Clifford Group [12]) Last but certainly not least for this section of definitions, we turn our attention to the Clifford group, which is the normalizer of the set of Pauli operators. Formally,

$$\mathcal{C}_2^n = \{g \text{ unitary} : gPg^\dagger \in \mathcal{P}_2^n,\ \forall\, P \in \mathcal{P}_2^n\}.$$

There are multiple useful ways to define a basis for the Clifford group, so here we will simply define some of the commonly used operators. Since these operators permute the elements of the Pauli group when acting by conjugation, we will define them by their actions on Paulis. There is a set of $2n$ Paulis which multiplicatively generate the n-qubit Pauli group up to global phases, given by weight-1 $\mathtt{X}$ and $\mathtt{Z}$ Pauli operators. Therefore, it suffices to define the action of Clifford operators on these basis elements:

$$\mathrm{H} : \begin{array}{l} \mathtt{X} \mapsto \mathtt{Z} \\ \mathtt{Z} \mapsto \mathtt{X} \end{array}$$

$$\mathrm{S} : \begin{array}{l} \mathtt{X} \mapsto \mathtt{Y} \\ \mathtt{Z} \mapsto \mathtt{Z} \end{array}$$

$$\mathrm{CNOT}_{12} : \begin{array}{l} \mathtt{XI} \mapsto \mathtt{XX} \\ \mathtt{IX} \mapsto \mathtt{IX} \\ \mathtt{ZI} \mapsto \mathtt{ZI} \\ \mathtt{IZ} \mapsto \mathtt{ZZ} \end{array}$$

$$\mathrm{CZ}_{12} : \begin{array}{l} \mathtt{XI} \mapsto \mathtt{XZ} \\ \mathtt{IX} \mapsto \mathtt{ZX} \\ \mathtt{ZI} \mapsto \mathtt{ZI} \\ \mathtt{IZ} \mapsto \mathtt{IZ} \end{array}$$

The above gates are called Hadamard, Phase, Controlled-Not (with a control on the first qubit and a target on the second qubit), and Controlled-Z, respectively. Importantly, these definitions have ignored the overall phases that are needed to make the above mappings formally correct. This will be discussed when it becomes relevant. □

Combining the above facts together, it's possible to see that we can simulate any circuit composed of Clifford gates efficiently using classical techniques, simply by tracking a basis of $2n$ Pauli operators. This was first formally proved by Gottesman and Knill, and the result has come to be known as the Gottesman-Knill Theorem [7]. This fact is crucial to most of the results in this thesis. Importantly, our restriction to Clifford circuits is not the only reasonable choice for NISQ optimizations. In fact, the CNOT and CZ gates above often prove more costly than some non-Clifford single-qubit gates on NISQ devices. Applications of non-Clifford transformations to the chemistry problem we will introduce in Section 2.2 have been considered [13, 14].

2.2 NISQ Applications

Here, we will discuss applications of the definitions in Section 2.1. As we enter the noisy intermediate-scale quantum (NISQ) era [15], with access to tens or hundreds of error-prone qubits, it is important to investigate the simplest algorithms to scour for minute speedups and gains in efficiency or robustness to leverage in our quest to obtain a useful quantum advantage.

This process begins with identifying the problems which are especially suited to realizing quantum speedups. Quantum systems offer the prospect of accelerating computations [16, 8, 17, 15] and simulations [18] within a wide range of applications, such as high-energy physics [19, 20, 21] and chemistry [22]. In particular, before Peter Shor developed the algorithm that would skyrocket the popularity of quantum computing by breaking RSA encryption in polynomial time [16], Richard Feynman proposed the natural application of simulating physics on quantum computers [23]. Physical phenomena, which are fundamentally quantum in nature, lend themselves to quantum speedups.

One field which has embraced this fact is quantum chemistry, the study of molecular properties by understanding arrangements of electrons. Of particular interest is the problem of determining the ground state energy, i.e. the lowest eigenvalue, of a molecular Hamiltonian. When we restrict to local Hamiltonians, which are efficient to represent classically, this is often referred to as the local Hamiltonian problem. We will state the decision version of the local Hamiltonian problem here.

Definition 10 (*k*-Local Hamiltonian Problem [11]) Given a k-local n-qubit Hamiltonian, $H = \sum_{j=1}^{\text{poly}(n)} H_j$, where each H_j has a bounded operator norm $\|H_j\| \leq \text{poly}(n)$, and two constants, $a, b \in \mathbb{R}$ with $a < b$ along with the promise that the lowest eigenvalue of H is less than a or greater than b, determine which is true. □

Unfortunately for quantum chemists, this problem is actually QMA-complete, which is the quantum analogue of NP-complete [24]. Fortunately, though, there is still plenty of hope for quantum devices to improve upon the best-known classical techniques, and this is especially true for restricted classes of Hamiltonians. There are fault-tolerant approaches [25] which will require fault-tolerant, large-scale quantum devices, but there are also clever ways to use NISQ-era devices alongside classical optimizers to try to solve this problem. This approach has become known as the variational quantum eigensolver, or VQE, and since this algorithm has provided much of the motivation for the results in this thesis, it is important to go over the VQE in some detail.

2.2.1 Variational Quantum Eigensolver

The goal of the *variational quantum eigensolver (VQE)* is to approximately solve the k-local Hamiltonian problem. To do so, the VQE iteratively prepares states which approach the ground state of the Hamiltonian; that is, the eigenstate associated with the lowest eigenvalue. As mentioned, this process is iterative, and the measurement outcomes in initial steps inform the state preparation in successive steps. The VQE can fundamentally be broken into three distinct steps:

- the ansatz,
- the measurement protocol,
- the classical optimizer [26].

The majority of this thesis will cover topics which relate to the measurement allocation protocol, but it is important to give context for the many choices which can be made at each step. Before doing so, we will go into a bit more detail on how these three steps are chained together.

As mentioned in the definition, the VQE will attempt to prepare the ground state of the Hamiltonian, which is a difficult task when the ground state is unknown. To tackle this, the VQE uses a parameterized quantum algorithm to prepare a family of states which may admit parameters which approximate the ground state to a sufficient error. This state preparation algorithm is often referred to as an ansatz, and we will use the quantum state, $|\psi(\theta)\rangle$, to represent the output of such an algorithm with a vector of parameters, θ.

Given this prepared state, though, we must find a way to measure how close this state is to the ground state, and eventually to be able to measure the ground state energy itself. To do so, we use the fact that, if $|\phi_1\rangle$ is the ground state of a Hamiltonian $\mathcal{H}$ with corresponding ground state energy, λ_1, then

$$\langle\phi_1|\mathcal{H}|\phi_1\rangle = \lambda\,\langle\phi_1|\phi_1\rangle = \lambda_1.$$

In other words, measuring this expectation gives the ground state energy. For an arbitrary state, $|\psi\rangle$, if we write $|\psi\rangle$ as a linear combination in the eigenbasis of $\mathcal{H}$,

$$|\psi\rangle = \sum_{i=1}^{2^n} \alpha_i \, |\phi_i\rangle\,,$$

then the expectation value of $\mathcal{H}$ with respect to $|\psi\rangle$ is

$$\sum_{i=1}^{2^n} |\alpha_i|^2 \,\langle\phi_i|\mathcal{H}|\phi_i\rangle = \sum_{i=1}^{2^n} |\alpha_i|^2 \lambda_i.$$

Since the $|\alpha_i|^2$ terms sum to 1 whenever $|\psi\rangle$ is normalized, this is just a weighted average over the eigenvalues of $\mathcal{H}$, which must be at least as large as the minimum eigenvalue. In other words, the absolute minimum of $\langle\psi|\mathcal{H}|\psi\rangle$ is achieved when $|\psi\rangle$ is the ground state of $\mathcal{H}$. This will be our objective function, so the goal of our measurement protocol will be to estimate $\langle\psi(\theta)|\mathcal{H}|\psi(\theta)\rangle$ to the required accuracy using as few measurements as possible.

On that note, the required accuracy will be determined by the classical optimizer of choice. While the task laid out above could be tackled with any optimization technique, there are a few eccentricities particular to NISQ-era implementations of the VQE which may influence the choice of optimizer. Since the number of runs of the ansatz and measurement protocol will likely be the bottleneck, and since NISQ-era devices are liable to produce measurement outcomes with larger variances, an optimizer which is robust against noise is likely advisable. Moreover, most molecular Hamiltonians of interest are highly structured, unlike random Hamiltonians, and thus the details of the problem, and the optimizer of choice, will depend highly on the chosen molecule. The best choice which balances these factors is generally unknown. A review of NISQ-era algorithms has discussed the pros and cons of various techniques at lengths far greater than I would be capable of replicating here [26].

Given a k-local Hamiltonian, an ansatz, a measurement protocol, and a classical optimizer, you are ready to run the VQE. Since the k-local Hamiltonian problem is QMA-complete, though, there are no performance guarantees, regardless of the quality of your quantum device. However, the ability to sample from the distribution of these expectation values, which is thought to be difficult classically, is enough reason to believe that the VQE might one day compete with the best known classical approaches to tackling this problem.

2.3 Background for Chapter 3

We will take this opportunity to cover relevant background specific to each of the following chapters. In these sections, I will cover background in broad swaths so as to not overlap with the more problem-specific background covered in the individual chapters.

First and most importantly for Chapters 3 and 4, it is important to note that the qubit Hamiltonian we assume we are given is not the true starting place of a quantum chemistry problem. In particular, to specify an electronic structure Hamiltonian in a given basis, one constructs a Fermionic Hamiltonian in second quantized form

$$\hat{H} = \sum_{i,j} h_{ij} a_i^\dagger a_j + \frac{1}{2} \sum_{i,j,k.l} h_{i,j,k,l} a_i^\dagger a_j^\dagger a_k a_l,$$

where $a_i^\dagger$ and a_i are the creation and annihilation operators on the i^{th} basis orbital and h_{ij} and h_{ijkl} are the Coulombic overlap and exchange integrals determined by the basis [27].

To map this Hamiltonian onto a quantum device, therefore, necessitates the use of one of various transforms, including the Jordan-Wigner transformation [28, 25], the parity transformation [29], or the Bravyi-Kitaev transformation [30, 29], among others. Each of these results from a different mapping of basis orbitals and creation and annihilation operators onto qubits.

The Jordan-Wigner transformation maps the i^{th} basis orbital to the i^{th} qubit in the natural way, where an occupied (unoccupied) i^{th} basis orbital corresponds to the i^{th} qubit in the state $|1\rangle$ ($|0\rangle$). The resulting creation and annihilation operators are represented by the Pauli operators

$$a_i^\dagger = \frac{1}{2}\left(\mathtt{X}_i - i\mathtt{Y}_i\right) \otimes_{j<i} \mathtt{Z}_j \qquad\qquad a_i = \frac{1}{2}\left(\mathtt{X}_i + i\mathtt{Y}_i\right) \otimes_{j<i} \mathtt{Z}_j.$$

Perhaps unfortunately, the Pauli operators corresponding to creation and annihilation operators have weight $\mathcal{O}(n)$ [27]. Since a group of high-weight operators may, in some cases, require more complicated methods to be diagonalized, it could be beneficial to define basis orbitals differently to provide more flexibility. The parity transformation maps the parity of the basis orbitals up to index i to the i^{th} qubit. The resulting creation and annihilation operators are

$$a_i^\dagger = \frac{1}{2}\left(\mathtt{Z}_{i-1} \otimes \mathtt{X}_i - i\mathtt{Y}_i\right) \otimes_{j>i} \mathtt{X}_j \qquad\qquad a_i = \frac{1}{2}\left(\mathtt{Z}_{i-1} \otimes \mathtt{X}_i + i\mathtt{Y}_i\right) \otimes_{j>i} \mathtt{X}_j.$$

These again have weight $\mathcal{O}(n)$, but the leading string of $\mathtt{Z}$s have been traded for a trailing string of $\mathtt{X}$s [29]. For specific Hamiltonians, one approach or the other might prove more beneficial. In comparison to the above transformations, the Bravyi-Kitaev transform manages to avoid the $\mathcal{O}(n)$ scaling with a more complicated mapping, where the information stored by the i^{th} qubit is determined by the binary expansion of i. In particular, $|i\rangle$ stores the parity of the occupation number of orbital i plus the value of $|i - 2^j\rangle$ for all j where the rightmost j digits of the binary expansion of i are all 1 [30]. This complicated mapping results in creation and annihilation operators given by

$$\begin{aligned} a_i^\dagger &= \frac{1}{2}\left(\mathtt{X}_{U(i)} \otimes \mathtt{X}_i \otimes \mathtt{Z}_{P(i)} - i\mathtt{X}_{U(i)} \otimes \mathtt{X}_i \otimes \mathtt{Z}_{P(i)}\right) \\ a_i &= \frac{1}{2}\left(\mathtt{X}_{U(i)} \otimes \mathtt{X}_i \otimes \mathtt{Z}_{P(i)} + i\mathtt{X}_{U(i)} \otimes \mathtt{X}_i \otimes \mathtt{Z}_{P(i)}\right), \end{aligned}$$

where $U(i)$ is the "update set" of qubit i and $P(i)$ is the "parity set" of qubit i. Important for our purposes is the fact that the resulting sets have size at most $\mathcal{O}(\log_2 i)$, resulting in lower-weight Pauli operators than those found in the Jordan-Wigner and parity transformations [27].

Regardless of which transformation is chosen, the output is a qubit Hamiltonian,

$$\mathcal{H} = \sum_{i=1}^{N} c_i \mathtt{P}_i,$$

where $\{\mathtt{P}_i\}_{i=1}^{N}$ are n-qubit Pauli operators and $\{c_i\}_{i=1}^{N}$ are real numbers. Since Pauli operators have eigenvalues ±1, this is indeed a k-local Hamiltonian, as given in Definition 10.

A clear first task might be to ask whether $\mathcal{H}$ can be simplified or reduced in any way. Qubit counts on NISQ devices constrain the sizes of molecules that can be simulated using the VQE, so minimizing the number of qubits required is an important first step. The three transformations map n basis orbitals to n qubits, and in the absence of symmetries, this is optimal. However, for structured Hamiltonians which satisfy symmetries, it should be possible to reduce the number of qubits. This has been studied in the context of the first and second quantization methods, and sparser[1] Hamiltonians with small qubit savings were produced [2].

In Chapter 3, we will expand upon the tapering method described in Ref. [2] and will provide an analysis of the benefits of this expansion when applied to various classes of Hamiltonians. In particular, we will describe instances in which the exponential classical cost of splitting into two eigenspaces can be reduced, and we will discuss promising future directions involving Clifford, rather than Pauli, symmetries.

2.4 Background for Chapter 4

Having simplified the qubit Hamiltonian, $\mathcal{H}$, we are ready now to find the ground state energy using the VQE. One bottleneck in the VQE is the number of measurements required. As described in Definition 2.2.1, the goal of the measurement step is to approximate the expectation value, $\langle\psi(\theta)|\mathcal{H}|\psi(\theta)\rangle$ for some parameterized input state, $|\psi(\theta)\rangle$.

Diagonalizing $\mathcal{H}$, which would be necessary to measure this expectation value directly, is often unfeasible on NISQ devices. If the set of Pauli operators do not all commute with

[1] Here, sparsity is measured by the number of qubitwise commuting parts in a partition of the Pauli operators in the Hamiltonian.

one another, then a non-Clifford circuit would be required. Therefore, we will make use of the linearity of expectation values to rewrite this as

$$\langle\psi(\theta)|\mathcal{H}|\psi(\theta)\rangle = \left\langle \psi(\theta) \middle| \sum_i c_i \mathtt{P}_i \middle| \psi(\theta) \right\rangle = \sum_i c_i \langle\psi(\theta)|\mathtt{P}_i|\psi(\theta)\rangle \,.$$

If a given Pauli, P, is a tensor product of only I and Z terms, then its matrix representation is diagonal with entries in $\{\pm 1\}$. In particular, the expectation value is given by the expression

$$\langle\psi(\theta)|\mathtt{P}|\psi(\theta)\rangle = \sum_k (-1)^{f(k)} \langle\psi(\theta)|k\rangle\langle k|\psi(\theta)\rangle \,,$$

where f is a binary function which can be efficiently calculated based on the locations of I and Z terms in P. The summands, $\langle\psi(\theta)|k\rangle\langle k|\psi(\theta)\rangle$, form a probability distribution and can be estimated by measuring $|\psi(\theta)\rangle$ repeatedly in the computational basis. The same measurements can also be used to estimate all diagonal Pauli operators simultaneously, since the functions, f, can be determined efficiently classically for each individual diagonal Pauli operator.

For non-diagonal operators, we can construct a Clifford circuit, $\mathcal{C}$, such that $\mathcal{C}^\dagger \mathtt{P} \mathcal{C}$ is diagonal. As a result, the state to measure is

$$\left\langle \psi(\theta) \middle| \mathcal{C}^\dagger \mathtt{P} \mathcal{C}^\dagger \middle| \psi(\theta) \right\rangle = \left(\mathcal{C}\,|\psi(\theta)\rangle\right)^\dagger \mathtt{P} \left(\mathcal{C}\,|\psi(\theta)\rangle\right),$$

which can be estimated by measuring $\mathcal{C}\,|\psi(\theta)\rangle$ repeatedly in the computational basis. A single-qubit Clifford circuit can diagonalize a set of qubitwise-commuting[2] Pauli operators. A general Clifford circuit can diagonalize an entire commuting set of Pauli operators, and this Clifford circuit can be efficiently constructed [31].

Therefore, a not-so-naive approach to measurement would be to partition the set of Pauli operators into qubitwise-commuting or general-commuting parts and to measure the parts one at a time. The benefit of using qubitwise commutation is that the measurement circuit will not require entangling gates, the cost of which is discussed further in Chapter 5, but the downside is that there will be more parts in the resulting partition [32].

Once a method of partition has been chosen, there is still plenty of work to be done, since different groupings of Pauli operators and assignments of measurements will result in

[2] Two Pauli operators qubitwise commute if they do not "disagree" on any non-identity qubit. For example, if you compute the symplectic inner product of XX and ZZ, you will find that they commute, but they do not qubitwise commute.

vastly different measurement errors. This has been studied in great detail, and some of their approaches will be discussed in a bit more detail in Chapter 4 [31, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48]. Our result is a measurement scheme called AEQuO, which builds upon much of this prior research, and expands upon it by incorporating a model for efficiently estimating the current measurement error using Bayesian estimates.

2.5 Background for Chapter 5

As alluded to in the second to last paragraph above, the cost of entangling gates on NISQ devices may be prohibitive, which might force researchers to use qubitwise commutation rather than general commutation. In Ref. [32], I addressed this concern by noting an upper bound on the number of entangling gates of $\mathcal{O}(n^2/\log n)$, which is guaranteed by Aaronson and Gottesman's canonical form for a Clifford circuit. At the time, I did not know whether this bound was tight.

In Chapter 5, we will prove a $\Theta(n^2/\log n)$ bound for entangling gates in Clifford circuits, proving that this upper bound is tight for some circuits. This will match the $\Theta(n^2/\log n)$ bound for classical reversible circuits, meaning that allowing for arbitrary single-qubit gates in a CNOT circuit does not necessarily result in asymptotically fewer CNOTs.

Asymptotics aside, this question opens the door for a much deeper investigation into the question of finding the Clifford circuit with the fewest entangling gates. An efficient method for doing so would open the door for all of the measurement schemes above which use general commutation. If $n^2/\log n$ entangling gates is not thought feasible, but n or $n \log n$ gates are, then investigating commuting sets of Cliffords which require asymptotically fewer gates will prove beneficial.

To answer these questions, we will use the graph state formalism, which we will define in Chapter 5. This formalism was first introduced in the use of cluster states for measurement-based quantum computation (MBQC) [49]. MBQC is a promising avenue for near-term quantum algorithms since, after the cluster state is prepared, only single-qubit measurements are required for universality. Graph states provide a visual interpretation of the entanglement contained in a group of commuting Pauli operators, and therefore are highly relevant for our proposed goals. Ref. [50] provides an overview of graph states and their many applications.

In particular, we will be interested in the local-complementation orbits of graph states, which will also be defined in Chapter 5. Local-complementation orbits encompass all groups of Paulis which are equivalent up to single-qubit Clifford circuits. Local complements were first studied in the context of isotropic systems [51]. Ref. [5] enumerates all local-complementation orbits for small numbers of qubits and discusses some applications of graph states to near-term quantum algorithms.

Chapter 3

Hamiltonian Simplification

3.1 Introduction

At this moment, we are seeing tremendous efforts in enhancing quantum hardware while lowering the complexity of the models to be simulated. For instance, approaches such as the variational quantum eigensolver [52, 53, 54, 34, 45, 55, 56], defined in the background section in Definition 2.2.1, rely on understanding and simplifying the Hamiltonian to design tailored and minimal resources for the experiment (see, e.g., Refs. [57, 58, 59, 60, 61] and Refs. [21, 20, 62, 63, 64, 65, 66] for chemistry and lattice gauge theories, respectively). But the search for simplification has many applications besides the VQE, including (besides countless others) quantum machine learning [67, 68, 69, 70], optimization tasks [71, 72, 73, 74], and tensor network techniques [75, 76, 77].

In this chapter, we will review one such simplification technique by the name of qubit tapering [2]. This algorithm eliminates superfluous qubits by determining redundancies and conserved charges of the model. Remarkably, these are found classically, efficiently, and without any prior knowledge. Tapered qubits allow for the parallelization of quantum simulations on both classical and quantum platforms, and individual study of independent subsectors of the Hamiltonian. The techniques presented in this chapter were developed by myself and my collaborators without knowledge of the cited 2017 article [1]. Therefore, the presentation will differ from that given in Bravyi et al.'s article, and we will make special

This chapter closely follows the work presented in a preprint on which I am a first author [1]. Many thanks are owed to my co-first author, Lane Gunderman, and my co-author Luca Dellantonio for their collaboration.

note of differences between our approach and the tapering qubits approach, as well as the future directions which we have begun to explore.

3.2 Main Results

As shown in Figure 3.1*(top)*, we consider an n-qubit system whose dynamics is governed by the Hamiltonian

$$\mathcal{H} = \sum_{i=1}^{N} c_i \mathtt{P}_i, \tag{3.1}$$

where $\{\mathtt{P}_i\}_{i=1}^N$ are n-qubit Pauli operators and $\{c_i\}_{i=1}^N$ are real numbers. To simplify $\mathcal{H}$ into the desired form, the main tool we employ is the Clifford group, under the form of a circuit $\mathcal{C}$ consisting of Hadamard $\mathtt{H}$, phase $\mathtt{S}$, and the entangling CNOT gates (for an example, see bottom panel in the figure).

As noted in Chapter 2, simulating Clifford circuits on classical computers requires only polynomial resources with respect to the number of qubits [7, 78]. Thanks to this fact, our algorithm has execution time $\mathcal{O}(n^2 N)$.

Furthermore, our method is equally applicable to simulations on classical or quantum devices. Indeed, we both provide (see Section 3.3) the Hamiltonian in its simplified form and the circuit $\mathcal{C}$ that can be readily implemented on every quantum setup.

Figure 3.1 illustrates the qubit tapering, with the top panel displaying the algorithmic subroutines on which it is based. Their high-level explanation is outlined in this section, alongside the example in the bottom panel. The main theoretical proofs demonstrating the feasibility and the optimality of our formalism are given in Section 3.3.

We begin by reducing the number of qubits required for exactly representing the full system dynamics. This is the “Redundancy Removal” ($\mathcal{RR}$) in Figure 3.1 (green box), which effectively eliminates the trivial portion of the n-qubits Hilbert space. With the scope of lowering the complexity of a considered problem, similar ideas have been always utilized in physics. For instance, in the case of Fermionic systems with a fixed number of excitations, a minimal qubit representation was found in Ref. [81]. However, this and other works [57, 62, 82] either consider specific (classes of) Hamiltonians, instead of an arbitrary $\mathcal{H}$ as in Eq. (3.1), or were not ensured to be (classically) efficient nor optimal. Qubit tapering is general, efficient *and* optimal (see Theorem 1), implying that we deterministically find the smallest possible subspace of the system Hilbert space to faithfully describe the whole dynamics.

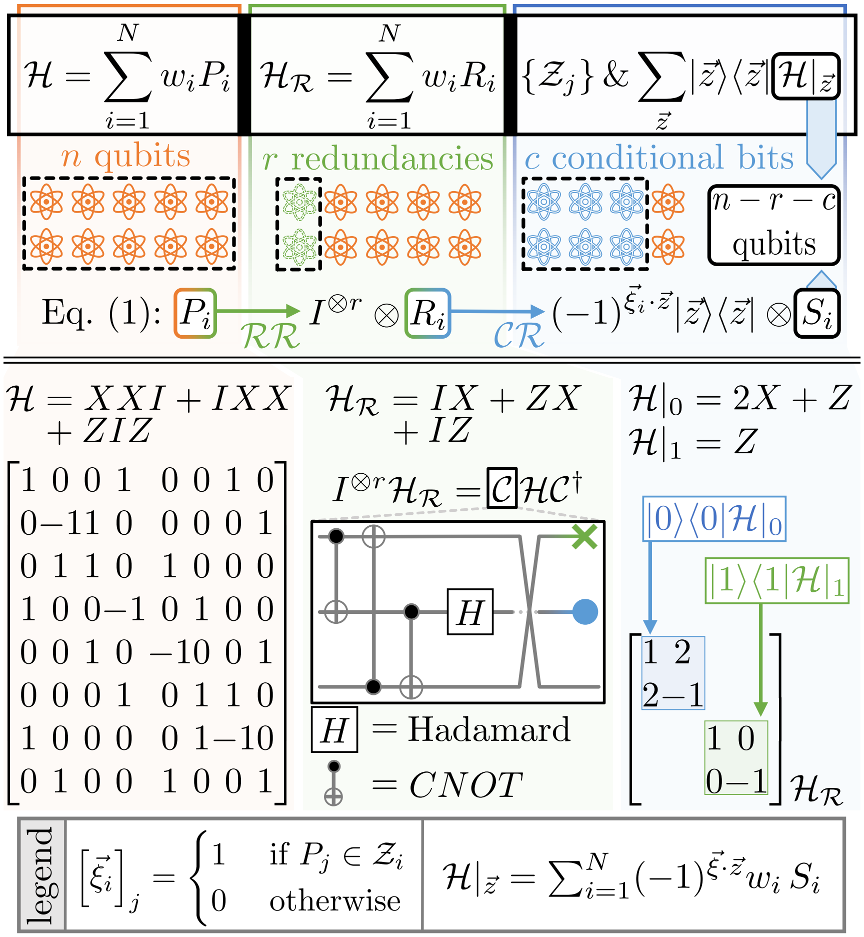

$\mathcal{H} = \sum_{i=1}^{N} w_i P_i$
$\mathcal{H}_\mathcal{R} = \sum_{i=1}^{N} w_i R_i$
$\{\mathcal{Z}_j\}$ & $\sum_{\vec{z}} |\vec{z}\rangle\langle\vec{z}| \mathcal{H}|_{\vec{z}}$
n qubits
r redundancies
c conditional bits
$n-r-c$ qubits
Eq. (1): $P_i \xrightarrow{\mathcal{RR}} I^{\otimes r} \otimes R_i \xrightarrow{\mathcal{CR}} (-1)^{\vec{\xi}_i \cdot \vec{z}} |\vec{z}\rangle\langle\vec{z}| \otimes S_i$
$\mathcal{H} = XXI + IXX + ZIZ$
$\mathcal{H}_\mathcal{R} = IX + ZX + IZ$
$\mathcal{H}|_0 = 2X + Z$
$\mathcal{H}|_1 = Z$
$I^{\otimes r}\mathcal{H}_\mathcal{R} = \boxed{\mathcal{C}}\mathcal{H}\mathcal{C}^\dagger$
H
$|0\rangle\langle 0|\mathcal{H}|_0$
$|1\rangle\langle 1|\mathcal{H}|_1$
$\mathcal{H}_\mathcal{R}$
H = Hadamard
= $CNOT$
legend
$\left[\vec{\xi}_i\right]_j = \begin{cases} 1 & \text{if } P_j \in \mathcal{Z}_i \\ 0 & \text{otherwise} \end{cases}$
$\mathcal{H}|_{\vec{z}} = \sum_{i=1}^{N} (-1)^{\vec{\xi} \cdot \vec{z}} w_i\, S_i$

Figure 3.1: *(top)*: Underlying idea. From left to right, an input $\mathcal{H}$ is given, acting on n qubits. $\mathcal{RR}$ first determines a minimal representation $\mathcal{H}_\mathcal{R}$, equivalent to $\mathcal{H}$ but requiring $n-r$ qubits. Afterwards, $\mathcal{CR}$ finds all conserved charges $\{\mathcal{Z}_j\}_{j=1}^{c}$ of $\mathcal{H}_\mathcal{R}$ (equiv., $\mathcal{H}$). Employing their eigenvectors $|\vec{z}\rangle$, the problem is further simplified to the $n-r-c$ qubits $\mathcal{H}|_{\vec{z}}$. The whole process is well represented by the operators $\mathtt{P}_i$, $\mathtt{R}_i$ and $\mathtt{S}_i$ in the Hamiltonians $\mathcal{H}$, $\mathcal{H}_\mathcal{R}$ and $\mathcal{H}|_{\vec{z}}$, respectively. *(bottom)*: Example from a modified $J_1 - J_2$ model with XX (ZZ) interactions between (next) neighbouring spins [79, 80] and $J_1 = J_2 = 1$. Via the Clifford circuit $\mathcal{C}$ consisting of $\mathtt{H}$, CNOT and a final SWAP gates, the input matrix is both reduced dimensionally and made block-diagonal, with blocks describing $n-r-c=1$ qubit. Qubits removed via $\mathcal{RR}$ and $\mathcal{CR}$ are indicated in $\mathcal{C}$ with the green cross and blue circle, respectively.

In more details, $\mathcal{RR}$ works as follows.[1] As schematically represented in Figure 3.1, we determine a Clifford circuit $\mathcal{C}$ such that $\mathcal{C}\mathtt{P}_i\mathcal{C}^\dagger = (\bigotimes_{k=1}^{r} I) \otimes \mathtt{R}_i$ for all $i = 1, \ldots, N$. Consequently, the first r qubits become negligible and the system Hamiltonian in Eq. (3.1) can be represented by $\mathcal{H} \xrightarrow{\mathcal{RR}} \mathcal{H}_{\mathcal{R}} = \sum_{i=1}^{N} \omega_i \mathtt{R}_i$.

An example of the $\mathcal{RR}$ subroutine is given in Figure 3.1*(bottom)*. We consider $\mathcal{H} = \mathtt{XXI} + \mathtt{IXX} + \mathtt{ZIZ}$ on $n = 3$ qubits, and employ our protocol to determine the Clifford circuit $\mathcal{C}$ in the figure. None of the terms in the resulting $\mathcal{C}\mathcal{H}\mathcal{C}^\dagger = \mathtt{IIX} + \mathtt{IZX} + \mathtt{IIZ}$ acts on the first $r = 1$ qubit, such that the whole system dynamics is described via the operator $\mathcal{H}_{\mathcal{R}}$ defined on a reduced Hilbert space of $n - r = 2$ qubits.

Our second reduction, called "Conditional Removal" ($\mathcal{CR}$), leverages conserved charges. These, indicated with $\{\mathcal{Z}_j\}_{j=1}^{c}$ in Figure 3.1, are the c Pauli symmetries of the Hamiltonian $\mathcal{H}_{\mathcal{R}}$, meaning that for each $j = 1, \ldots, c$, $[\mathcal{Z}_j, \mathcal{H}_{\mathcal{R}}] = 0$ and $\mathcal{Z}_j$ is a tensor product of $n - r$ Pauli operators. Importantly, tapering qubits ensures that these conserved charges fulfil two properties. First, $\mathcal{CR}$ finds *all* of them, meaning that *any* Pauli symmetry of $\mathcal{H}_{\mathcal{R}}$ can be written as a linear combination of the $\{\mathcal{Z}_j\}_{j=1}^{c}$. Second, that we can express the conserved charges in the practical form

$$\mathcal{Z}_j = \underbrace{\mathtt{I} \otimes \mathtt{I} \cdots \otimes \overbrace{\mathtt{Z}}^{j^{\text{th}}} \otimes \cdots \otimes \mathtt{I}}_{c \text{ qubits}} \bigotimes_{i=1}^{n-r-c} \mathtt{I}, \tag{3.2}$$

i.e., in terms of c diagonal, independent generators with a single $\mathtt{Z}$ operator each. This form for the conserved charges can be chosen since they must commute with each other and as such they can be efficiently diagonalized [31, 4].

Expressing the conserved charges $\mathcal{Z}_j$ as in Eq. (3.2) is not necessary. To better understand the symmetries $\{\tilde{\mathcal{Z}}_j\}_{j=1}^{c}$ of $\mathcal{H}$ in Eq. (3.1) one can revert the basis change characterized by $\mathcal{C}$ to find that $[\tilde{\mathcal{Z}}_j, \mathcal{H}] = 0$ for $\tilde{\mathcal{Z}}_j \equiv \mathcal{C}^\dagger((\bigotimes_{i=1}^{r} \mathtt{I}) \otimes \mathcal{Z}_j)\mathcal{C}$.

Furthermore, an interesting question is whether it is feasible to lower the complexity of simulating $\mathcal{H}_{\mathcal{R}}$ by changing the form of the conserved charges (e.g., concentrating the system entanglement in the c qubits onto which they act).

In the context of this chapter and the $\mathcal{CR}$ subroutine specifically, we express the conserved charges as in Eq. (3.2) for clarity and functionality. In fact, in this form the computational basis $|\vec{z}\rangle$ with $\vec{z} = \{0,1\}^c$ (e.g., $\vec{z} = 0111001$ for $c = 7$) contains all eigenvectors of the conserved charges $\{\mathcal{Z}_j\}_{j=1}^{c}$, and therefore of the subset of corresponding c qubits of the $\{\mathtt{R}_i\}_{i=1}^{N}$ operators within $\mathcal{H}_{\mathcal{R}}$ (see Figure 3.1). Even better, we can determine the associated

[1] Note that a subroutine similar to $\mathcal{RR}$ was previously investigated in Ref. [3]

charge eigenvalues $(-1)^{\vec{z}\cdot\vec{\zeta}_i}$ and consequently divide $\mathcal{H}_\mathcal{R}$ into 2^c separated, non-interacting subsectors, each governed by its own Hamiltonian $\mathcal{H}|_{\vec{z}}$

$$\mathcal{H}_\mathcal{R} = \sum_{\vec{z}} |\,\vec{z}\rangle\langle\vec{z}\,| \otimes \overbrace{\sum_{i=1}^{N}(-1)^{\vec{z}\cdot\vec{\zeta}_i}\omega_i\mathtt{S}_i}^{\mathcal{H}|_{\vec{z}}}, \tag{3.3a}$$

$$\left[\vec{\zeta}_i\right]_j = 1 \text{ if } \mathcal{Z}_j \in \mathtt{R}_i,\ \ 0 \text{ otherwise.} \tag{3.3b}$$

Here, "$\cdot$" is the dot product modulo 2, $\left[\vec{\zeta}_i\right]_j$ is the j-th component of $\vec{\zeta}_i$ and $\mathcal{Z}_j \in \mathtt{R}_i$ indicates that $\mathtt{R}_i$ has nontrivial support from $\mathcal{Z}_j$. The $\mathtt{S}_i$ are the last $n-r-c$ Pauli terms within $\mathtt{R}_i$ [see Figure 3.2*(right)*].

The physical intuition behind Eqs. (3.3) is the following. After $\mathcal{RR}$ and the removal of r redundant qubits, the trivial dynamics has been fully eliminated. Yet, via $\mathcal{CR}$, it is still possible to efficiently analyze c out of the remaining $n-r$ qubits via the conserved charges $\{\mathcal{Z}_j\}_{j=1}^{c}$. Practically, this is done by collapsing the system dynamics onto their eigenvectors $|\vec{z}\rangle$ [hence the projectors in Eq. (3.3a)] and via the corresponding sub-Hamiltonians $\mathcal{H}|_{\vec{z}}$. These are determined from $\mathcal{H}_\mathcal{R}$ by writing its constituents $\mathtt{R}_i$ as $\mathtt{R}_i = \sum_{\vec{z}}(-1)^{\vec{z}\cdot\vec{\zeta}_i}\,|\,\vec{z}\rangle\langle\vec{z}\,|\otimes\mathtt{S}_i$.

Going back to the example in Figure 3.1, above we employed $\mathcal{RR}$ to describe the system via the $n-r=2$ qubits Hamiltonian $\mathcal{H} \xrightarrow{\mathcal{RR}} \mathcal{H}_\mathcal{R} = \mathtt{IX}+\mathtt{ZX}+\mathtt{IZ}$. As shown in the figure, our algorithm ensures (via the Clifford circuit $\mathcal{C}$) that the matrix form of $\mathcal{H}_\mathcal{R}$ is block diagonal. These blocks, each of size $2^{n-r-c}=2$, encode the $n-r-c=1$ qubit Hamiltonians $\mathcal{H}|_{\vec{z}}$ corresponding to the eigenvectors $|\vec{z}\rangle = |0\rangle$ and $|\vec{z}\rangle = |1\rangle$ of the single charge $\mathcal{Z}_1 = Z\otimes I$. Therefore, the $\mathcal{CR}$ subroutines permits rewriting $\mathcal{H}_\mathcal{R} \xrightarrow{\mathcal{CR}} |\,0\rangle\langle 0\,|\,[\mathtt{X}+\mathtt{X}+\mathtt{Z}]+|\,1\rangle\langle 1\,|\,[\mathtt{X}-\mathtt{X}+\mathtt{Z}]$, where the sign of each term in the square parentheses depends on the charge eigenvalue of $|\vec{z}\rangle$ and whether $\mathcal{Z}_1 \in \mathtt{R}_i$.

Concluding, Eqs. (3.3) are the mathematical formulation of qubit tapering applied to qubit-based physical models. They tell us that when conserved charges are present, simulating the system as a whole is not necessary. We can restrict ourselves to smaller subsectors that operate non-trivially on fewer qubits. Furthermore, knowing the conserved charges allows the identification of the subsectors describing the dynamics of interest. For instance, only the case of zero total electrical charge is usually investigated in $\mathbb{Z}_2$ models of lattice gauge theory (see Section 3.5), making most of the Hilbert space irrelevant. Tapering qubits allows determining the Hamiltonian describing the corresponding dynamics, resulting in a significant reduction of the simulation complexity.

Finally, even in the scenario in which all 2^c subsectors must be studied, tapering qubits allows for complexity reduction and/or parallelization of quantum processes, as each of the $\mathcal{H}|_{\vec{z}}$ Hamiltonians can be individually addressed. In the classical case, diagonalization algorithms require runtimes $O(2^{\alpha n})$, with $2 \leq \alpha \leq 3$ [83, 84, 85]. Therefore, (without) studying the 2^c Hamiltonians $\mathcal{H}|_{\vec{z}}$ in parallel, our method allows reducing the total time from $2^{\alpha n}$ to $2^{\alpha(n-r-c)}$ $(2^c \cdot 2^{\alpha(n-r-c)})$, or by a factor of $2^{\alpha(r+c)}$ $(2^{\alpha r+(\alpha-1)c})$. In the quantum case, on the other hand, Eqs. (3.3) imply that all eigenstates of $\mathcal{H}_{\mathcal{R}}$ are such that the first c qubits are in the computational basis. By taking the variational quantum eigensolver [52, 53, 54, 34, 45, 55] as an example, this means that these qubits only require local gates to reach the groundstate, which provides likely reductions in circuit complexity and depth.

3.3 Algorithm

The algorithm is described in Figure 3.2. We represent $\mathcal{H}$ in Eq. (3.1) as an $N \times n$ tableau [2] with entries $\mathtt{I}$, $\mathtt{X}$, $\mathtt{Y}$ and $\mathtt{Z}$ [see Figure 3.2*(right)*], where the i-th row corresponds to $\mathtt{P}_i$ and the k-th column to the qubit. Therefore, the (i, k) element $\mathtt{P}_i|_k$ is the Pauli operator acting on the k-th qubit of $\mathtt{P}_i$.

With $\mathcal{H}$ written as a tableau, one can identify the set of standard operations corresponding to Clifford gates (see Section 3.4 and Ref. [78]). These, in turn, are employed in the circuits $\mathcal{C}_k^{\mathrm{z}}$ and $\mathcal{C}_k^{\mathrm{x}}$ used in the k-th iteration of the algorithm to build $\mathcal{C} = \prod_k \mathcal{C}_k^{\mathrm{x}} \mathcal{C}_k^{\mathrm{z}}$ [see Figure 3.2*(left)*]. $\mathcal{C}$ is the Clifford circuit required by the $\mathcal{RR}$ subroutine to determine $\mathcal{H}_{\mathcal{R}}$. This is deduced from the two blue boxes in the figure, explaining that $\mathcal{C}$ is constructed to ensure only identities act on the first r qubits, while $\mathtt{X}$ and $\mathtt{Y}$ are collected in the last $n - c - r$, leaving the c in the middle with only $\mathtt{I}$ or $\mathtt{Z}$ [3].

Our algorithm in Figure 3.2*(left)* always ensures $\mathcal{H}_{\mathcal{R}}$ is block diagonal, as its first c qubits are acted upon by diagonal operators. To find the form of the block $\mathcal{H}|_{\vec{z}}$, the $\mathcal{CR}$ subroutine first determines the $\{\mathtt{S}_i\}_{i=1}^{N}$ from the last $n - r - c$ column of the tableau in Figure 3.2*(right)*. Then, depending on the chosen $\vec{z}$, the corresponding signs are calculated from the c columns with only $\mathtt{I}$ and $\mathtt{Z}$.

With more details presented in Section 3.4, we demonstrate here the optimality of this

[2] As better explained in Section 3.3 and in Section 3.4, our algorithm does not work with all N elements of the input Hamiltonian $\mathcal{H}$. Instead, it employs a generating subset that can be much smaller in practice, such that the efficiency is enhanced.

[3] Additional $SWAP$ operations may be required. For clarity, we did not explicitly include those in the algorithm.

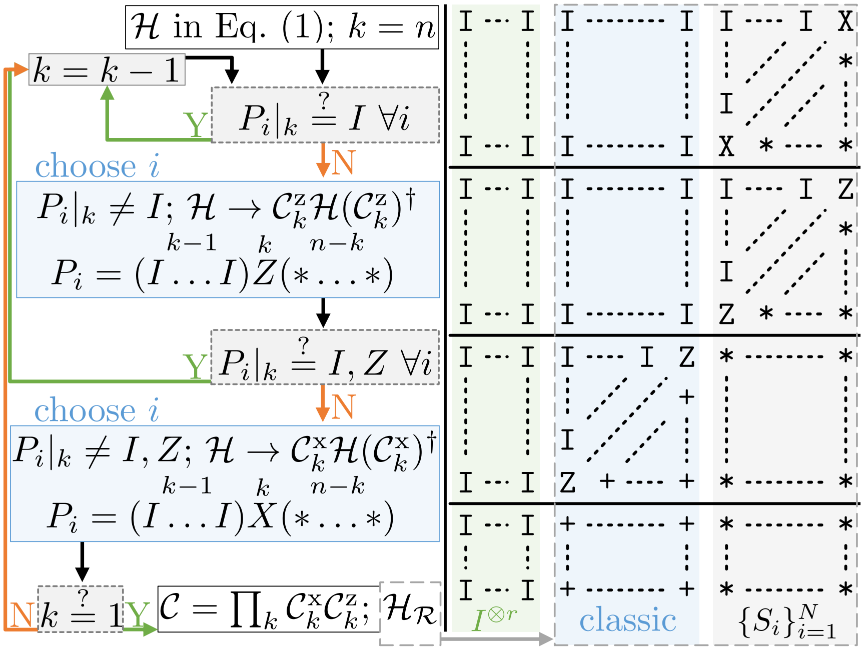

$\mathcal{H}$ in Eq. (1); $k = n$
$k = k - 1$
$P_i|_k \overset{?}{=} I\ \forall i$
Y
N
choose i
$P_i|_k \neq I$; $\mathcal{H} \to \mathcal{C}^{\mathrm{z}}_k \mathcal{H}(\mathcal{C}^{\mathrm{z}}_k)^\dagger$
$P_i = (\overset{k-1}{I \dots I})\overset{k}{Z}(\overset{n-k}{* \dots *})$
$P_i|_k \overset{?}{=} I, Z\ \forall i$
Y
N
choose i
$P_i|_k \neq I, Z$; $\mathcal{H} \to \mathcal{C}^{\mathrm{x}}_k \mathcal{H}(\mathcal{C}^{\mathrm{x}}_k)^\dagger$
$P_i = (\overset{k-1}{I \dots I})\overset{k}{X}(\overset{n-k}{* \dots *})$
N
$k \overset{?}{=} 1$
Y
$\mathcal{C} = \prod_k \mathcal{C}^{\mathrm{x}}_k \mathcal{C}^{\mathrm{z}}_k$; $\mathcal{H}_{\mathcal{R}}$
$I^{\otimes r}$
classic
$\{S_i\}_{i=1}^N$

Figure 3.2: *(left)*: An overview of the algorithm. From the input $\mathcal{H}$ and for each qubit $k = n, n-1, \dots, 1$, in the first [second] step we iteratively check (gray boxes with dashed lines) whether all $\mathtt{P}_i$ acting on the k-th qubits are all $\mathtt{I}$ [$\mathtt{I}$ or $\mathtt{Z}$]. If they are, we successfully identified a redundant [conditional] qubit and proceed. If they are not, we pick a $\mathtt{P}_i$ such that $\mathtt{P}_i|_k \neq \mathtt{I}$ [$\mathtt{P}_i|_k \neq \mathtt{I}, \mathtt{Z}$] and build (lightblue boxes) the $\mathcal{C}^{\mathrm{z}}_k$ [$\mathcal{C}^{\mathrm{x}}_k$] that ensures $\mathtt{P}_i|_j = \mathtt{I}$ for all $j = 1, \dots, k-1$, and $\mathtt{P}_i|_k = \mathtt{Z}$ [$\mathtt{P}_i|_k = \mathtt{X}$]. The outputs of the algorithm are $\mathcal{C}$ and $\mathcal{H}_{\mathcal{R}}$. This process ensures that $\mathcal{H}_{\mathcal{R}}$ is as in the *(right)* panel, i.e., the first r qubits are only acted upon by $\mathtt{I}$ and the next c by $\mathtt{I}$ or $\mathtt{Z}$. The operators $\mathtt{S}_i$ are determined from the last $n - r - c$ column of the tableau, $+$ and $*$ are used to indicate $\mathtt{I}$ or $\mathtt{Z}$, and any Pauli operator, respectively. The coefficients (omitted for clarity) are tracked with their associated Pauli operator.

algorithm. I.e., both the reduction of r redundant qubits and the number c of conserved charges are the maximal ones. This is stated by the following

Theorem 1 *Let $\mathcal{H}$ be as in Eq. (3.1). For any basis of $\{\mathtt{P}_i\}_{i=1}^N$, let M be the corresponding anticommutation matrix. Then, for any non-negative $c \leq \dim(M) - \operatorname{rank}(M)$, $\mathcal{H}$ can be simulated using $\frac{1}{2}\operatorname{rank}(M) + [\dim(M) - \operatorname{rank}(M) - c]$ qubits and a maximum of 2^c subproblems.* □

A basis of $\{\mathtt{P}_i\}_{i=1}^N$ is a product-wise independent collection of elements such that any of the $\mathtt{P}_i$ can be expressed as a product of elements in the collection. The element M_{ij} of M is one if $[\mathtt{P}_i, \mathtt{P}_j] \neq 0$, zero otherwise, and the 2^c subproblems are (e.g.) the $\mathcal{H}|_{\vec{z}}$ defined above, that generally depend on the the chosen generating subset (albeit their size does not). $\dim(M)$ and $\operatorname{rank}(M)$ are the dimension and the rank (computed over the field of integers modulo 2) of M, respectively.

The detailed proofs of Theorem 1 and the optimality of the algorithm in Figure 3.2 are extensions of the result presented in Bravyi et al.'s article [2]. Intuitively, the first relies on symplectic linear algebraic subspaces, while the latter follows from the properties of the resulting $\mathcal{H}|_{\vec{z}}$. Indeed, they saturate the bound $c = \dim(M) - \operatorname{rank}(M)$ by construction, i.e., their size is the minimal possible one that is achieved via optimal $\mathcal{RR}$ and $\mathcal{CR}$ processes. It is likewise true of the qubit tapering approach that all Pauli symmetries are found by the row reduction algorithm, so here we extend the result by giving formulas based on the anticommutation matrix to determine number of tapered qubits.

3.4 Proof of Optimality

We will use the symplectic representation of Pauli operators, given in Definition 7 [12, 8, 86, 87]. Here, we discuss how the commutation relations between a basis of the Pauli operators can be used to identify the number of qubits which can be tapered off. The result of transforming the matrix of commutation relations represents a set of Pauli operators which are related to the original ones through a Clifford circuit, which is classically easy to reproduce. The techniques used here are equivalent to those used in [3], however, the conclusions are taken further here. Gunderman's article only quantifies the number redundant qubits as $n - (\dim(M) - \frac{1}{2}\operatorname{rank}(M))$. Here, we go on to quantify the number of conserved charges as $\dim(M) - \operatorname{rank}(M)$, resulting in sub-Hamiltonians on only $\frac{1}{2}\operatorname{rank}(M)$ qubits.

Let $\mathcal{P}$ be the given collection of Pauli operators. The elements of $\mathcal{P}$ can be generated by compositions of generators, $\mathcal{G}$, of the smallest subgroup of Pauli operators which contains $\mathcal{P}$. Then define the anticommutation matrix $M(\mathcal{G})$ as:

$$M(\mathcal{G})_{ij} = \mathcal{G}_i \odot \mathcal{G}_j, \tag{3.4}$$

where $\odot$ is the symplectic inner product given in Definition 8 [12, 88].

The anticommutation matrix itself is not invariant under the selection of the generators in $\mathcal{G}$. Replacing generator $\mathcal{G}_1$ with $\mathcal{G}_1 \circ \mathcal{G}_2$ is equivalent to adding the row corresponding to $\mathcal{G}_2$ to the row corresponding to $\mathcal{G}_1$, and likewise adding the same column. This forms a symmetric Gaussian elimination technique and so the *rank* of $M(\mathcal{G})$ is unaltered under the selection of generators.[4]

Using this, there is a set of generators $\mathcal{D}$ such that:

$$M(\mathcal{D}) = \left(\bigoplus_{i=0}^{dim(M)-rank(M)} [0] \right) \bigoplus \left(\bigoplus_{i=0}^{\frac{1}{2}rank(M)} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right). \tag{3.5}$$

This can be satisfied by generators $\mathtt{Z}_i$ for the first $\dim(M) - \operatorname{rank}(M)$ qubits, and anticommuting pairs $\{\mathtt{X}_i, \mathtt{Z}_i\}$ for the remaining qubits. This means that to represent the commutation relations for $\mathcal{P}$, only $\dim(M(\mathcal{G})) - \frac{1}{2}\operatorname{rank}(M(\mathcal{G}))$ qubits are needed. It is not possible to use fewer qubits than this and still be Clifford equivalent to the original collection $\mathcal{P}$. This means that if the original collection $\mathcal{P}$ operated on $n > \dim(M(\mathcal{G})) - \frac{1}{2}\operatorname{rank}(M(\mathcal{G}))$ qubits, the surplus qubits can be turned into identity operators. We call these *redundant* qubits.

While the above provides the number of qubits that are needed to represent the commutation relations represented by $\mathcal{P}$, while still retaining Clifford equivalence, it does not immediately provide Clifford equivalent Pauli operators to those in $\mathcal{P}$ itself. For that, select a generating set for $\mathcal{P}$, $\mathcal{G}(\mathcal{P})$, then $M(\mathcal{G}(\mathcal{P}))$ will be a binary symmetric matrix of commutation values of elements from $\mathcal{P}$. Relating $M(\mathcal{G}(\mathcal{P}))$ to $M(\mathcal{D})$ will then provide a way to transform the minimal qubits from $M(\mathcal{D})$ into ones that are Clifford equivalent to $\mathcal{P}$. As remarked before, composition of generators is equivalent to a simultaneous row and column addition in the commutator matrix, which forms our elementary matrix operations for a symmetric Gaussian elimination. Then there is a sequence of these operations, whose product of row operations is L, such that:

$$M(\mathcal{G}(\mathcal{P})) = LM(\mathcal{D})L^T. \tag{3.6}$$

[4] Formally the rank is computed for the matrix over the field of integers modulo 2

This then provides a minimal qubit representation for the commutation relations expressed in $\mathcal{P}$ itself. This technique is efficient as each step is simply Gaussian elimination. At the end of the procedure some qubits are left as identity operators which may be dropped in circuits. To determine the Clifford circuit mapping from the original Pauli operators to these minimal qubit representations, one may apply the results of [7, 78]. The prior result provides the solution to these methods, for which these methods then find a satisfying Clifford circuits.

We may further extend the observations to condition on separably measurable qubits. This leads to Theorem 1, which states that for simulating a given $\mathcal{H}$ only $\frac{1}{2}rank(M)$ qubits are required, although at a classical repetition cost. To prove this we approach the problem through a broad symplectic linear algebraic perspective. In essence, we quotient out the isotropic subspace, stratifying by possible measurement outcomes for that space, then return to a Pauli representation with updated weights. Each of the remaining subspaces may have different coefficients, and so without further assumptions will need to be simulated.

Proof Let $\mathbf{P}$ be the group generated by $\{\mathtt{P}_i\}$ with d generators for $\mathbf{P}$. Within $\mathbf{P}$ there is an ordered list of generators such that the anticommutation matrix for this collection takes the form of Equation 3.5. Let this set of generators with anticommutation matrix, $\tilde{D}$, be given by $\tilde{g}$. Then the isotropic generators (those which contribute [0] terms to the direct sum) are Clifford equivalent to $\mathtt{Z}_i$ single qubit Paulis in the minimal qubit form. Upon reconstructing minimal qubit versions for $\{\mathtt{P}_i\}$ from the generators $\{\mathtt{Z}_i\}$ and $\{\mathtt{X}_i, \mathtt{Z}_i\}$, these $\mathtt{Z}_i$ will still commute with these Pauli operators, and moreover qubit-wise commute. This then means that we may measure these Pauli operators and update the $\mathtt{P}_i$ constructed by quotienting out by $\mathtt{Z}_i$ and updating the coefficient based on the measurement value. From the direct sum decomposition, we will have $(\dim(M) - \text{rank}(M))$ $\mathtt{Z}_i$ terms which can be replaced with their eigenvalues ±1 and $\frac{1}{2}\,\text{rank}(M)$ pairs of generators which cannot. This will require $\frac{1}{2}\,\text{rank}(M)$ qubits to represent these Paulis, and one may select not to measure all these $\mathtt{Z}_i$, so our result is shown. ■

It's worth noting that inherently the matrix $M(\mathcal{G}(\mathcal{P}))$, following the procedure, automatically separates out the $\mathtt{Z}$ only qubits as these qubits only consist of $\mathtt{Z}$ operators from the bases we selected for the matrix $M(\mathcal{D})$. Above, we also describe this process from an algorithmic Clifford group perspective. This reduction is represented by the following symplectic matrix representation:

$$\left(\begin{array}{cccc|cccc|ccc|cccc|cccc|ccc}
1 & 0 & \cdots & 0 & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 \\
\ast & \ddots & \ddots & \vdots & \vdots & & & \vdots & \vdots & & \vdots & \ast & \ddots & \ddots & \vdots & \vdots & & & \vdots & \vdots & & \vdots \\
\vdots & \ddots & \ddots & 0 & \vdots & & & \vdots & \vdots & & \vdots & \vdots & \ddots & \ddots & 0 & \vdots & & & \vdots & \vdots & & \vdots \\
\ast & \cdots & \ast & 1 & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 & \ast & \cdots & \ast & 0 & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 \\
\hline
0 & 0 & \cdots & 0 & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 \\
\ast & \ddots & \ddots & \vdots & \vdots & & & \vdots & \vdots & & \vdots & \ast & \ddots & \ddots & \vdots & \vdots & & & \vdots & \vdots & & \vdots \\
\vdots & \ddots & \ddots & 0 & \vdots & & & \vdots & \vdots & & \vdots & \vdots & \ddots & \ddots & 0 & \vdots & & & \vdots & \vdots & & \vdots \\
\ast & \cdots & \ast & 0 & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 & \ast & \cdots & \ast & 1 & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 \\
\hline
\ast & \cdots & \cdots & \ast & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 & \ast & \cdots & \cdots & \ast & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
\vdots & & & \vdots & \vdots & & & \vdots & \vdots & & \vdots & \vdots & & & \vdots & \ast & \ddots & \ddots & \vdots & \vdots & & \vdots \\
\vdots & & & \vdots & \vdots & & & \vdots & \vdots & & \vdots & \vdots & & & \vdots & \vdots & \ddots & \ddots & 0 & \vdots & & \vdots \\
\ast & \cdots & \cdots & \ast & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 & \ast & \cdots & \cdots & \ast & \ast & \cdots & \ast & 1 & 0 & \cdots & 0 \\
\hline
\ast & \cdots & \cdots & \ast & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 & \ast & \cdots & \cdots & \ast & \ast & \cdots & \cdots & \ast & 0 & \cdots & 0 \\
\vdots & & & \vdots & \vdots & & & \vdots & \vdots & & \vdots & \vdots & & & \vdots & \vdots & & & \vdots & \vdots & & \vdots \\
\ast & \cdots & \cdots & \ast & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 & \ast & \cdots & \cdots & \ast & \ast & \cdots & \cdots & \ast & 0 & \cdots & 0
\end{array}\right)$$

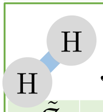
Without further assumptions on the structure of the elements of $\mathcal{P}$, no further reductions in the number of qubits needed may be made without losing some of the non-commuting properties.

3.5 Results

In Figure 3.3 we present the results from the qubit tapering algorithm applied to several physical models. In the left, we consider chemistry Hamiltonians (equiv. in the JW, BK or Parity encoding [22, 89, 90]) and empirically show that, albeit molecule dependent, the number of conserved charges scales as a constant with respect to the molecule size. This is unsurprising, as molecules generally have limited symmetries associated to conserved charges [91]. These results align closely with those in Ref. [2], with slight variation due to a difference in the chosen parameters defining the molecular Hamiltonians. In previous works [59, 55, 92, 93, 94], some were employed to simulate small molecules on quantum computers. Specifically, finding the reduced Hamiltonians of H_2 [59] and LiH [55] required significant efforts. The qubit tapering approach substitutes a general formula in place of these problem-specific efforts.

In the right panels of the figure we investigate the $\mathbb{Z}_2$ [54, 53, 20] (violet), Hubbard [89] (green), Kitaev (blue) and Kitaev with magnetic field [97, 98] (orange) models. The smallest lattices we employed are at the bottom, where we indicate whether qubits lie on

Molecule	n	c
H_2	4	3
H_2	8	3
LiH	12	4
BeH_2	14	5
CH_4	18	3
C_2H_2	24	4
C_2H_4	28	5
C_2H_6	32	4



H_2, 6-31G
JW encoding

$\tilde{\mathcal{Z}}_1$	$ZZZZIIII$
$\tilde{\mathcal{Z}}_2$	$IIIIZZZZ$
$\tilde{\mathcal{Z}}_3$	$IZIZIZIZ$

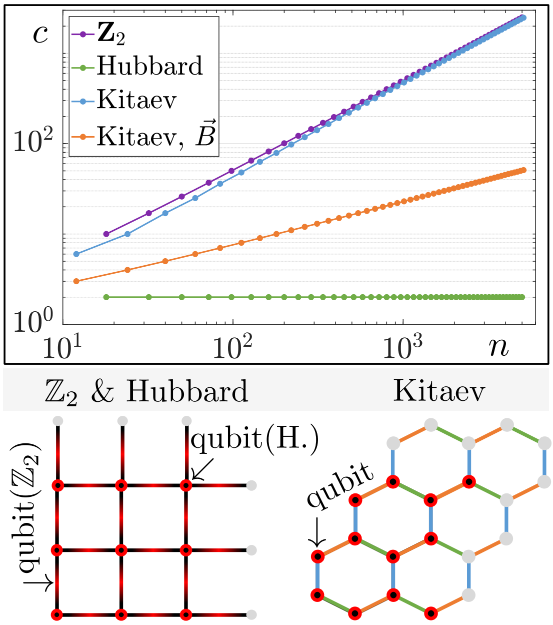


Figure 3.3: Values of c for *(left)* chemistry Hamiltonians in Refs. [95, 90] and *(right)* the $\mathbb{Z}_2$ [54, 53, 20, 96], Hubbard [89] and Kitaev (with and without magnetic field $\vec{B}$) [97, 98] models – additional information in Appendix A. Below the table, we write the conserved charges for H_2 (sketch) in the 6-31G basis and JW encoding. For the other molecules we employ the STO-3G basis. Below the plot, we draw the smallest instances of the square and hexagonal lattices employed. Grey vertices indicate periodic boundary conditions. See Table 3.1 for data. Models with higher n are obtained by scaling up the plaquettes/hexagons in both dimension at the same time.

vertices (Hubbard and Kitaev) or edges ($\mathbb{Z}_2$). Larger instances of the models are obtained by increasing the number of plaquettes/hexagons in both dimensions at the same time (for more details, see Appendix A).

From the plot we deduce that the Hubbard model has a constant number $c = 2$ of conserved charges, corresponding to global properties (see Appendix A). For $\mathbb{Z}_2$, describing (discretized) cavity electromechanics on a torus, our algorithm efficiently implements all $c = n/2 + 1$ Gauss laws and determines the Hamiltonians with any configuration of static charges [21]. For the Kitaev honeycomb lattice [97], when a magnetic field $\vec{B}$ is present, we still find $c = (\sqrt{2n+1} + 1)/2$ conserved charges that can be exploited to both better understand the model and significantly enhance ongoing classical and quantum simulations of the system [97]. Without $\vec{B}$, an analytical solution of the Kitaev honeycomb lattice is known [98], however our algorithm yields $n - c \neq \mathcal{O}(1)$. This is because not all the symmetries correspond to conserved charges. As better discussed in Section 3.7, there is a clear road map to extend our method to identify and exploit more symmetries and thus further simplify input models.

3.6 Conclusions

In this chapter, we have presented our interpretation of the qubit tapering approach. Here we will discuss the similarities and differences between our work and the work of Bravyi et al.

Bravyi et al. described a method for computing conserved charges of a Hamiltonian. Their work encompasses the $\mathcal{CR}$ subroutine, and they use a similar mapping of the symplectic matrix to form similar to reduced row-echelon form. Bravyi et al. discuss how to back-solve for a Clifford circuit, $\mathcal{C}$, which maps $\mathcal{H}$ to their reduced form.

Left unsaid in Bravyi et al.'s description of the algorithm is the fact that, if two anticommuting Pauli operators are both conserved charges of the state, then a qubit can be entirely removed rather than conditioned upon. This is an important, though minor, point, since the ability to simulate only a single sub-Hamiltonian halves the classical cost. For molecular Hamiltonians and the other families of Hamiltonians we tested, though, conserved charges did not appear in such anticommuting pairs.

Similarly, Bravyi et al. do not discuss the optimality of their approach. Though it is strongly implied by the algorithmic reduction to a reduced row-echelon-like form, the proof of optimality, which uses properties of the anticommutation matrix of the Paulis operators in the Hamiltonian, provides a method for more efficiently computing the number of conserved charges without determining the Clifford circuit. We have similarly extended their analysis of molecules to include larger families, and for the first time have found families which benefit from a number of conserved charges which scales proportionally to the number of qubits.

We used the efficient simulation of Clifford circuits provided by the Gottesman-Knill theorem [7, 78] to find the Clifford equivalent Hamiltonian that requires the smallest number of qubits to simulate the full system dynamics. We showcased our method with chemical molecules, $\mathbb{Z}_2$ cavity electrodynamics, the Hubbard and the Kitaev (with and without a magnetic field) models. In all cases, we were able to find symmetries that could be leveraged to lower the complexity of classical and quantum simulations.

From Figure 3.3, we deduce that the numbers of conserved charges in the Hubbard and Kitaev (without $\vec{B}$) models are less than the known symmetries of these systems. In some cases (e.g., Ref. [98]), it is even possible to find an analytical solution. Generally, this requires significant efforts that we want to eliminate via our automatized framework. To do so, we hope to extend these results to include Clifford symmetries, which we describe in a bit more detail in Section 3.7.

We would be remiss not to note, as we summarize the conclusions of this approach, that classical and quantum simulations are not the only avenues where this method is applicable. Minimizing the number of entangling gates for diagonalizing commuting operators [31], efficient Trotterization [99, 100], optimal disentangling operations [50], measurement protocols [4] and circuit synthesis [101, 102] will also benefit from this framework. To facilitate the advent of useful quantum computation, the theory must provide simplified models that are adapted to the available hardware. The Clifford group allows for this adaptability, yet there are exponentially many ways to build n-qubit Clifford circuits. This is a first step towards understanding how to efficiently build these circuits, with a clear path to improvement.

3.7 Clifford Symmetries

Without going into excruciating detail about ongoing work, we will discuss why we believe simple Clifford symmetries to be a promising avenue for tapering off additional qubits. Consider, for example, the Hamiltonian described by:

$$\mathcal{H} = c_1\texttt{XX} + c_1\texttt{ZX} + c_2\texttt{XZ} + c_2\texttt{ZZ} + c_3\texttt{IX} + c_4\texttt{IZ}.$$

Applying the qubit tapering approach described in this chapter will result in no conserved charges, so there are no Pauli symmetries to be found. However, if we map this Hamiltonian under conjugation by a Hadamard gate on the first qubit, the resulting Hamiltonian is:

$$\mathtt{H}_1\mathcal{H}\mathtt{H}_1^\dagger = c_1\mathtt{ZX} + c_1\mathtt{XX} + c_2\mathtt{ZZ} + c_2\mathtt{XZ} + c_3\mathtt{IX} + c_4\mathtt{IZ} = \mathcal{H}.$$

This condition implies that $\mathtt{H}_1$ commutes with $\mathcal{H}$, so the eigenspaces of these operators coincide. Rewriting the Hamiltonian in a way that makes this clear, we have:

$$\mathcal{H} = c_1(\mathtt{X}+\mathtt{Z})\otimes\mathtt{X} + c_2(\mathtt{X}+\mathtt{Z})\otimes\mathtt{Z} + c_3(\mathtt{I})\otimes\mathtt{X} + c_4(\mathtt{I})\otimes\mathtt{Z}.$$

$\mathtt{H}_1$ has eigenvalues ±1, so the non-identity terms, $(\mathtt{X}+\mathtt{Z})$, will take (normalized) coefficients $\pm\sqrt{2}$ in the reduced Hamiltonian. On the other hand, the identity terms are unchanged by all operators, so they take the coefficient +1 for the reduced Hamiltonians. Therefore, by mapping to the ±1 eigen-subspaces of the $(\mathtt{X}+\mathtt{Z})$ operator on the first qubit, we are left with the following sub-Hamiltonians:

$$\begin{aligned}\mathcal{H}_{\mathcal{R}}^{+1} &= (\sqrt{2}c_1 + c_3)\mathtt{X} + (\sqrt{2}c_2 + c_4)\mathtt{Z}\\ \mathcal{H}_{\mathcal{R}}^{-1} &= (-\sqrt{2}c_1 + c_3)\mathtt{X} + (-\sqrt{2}c_2 + c_4)\mathtt{Z}.\end{aligned}$$

It can be verified that the four eigenvalues of these two Hamiltonians, $\mathcal{H}_{\mathcal{R}}^{+1}$ and $\mathcal{H}_{\mathcal{R}}^{-1}$, are equal to the four eigenvalues of $\mathcal{H}$. This appears to be a promising avenue, but there are complications with our foray into non-Pauli conserved charges.

The first question to answer is how to efficiently determine all of the symmetries. For Paulis, coefficients were irrelevant, since for a Pauli to commute with the Hamiltonian it is sufficient and necessary for it to commute with every Pauli operator in the sum. For other unitaries, this is most definitely not a necessary condition, and determining whether an arbitrary unitary commutes with the Hamiltonian cannot be done easily. Staying within the Clifford group, though, we at least have the promise that each Pauli operator in $\mathcal{H}$ is mapped to exactly one operator in $\mathcal{C}\mathcal{H}\mathcal{C}^\dagger$. Since Paulis form a basis, if $\mathcal{C}$ is to commute with $\mathcal{H}$, then it must be true that $\mathcal{C}$ is a permutation of the operators in $\mathcal{H}$ which respects the coefficients. This certainly appears tractable in contrast to the general case.

The second question, though, is to determine the eigenvalues of the Clifford. For Pauli symmetries, there were always two unique eigenvalues, ±1. For general Clifford circuits, there is no such guarantee. On this point, we expect it to be impractical to work with Clifford circuits of a sufficient depth, since it may be the case that the number of distinct eigenvalues of $\mathcal{C}$ is almost as many as $\mathcal{H}$, in which case there is not much to be gained from this exercise. For sufficiently small Clifford circuits, or for families with predictable eigenvalues, this is not an issue.

The last question with this approach is how to efficiently determine the reduced Hamiltonians. Once the eigenvalues of a commuting Clifford circuit, $\mathcal{C}$, are known, each unique eigenvalue, e_i will correspond to a reduced Hamiltonian, $\mathcal{H}_{\mathcal{R}}^{e_i}$. In some cases, such as the entangling gates CNOT and CZ as well as the SWAP gate, the sub-Hamiltonians do not even have dimension which is a power of 2. This will necessitate a mapping to a qudit[5] Hamiltonian, and we will have to take carefully calculate the terms and coefficients in each of the sub-Hamiltonians to ensure that the eigenvalues are accounted for correctly.

Having brought up all of these questions, there is reason to believe that this is both tractable and valuable. For tractability, as we noted, we are simply trying to identity permutations of the summands of the Hamiltonian which respect the coefficients. While the restrictions on circuit depth and the mappings to non-qubit Hamiltonians may be difficult, for many examples it should at least be feasible. If it is truly feasible as we hope, then we believe this will be a valuable direction for future research. Not only are Cliffords a larger, and therefore more powerful, class of symmetries to exploit, but there are also physical reasons why some Clifford symmetries may be abundant. In particular, SWAP symmetries are likely to correspond to real-world symmetries in which the structure of a molecule or lattice are symmetric about some axes.

[5] A qudit is a d-dimensional quantum state, in contrast to a qubit.

Maximal Reductions From Algorithm			
Hamiltonian	Qubits a Priori	Separable Measurements	Qubits Required
H_2	4	3	1
LiH	12	4	8
BeH_2	14	5	9
$\mathbb{Z}_2$ LGT with Gaussian Perturbation: 2×2	8	5	3
$\mathbb{Z}_2$ LGT with Gaussian Perturbation: 5×5	50	26	24
$\mathbb{Z}_2$ LGT with Gaussian Perturbation: 6×6	72	37	35
$\mathbb{Z}_2$ LGT with Gaussian Perturbation: 7×7	98	50	48
$\mathbb{Z}_2$ LGT with Gaussian Perturbation: 10×10	200	101	99
$\mathbb{Z}_2$ LGT with Gaussian Perturbation: 15×15	450	226	224
Hubbard	$2n$	2	$2(n-1)$
Kitaev Honeycomb: 1×1	4	3	1
Kitaev Honeycomb: 2×2	12	6	6
Kitaev Honeycomb: 5×5	60	25	35
Kitaev Honeycomb: 10×10	220	98	122
Kitaev Honeycomb: 15×15	480	220	260
Kitaev Honeycomb with J_z and J_{zz} Perturbation: 1×1	4	2	2
Kitaev Honeycomb with J_z and J_{zz} Perturbation: 2×2	12	3	9
Kitaev Honeycomb with J_z and J_{zz} Perturbation: 5×5	60	6	54
Kitaev Honeycomb with J_z and J_{zz} Perturbation: 10×10	220	11	209
Kitaev Honeycomb with J_z and J_{zz} Perturbation: 15×15	480	16	464
Kitaev Honeycomb with J_{zz} Perturbation: 1×1	4	3	1
Kitaev Honeycomb with J_{zz} Perturbation: 2×2	12	6	6
Kitaev Honeycomb with J_{zz} Perturbation: 5×5	60	25	35
Kitaev Honeycomb with J_{zz} Perturbation: 10×10	220	98	122
Kitaev Honeycomb with J_{zz} Perturbation: 15×15	480	220	260

Table 3.1: Qubit requirement reductions by applying these methods. LGT with or without Gaussian perturbation have the same requirements. More work is required to determine which models benefit most from the techniques shown above. When restricting to Pauli symmetries, these results are optimal.

Chapter 4

Adaptive Estimation of Quantum Observables

4.1 Introduction

A large challenge with variational algorithms challenge is the statistical nature of quantum mechanical measurements [103, 104, 105], requiring repeated measurements to estimate a quantum observable. The measurement problem is particularly relevant in current algorithms for NISQ devices that resort to an extensive sampling of the quantum system [106, 54]. The last few years have therefore seen an increased effort to find better measurement protocols that can lower the requirements on the quantum machine [31, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48]. Even for the restricted setting which we consider, namely partitioning the Hamiltonian into qubitwise or general commuting sets of Pauli operators, determining an optimal measurement strategy is a difficult task.

In this work, we introduce a novel algorithm called AEQuO that maximizes the information obtained from sampling the quantum system and learns from previous outcomes to improve the allocation of the remaining measurement budget. Compared to previous strategies, AEQuO is based on the ability of on-the-fly estimating not only the average of

This chapter closely follows the work presented in a published article on which I am a first author [4]. Ariel Shlosberg, my co-first author, was integral to developing the ideas and writing the code for the machine learning version of AEQuO, which outperforms the greedy bucket filling algorithm on large Hamiltonians. Thanks are owed as well to my other co-authors.

a given quantum observable, but also its error. This allows us to faithfully determine the precision of the estimated quantity, without the requirement of deriving error bounds [35, 37, 41, 42]. Furthermore, instantaneous error knowledge also permits better allocation of the measurement budget and is at the core of the learning capabilities of our algorithm.

We focus on the problem of accurately estimating a given observable O. This is a typical challenge encountered for example in quantum-classical hybrid protocols such as the variational quantum eigensolver [106, 54]. In particular, we want to estimate the expectation value of

$$O = \sum_{i=1}^{N} c_i \mathtt{P}_i, \quad \text{i.e.} \quad \langle O \rangle = \sum_{i=1}^{N} c_i \langle \mathtt{P}_i \rangle, \tag{4.1}$$

using M repeated preparations of the quantum state. Here, all c_i are real constants, the Pauli operator $\mathtt{P}_i$ labels a tensor product of Pauli operators, and $\langle Q \rangle \equiv \langle Q \rangle_\rho = \mathrm{tr}(\rho Q)$ denotes the expectation value of an observable Q with respect to the quantum state ρ.

Devising a protocol that allocates the budget M to minimize the estimation error is far from trivial. Several approaches have been proposed, including joint measurements of pairwise commuting operators [33], randomized and derandomized measurements [35, 37, 41, 42], grouping by weights [36], neural networks [38], minimizing the number of measurement groups [31, 34, 41, 43, 45, 46, 47, 48], and a method based on maximum entropy and optimal transport [107]. In this work, we introduce a measurement scheme for estimating observables that adaptively allocates the measurement budget based on previously collected data, allows for both non-bitwise commutation between Pauli operators and overlap in their grouping, and assesses both the average and variance of the observable O in Eq. (4.1).

This chapter is structured as follows. In Section 4.2, we provide an overview of our main results. In Section 4.3, we introduce the necessary background for observable estimation. We explain in detail the process of encoding an observable as in Eq. (4.1) in a weighted graph, and we discuss the connection between the estimation task and clique covers of a weighted graph. We explain AEQuO's subroutines in Section 4.4 and provide numerical results using chemistry Hamiltonians as a benchmark in Section 4.5. We conclude in Section 4.12 with some open problems and future directions of research. The appendices contain additional information further explaining our methods and results.

4.2 Overview of the main results

We introduce the Adaptive Estimator of Quantum Observables (AEQuO), a protocol designed to allocate the measurement budget in order to minimize the error affecting the estimate of an observable O as in Eq. (4.1). AEQuO iteratively changes the employed estimator based on the continuous inflow of new measurement data, it allows for both non-bitwise commutation relations between the Paulis and overlap in their grouping, and it yields estimates for both the average value and the error of O. It also employs Bayesian statistics, which permits the inclusion of prior knowledge and gives meaningful results even for small sample sizes. As a benchmark, we use AEQuO to estimate chemistry Hamiltonians, obtaining error estimates that improve on all state-of-the-art methods of estimating quantum observables.

A qualitative description of AEQuO's subroutines is given below. First, an operator O as in Eq. (4.1) is encoded in a (weighted) graph with vertex set $\{c_j\mathtt{P}_j\}$ [31]. An edge is drawn between $c_i\mathtt{P}_i$ and $c_j\mathtt{P}_j$ whenever the two Paulis commute, and the corresponding edge weight, representing their covariance, is initialized (see Eq. (4.2b) and Section 4.4). An estimate of $\langle O\rangle$ and its error can be directly calculated from this graph by assigning the vertices to groups of commuting operators that can be measured simultaneously. Crucially, we allow for overlap among these groups, that is, the operators $\mathtt{P}_j$ in Eq. (4.1) can belong to multiple groups, thus effectively increasing the amount of gathered information.

The second step of AEQuO consists in iteratively assigning a number of measurements to each of these groups until the budget M is exhausted. Importantly, we develop a method (see Section 4.3) to calculate the estimation error that AEQuO minimizes while allocating the M shots. We provide two different subroutines for this assignment. One is a greedy "bucket-filling" (BF) algorithm that assigns the M measurements one by one. This algorithm gives a good estimation error in small problem instances, but performs poorly for large problem sizes due to its greedy nature. To remedy this, we developed another subroutine based on machine learning (ML) that requires fewer repetitions, as it allocates a fraction of the total budget M to the most promising groups of commuting Pauli operators in each iteration. As a result, the ML algorithm is faster for large values of M and N, the latter being the number of Paulis in Eq. (4.1).

In a final step, we post-process the gathered data to obtain the estimator characterized by the smallest error based on the performed measurements. The post-processing utilizes the cumulative knowledge gathered during the measurement phase. It considers the contribution to the estimation error from each Pauli operator in different commuting groups, removing it if it is statistically likely to improve the error.

4.3 Theory

The Pauli operators, $\mathtt{P}_i$, in Eq. (4.1) acts on n qubits, $\mathtt{P}_i = \bigotimes_{i=1}^{n} \mathtt{W}(ij)_j$. Here, $\mathtt{W}(ij)$ labels the element of the Pauli operator, i.e. $\mathtt{W}(ij) \in \{\mathtt{I}, \mathtt{X}, \mathtt{Y}, \mathtt{Z}\}$, and the subscript j indicates the corresponding qubit on which a Pauli operator acts. A decomposition of an observable into Pauli operators as in Eq. (4.1) has at most $N \leq 4^n$ terms; however, in most physical examples the number of Paulis with non-zero weight c_i scales polynomially in n.

In an experiment, we collect m_i measurement outcomes ("shots") for each $\mathtt{P}_i$ and take their average to obtain an estimate $\tilde{\mathtt{P}}_i$ of $\langle \mathtt{P}_i \rangle$, which subsequently allows us to estimate $\langle O \rangle$ through $\tilde{O} = \sum_i c_i \tilde{\mathtt{P}}_i$. Considering that two commuting Pauli operators $\mathtt{P}_i$ and $\mathtt{P}_j$ can be measured simultaneously (i.e., in the same shot), an estimate for the error $(\Delta\tilde{O})^2$ affecting $\tilde{O}$ is

$$(\Delta\tilde{O})^2 = \sum_{i,j=1}^{n} c_i c_j \, C(\tilde{\mathtt{P}}_i \, ; \tilde{\mathtt{P}}_j), \tag{4.2a}$$

$$C(\tilde{\mathtt{P}}_i \, ; \tilde{\mathtt{P}}_j) = \tilde{\mathcal{Q}}_{ij} \frac{m_{ij} + \delta(m_{ij})\delta(i-j)}{m_i m_j + \delta(m_i)\delta(m_j)}. \tag{4.2b}$$

Here, $\tilde{\mathcal{Q}}$ is the covariance matrix of all measured data, m_{ij} denotes the number of shots where $\mathtt{P}_i$ and $\mathtt{P}_j$ have been sampled simultaneously, and δ is the Kronecker delta function with $\delta(0) = 1$ and $\delta(x) = 0$ for $x \neq 0$. In the limit of many measurements,

$$\tilde{\mathcal{Q}}_{ij} \to \mathrm{cov}(\mathtt{P}_i, \mathtt{P}_j) \equiv \langle \mathtt{P}_i \mathtt{P}_j \rangle - \langle \mathtt{P}_i \rangle \langle \mathtt{P}_j \rangle .$$

Note that $m_{ii} = m_i$, and that $\sum_i m_i = M$ if and only if $m_{ij} = 0$ for all $i \neq j$. In this special case, Eqs. (4.2) yield the well known result

$$(\Delta\tilde{O})^2 = \sum_i \frac{c_i^2 (\Delta\tilde{\mathtt{P}}_i)^2}{m_i},$$

where $(\Delta\tilde{\mathtt{P}}_i)^2 \equiv \tilde{\mathcal{Q}}_{ii}$.

To obtain $\tilde{O}$, there is no unique way to choose groups of Paulis that are pairwise commuting, in particular if one allows these groups to have overlap. Every choice of groups corresponds to a specific estimator in the form of Eqs. (4.2) characterized by the values of m_{ij} and m_j. In the following, we explain the aforementioned graph representation used in our protocol, and the idea behind our algorithm for grouping Paulis.

As shown in Figure 4.1(a), we represent the operator O in Eq. (4.1) with a weighted graph whose vertices correspond to $c_i \tilde{\mathtt{P}}_i$. If $\mathtt{P}_i$ and $\mathtt{P}_j$ commute, their vertices are connected

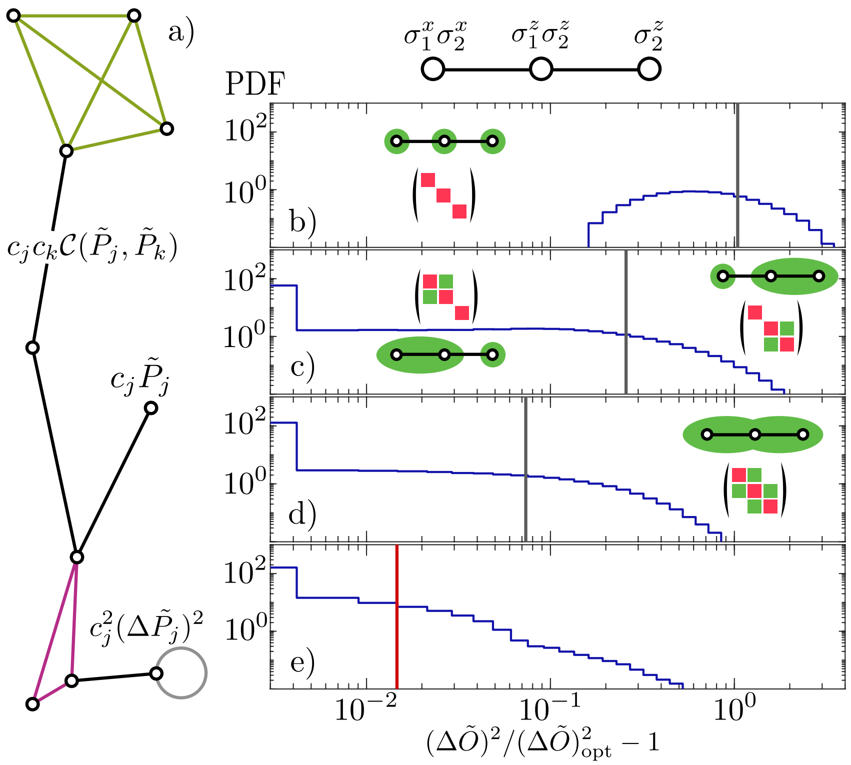


Figure 4.1: (a): Example of a graph. Vertices and edges are weighted with $c_i\tilde{\mathtt{P}}_i$ and $c_ic_jC(\tilde{\mathtt{P}}_i\ ;\tilde{\mathtt{P}}_j)$, respectively. As shown by the grey circle, we include self-edges (omitted in the rest of the graph) with variances $c_i^2(\Delta\tilde{\mathtt{P}}_i)^2$. Two maximal cliques are highlighted in green and violet. (b-e): Case study of the operator $O = \mathtt{XX} + \mathtt{ZZ} + \mathtt{IZ}$ using its graph representation (top). The four non redundant covers are highlighted within plots (b-d), along with the $\tilde{\mathcal{Q}}_{ij}$ elements contributing to the associated error $(\Delta\tilde{O})^2$ (matrices with red and green squares). Results from measurements of these covers are plotted in the graphs. We considered $2 \cdot 10^5$ experiments where $M = 10^4$ experimental shots are used to estimate $\tilde{O}$ and $(\Delta\tilde{O})^2$ for pure states uniformly sampled on the Bloch sphere. For each cover, measurement allocation is optimal and is determined with the full knowledge of $\tilde{\mathcal{Q}}$. As discussed in Section 4.3 and Section 4.6, for each state one cover gives the smallest error $(\Delta\tilde{O})^2_{\mathrm{opt}}$, to which the results obtained for the four plots are compared. Vertical lines represent averages. In the bottom panel (e), we collect the results obtained by AEQuO *without* prior knowledge of $\tilde{\mathcal{Q}}$, and using the ML subroutine with $L = 1$.

by an edge with weight $c_i c_j\, C(\tilde{\mathtt{P}}_i\ ; \tilde{\mathtt{P}}_j)$. Therefore, $\tilde{O}$ is obtained by summing over all vertices, while $(\Delta\tilde{O})^2$ is the sum of all edges (including self-edges). Different $\mathtt{P}_i$ can only be measured simultaneously if they belong to the same clique of the graph, i.e., a fully connected subset of vertices. A clique is called maximal if no further vertex can be added to it [see Figure 4.1(a)]. Evidently, an estimator corresponds to a clique cover, i.e., a set of cliques such that each vertex of the graph is included in at least one clique.

For illustrative purpose, let us consider the example $O = \mathtt{XX}+\mathtt{ZZ}+\mathtt{IZ}$ in Figure 4.1(b-e). There are five cliques that can be arranged in four different, non-redundant covers[1]. The particular choice of clique cover determines the terms $C(\tilde{\mathtt{P}}_i\ ; \tilde{\mathtt{P}}_j)$ contributing to the error $(\Delta\tilde{O})^2$, as can be seen in the pictorial representations of the matrices $\tilde{\mathcal{Q}}$ in Figure 4.1(b-d). For a given input state, the estimator $\tilde{O}$ corresponding to the cover yielding the minimal $(\Delta\tilde{O})^2$ for fixed M gives the most accurate estimation (since all estimators are unbiased).

We test the performances of the four estimators (viz. covers) in Figure 4.1(b-d) with randomly chosen two-qubit pure states, and calculate the relative deviation of the error estimate w.r.t. the best estimator. Here, we use the asymptotic values for Eq. (4.2b) that are obtained for $M \to \infty$. The results demonstrate that, as expected, measuring each Pauli operator separately [Figure 4.1(b)] is never a good strategy, while the intuitively best estimator [Figure 4.1(d)], corresponding to the cover made of maximal cliques only, is optimal in only ∼54% of the cases. More details about how the best estimator is determined and the measurement budget is allocated can be found in Section 4.6.

In practice, the state to be measured is unknown and it is not possible to test all covers beforehand to identify the best one. This motivates an adaptive approach in which the clique cover is changed during the measurement process depending on former outcomes. In our protocol, we update the vertex and edge weights of the graph after some (or each) of the M shots. We then decide which clique is measured next, based on its contribution to Eq. (4.2a).

Conveniently, the choice of the particular estimator can be relegated to a second post-processing step; in the data acquisition phase we may restrict to considering maximal cliques only. Changing the estimator (or equivalently, the clique cover) can then be done subsequently by adjusting the measurement numbers m_{ij} and m_i, and updating $\tilde{Q}$. For example, if one removes $\mathtt{P}_i$ from a clique, then for all $\mathtt{P}_j$ in that clique one has to reduce m_{ij} and m_i by the number of shots allocated to that clique. Furthermore, all outcomes gathered for $\mathtt{P}_i$ in the considered clique have to be discarded in order to avoid introducing biases

[1] In general, one can find more covers by nesting smaller cliques into bigger ones. However, these cases are not relevant to us since only one of these nested cliques will yield minimal error. Therefore, without loss of generality, we can limit ourselves to the study of the four covers in Figure 4.1.

in the estimator. In the example in Figure 4.1(d), changing the estimator corresponds to removing the Pauli operator ZZ from one of the two cliques.

The advantages of employing maximal cliques and post-processing are twofold. First, restricting to maximal cliques reduces the number of choices in each shot. Second, potentially available data is never neglected, resulting in a better knowledge of the vertex and edge weights of the graph. This knowledge can then be used for more informed choices, both of the maximal cliques to be measured at each shot, and in the post-processing stage itself.

In the histograms of Figure 4.1(b-e), we plot the expected deviations from the minimal error, which is derived by always identifying the best cover out of the four. The black and red lines in the figures indicate the mean values of the associated histograms. A comparison shows that AEQuO [in Figure 4.1(e)], based on maximal cliques and post-processing, consistently provides near-minimal error and greatly outperforms the static approaches [in Figure 4.1(b-d)] represented by the four covers in the figure. For more details, see Section 4.6.

As a conclusive remark, we present two corollaries following from our graph representation of the observable O and Eqs. (4.2). Namely, we derive simple upper bounds on both the error $(\Delta\tilde{O})^2$ and its scaling with respect to the number N of Paulis within O. To bound the error, we observe that the maximum error contribution of two Paulis $\mathtt{P}_i$ and $\mathtt{P}_j$ measured together is $c_i^2 + c_j^2 + 2|c_i c_j|$, which can be obtained by setting all $\tilde{\mathcal{Q}}_{ij}$ equal to $\mathrm{sgn}(c_i)\mathrm{sgn}(c_j)$ in Eq. (4.2b). This yields the tight upper bound

$$(\Delta\tilde{O})^2 \leq \sum_{i,j=1}^{n} |c_i c_j| \, \frac{m_{ij}}{m_i m_j}.$$

Here, we have omitted the delta functions for clarity. Given any measurement strategy, this equation determines, *beforehand*, the maximum error that can possibly affect the resulting estimator. The scaling of $(\Delta\tilde{O})^2$ in terms of N can be understood using graph theory. Considering that it is always possible to find a graph's cover of N cliques, one concludes that in the worst-case scenario $(\Delta\tilde{O})^2$ grows linearly in N.

4.4 Algorithm

In the following, we present our algorithm AEQuO, which is outlined in the diagram in Figure 4.2. First, the graph is constructed in time $\mathcal{O}(N^2)$. At this stage, it is possible to choose whether or not to connect vertices whose associated Paulis are non-bitwise commuting. As discussed below in Section 4.5, restricting to bitwise commuting Pauli operators results in errors $(\Delta\tilde{O})^2$ that are higher. However, simultaneously measuring non-bitwise commuting Paulis requires entangling gates, which in the context of NISQ devices are expensive. AEQuO includes a sub-routine for finding a suitable circuit that diagonalizes a group of commuting Paulis [108, 31, 78]. This subroutine is classically efficient for large values of the numbers N and n of Paulis and qubits, respectively, and the resulting circuit is ensured to contain at most $n(n-1)/2$ (zero) entangling gates when allowing (forbidding) non-bitwise commutation. We note that this is slightly worse than the known optimal bound, $\Theta(n^2/\log n)$, given in Chapter 5, but for our classical simulations, extra entangling gates does not introduce errors. For noisy simulations or real-world implementations, Chapter 5 provides techniques for optimizing these values.

After the graph is built, AEQuO finds a desired number r of maximal cliques. We prioritize cliques that are statistically likely to have bigger contributions to the error $(\Delta\tilde{O})^2$ in Eq. (4.2a) (see Section 4.9 and [108]) and only consider maximal ones for the reasons explained above. However, our protocol may also operate on all cliques. The runtime required to find the clique cover scales as $\mathcal{O}(r)$. Efficient algorithms for finding locally-maximal cliques are known, e.g., the Bron-Kerbosch algorithm [109].

We now assign weights to the vertices and edges of the graph constructed from the observable O. We resort to Bayesian estimation (see Section 4.7) for both the initialization and all subsequent updates of the weights of the graph, which gives meaningful results with scarce statistics or even in the absence of any data. This is crucial due to the adaptive nature of our protocol, and allows us to initialize the graph without any pre-sampling. Indeed, when no shots are taken, Bayesian estimation prescribes $\tilde{\mathtt{P}}_i = 0$ and $\tilde{\mathcal{Q}}_{ij} = 2\delta(i-j)/3$ for all $i, j = 1, \dots, N$.

Once the r cliques have been found and the graph has been initialized, the system can be measured. In order to choose the clique to be probed at each shot, we propose two possible subroutines whose objective is to minimize the cost function $(\Delta\tilde{O})^2$ in Eqs. (4.2) (for a numerically efficient method to compute $(\Delta\tilde{O})^2$ see Section 4.8). The first choice is a greedy "bucket-filling" (BF) algorithm [110] whose premise is to allocate measurements one-by-one by evaluating the predicted cost function before each shot. After a certain amount of measurements (the M shots are divided into L chunks of increasing sizes, as explained below), the graph is updated and the BF algorithm is run again. This approach works well for instances where r and M are small, since the number of cost function evaluations increases linearly in these variables.

As an alternative to the BF method, my co-author, Ariel Shlosberg, developed and

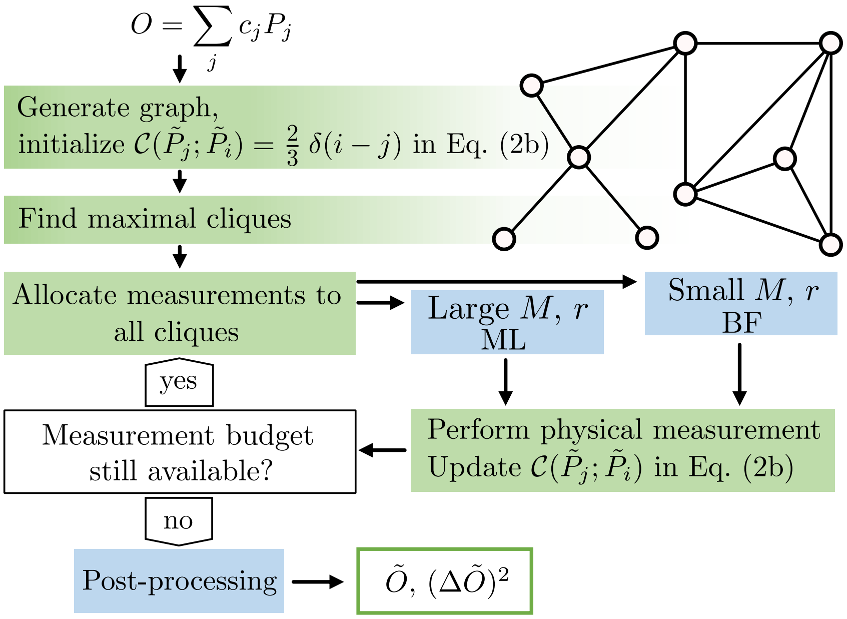

$O = \sum_j c_j P_j$
Generate graph,
initialize $\mathcal{C}(\tilde{P}_j; \tilde{P}_i) = \frac{2}{3}\,\delta(i-j)$ in Eq. (2b)
Find maximal cliques
Allocate measurements to all cliques
Large M, r
ML
Small M, r
BF
Perform physical measurement
Update $\mathcal{C}(\tilde{P}_j; \tilde{P}_i)$ in Eq. (2b)
yes
Measurement budget still available?
no
Post-processing
$\tilde{O}$, $(\Delta\tilde{O})^2$

Figure 4.2: Schematic diagram of our protocol. From a given observable O, its graph representation is derived and the maximal cliques are found. Depending on the number of cliques r and the measurement budget M, the BF or the ML algorithm is chosen. Afterwards, the system is probed and vertices and edges of the graph are updated according to the outcomes. If M shots have been taken, the post-processing resorts to the desired estimates for the average and error. Otherwise, another round of allocation and measurements is performed. Green and light blue boxes represents essential and optional subroutines, respectively. In fact, the user is free to choose between the BF and the ML algorithm, while post-processing can be turned off if desired.

implemented an approach inspired by a recent structural optimization algorithm based on machine learning (ML) [111]. We reparamterize the vector of clique measurements, which are the set of optimization variables in the BF approach, in terms of the weights and biases of a densely-connected neural network. The output of the network yields the (locally) optimal measurement allocations after learning occurs. This approach alters the optimization landscape by increasing the number of parameters.

In contrast to BF, the ML algorithm allocates a fraction of the total budget to the most promising cliques. Subsequently, the graph is updated and the ML algorithm is run again. We perform L iterations of the process, at each step updating the covariance matrix $\tilde{\mathcal{Q}}$ with the experimental outcomes and progressively increasing the available shots by a factor $l > 0$ until all M measurements are exhausted. Thus, for large problem sizes the ML subroutine scales more favorably compared to the BF algorithm, since it requires fewer repetitions (L for ML versus M for BF), and is more efficient for large values of r.

The ML approach is based on minimizing the cost function, Eq. (4.2a), over the trainable parameters of the neural network. We implement a variety of different optimizers to determine the weights and activation biases [108], including stochastic gradient descent [112], Adam [113], and a Limited-Memory BFGS method [114, 112, 115]. These different optimizers yield similar performances; for instance, for Figs. 4.3 and 4.4 the stochastic gradient descent and the Limited-Memory BFGS methods, respectively, were employed.

The architecture of the neural network we settled on is composed of three densely-connected layers of width r, and with ReLU activation functions on hidden layers and a softmax activation function on the output [116]. Such a structure yields a probability distribution at the output corresponding to the ratio of times that each clique should be measured. The output of the neural network is then converted to an integer measurement allocation vector by first scaling by the number of measurements to be performed, then flooring the entries, and finally by allocating excess measurements to the cliques with the largest percentage change in budget due to the flooring operation [108]. Before the learning phase, the network is initialized such that each clique is measured the same number of times and so that measurements between various cliques are initially uncorrelated.

After completing all measurements, post-processing can be applied. In this step of the protocol, we determine the estimator with the lowest variance that could be realized with the available data. For each pair, $\mathtt{P}_i$ and $\mathtt{P}_j$, such that $m_{ij} \neq 0$ and $c_i c_j C(\tilde{\mathtt{P}}_i\ ; \tilde{\mathtt{P}}_j) > 0$, we find all cliques where these strings have been simultaneously probed and consider all possibilities of removing either of those or keeping the estimator as it is. Eventually, the configuration minimizing the updated error function $(\Delta\tilde{O})^2$ is kept. The runtime of this procedure scales exponentially in the size of the subset of cliques where $\mathtt{P}_i$ and $\mathtt{P}_j$ have been measured simultaneously. In practical examples, this number is small and post-processing can be used for large values of N. For instance, post-processing has been used for all numerical results presented in Figure 4.3, where we considered observables with up to $N = 1086$ terms corresponding to the Hamiltonian of H_2O.

4.5 Results

Our numerical results are reported in Figure 4.3. In panel (a), we list the standard deviations Σ (see caption and Section 4.9) affecting the estimated ground state energies for chemistry Hamiltonians [22] of various molecules, choosing $M = 1000$ in each case. Values in square brackets are averages of $\sqrt{(\Delta\tilde{O})^2}$, calculated by AEQuO with the graph representation described in Section 4.3. We compare AEQuO with an in-house developed version (see below and Section 4.9) of the largest degree first (LDF) [31], the overlapped grouping measurement (OGM) [43], the adaptive Pauli shadow (APS) [41, 42], and the derandomized shadow (Derand) [37] methods.

In the first two, the goal is to minimize the number of cliques in the resulting cover, while the latter two reconstruct the desired estimator from the outcomes of partially random as well as deterministically allocated measurements. We choose these as representatives of the variety of algorithms based on different grouping strategies [31, 33, 34, 35, 36, 117, 46, 47, 48] and the classical shadow technique [37, 38, 39, 40, 41, 42]. AEQuO is run both for $L = 1$ and $L = 2$ (in this case $l = 9$) using the ML subroutine, and outperforms other approaches in determining more precise estimates.

As discussed in more detail below, this improvement has two reasons. First, we allow for both non-bitwise commutation relations between Pauli operators and overlapping cliques. On the one hand, this allows us to gather more information from the same number of shots. On the other hand, it increases the number of non-zero elements $\tilde{\mathcal{Q}}_{ij}$ in Eqs. (4.2). While these covariances can either lower or increase the error $(\Delta\tilde{O})^2$, the adaptive nature of AEQuO (see next paragraphs) and the post-processing ensure that negative ones are preferred. We remark that our version of the LDF protocol also allows for non-bitwise commutation relations; as a result, it typically outperforms OGM, APS and Derand. As shown in Section 4.10, when restricted to bitwise commutation relations, LDF yields errors that are generally higher than all other protocols.

Second, AEQuO allocates the shots by directly minimizing the estimated error $(\Delta\tilde{O})^2$ in Eqs. (4.2) based on on the available experimental information. Provided the ML and/or the BF subroutines find the global minimum of $(\Delta\tilde{O})^2$, this results in AEQuO finding a)

$10^{-2}\times$	**LDF** [16]	**OGM** [27]	**APS** [25,26]	**Derand** [21]	**AEQuO** L=1	**AEQuO** L=2, l=9
H_2(8 JW)	3.5	5.1	5.2	6.2	2.8 [3.2(0)]	**2.5** [2.4(1)]
H_2(8 BK)	4.5	5.0	5.0	4.8	3.1 [3.2(0)]	**2.1** [2.4(1)]
H_2(8 Par)	3.8	4.1	4.8	5.3	3.4 [3.2(0)]	**1.7** [2.4(1)]
LiH(12 JW)	3.0	2.2	3.8	4.5	**2.0** [2.3(1)]	2.1 [2.2(1)]
LiH(12 BK)	3.8	2.7	4.2	3.6	2.2 [2.3(1)]	**2.2** [2.2(1)]
LiH(12 Par)	4.6	2.6	3.8	3.9	2.6 [2.3(1)]	**1.9** [2.2(1)]
BeH_2(14 JW)	5.8	5.4	6.5	10.3	3.8 [4.0(1)]	**3.7** [3.6(1)]
BeH_2(14 BK)	6.5	7.4	6.9	10.3	4.4 [4.0(1)]	**3.7** [3.6(1)]
BeH_2(14 Par)	4.6	4.7	6.9	8.4	**3.3** [4.0(1)]	3.9 [3.7(1)]
H_2O(14 JW)	20.1	12.9	11.7	20.6	8.7 [8.4(2)]	**6.2** [6.5(1)]
H_2O(14 BK)	22.1	14.7	10.8	29.9	**6.2** [8.4(3)]	7.0 [6.5(1)]
H_2O(14 Par)	15.7	12.4	12.6	20.6	9.0 [8.4(3)]	**5.8** [6.5(1)]

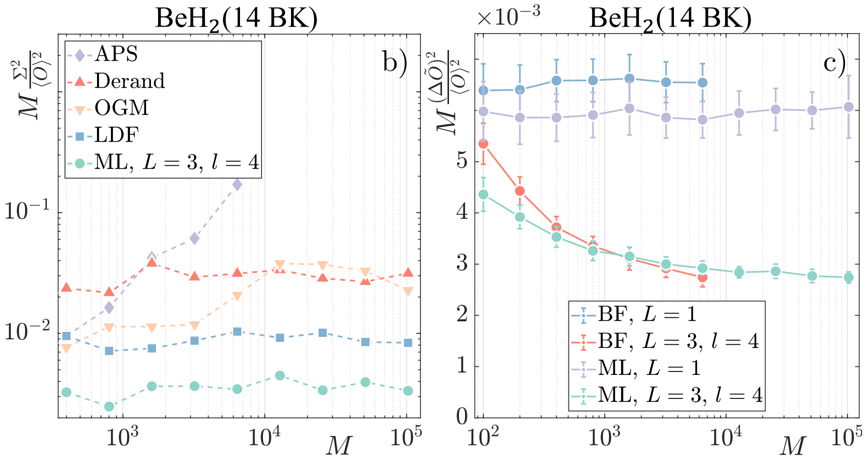


Figure 4.3: (a): Errors obtained with LDF [31], OGM [43], APS [41, 42], Derand [37], and AEQuO with the ML subroutine and $L=1$ or $L=2$ and $l=9$. We consider chemistry Hamiltonians with different numbers of qubits (in parentheses), and three encodings: Jordan-Wigner (JW), Bravyi-Kitaev (BK), and parity (Par) [30, 22]. Values (in bold the lowest) are standard deviations $\Sigma \equiv [\sum_{j=1}^{R}(\tilde{O}_j - \langle O\rangle)^2/R]^{1/2}$, where $\tilde{O}_j$ is the j-th estimated average for O. In square brackets, we report averages of $[(\Delta\tilde{O})^2]^{1/2}$ with their statistical error. The input is the ground state, $R=40$, $M=10^3$, and values are rescaled by 10^{-2}. (b-c): Relative errors as a function of M obtained considering the BeH$_2$ Hamiltonian with BK encoding. In (b) we compare values of $M\Sigma^2/\langle O\rangle^2$ for the approaches in the legend. In (c), we report $M(\Delta\tilde{O})^2/\langle O\rangle^2$ calculated with the BF and ML subroutines of AEQuO with different L and l. Error bars are statistical errors. In (b) and (c) we set $R=10^2$ and $R=40$, respectively. Details in Section 4.9.

the estimator $\tilde{O}$ that, constrained by the used cliques and the transient knowledge about $\tilde{\mathcal{Q}}$, is characterized by the smallest possible error. This is supported by Figure 4.3(b-c), where we plot the relative errors $M\Sigma^2/\langle O\rangle^2$ and $M(\Delta\tilde{O})^2/\langle O\rangle^2$, respectively, of the BeH_2 Hamiltonian in the Bravyi-Kitaev (BK) encoding. Despite statistical fluctuations, panel (b) shows that AEQuO outperforms all other approaches. Similar to other methods, we also see that AEQuO provides errors that scale as $1/M$ in the chosen range of M.

The adaptive features of AEQuO are evident in Figure 4.3(c), where we compare the BF and ML subroutines for different values of L and l. For $L = 1$ there is no memory of prior outcomes; in this case, the points obtained with either subroutine lie on a horizontal line. However, for $L = 3$ and $l = 4$, AEQuO learns and uses the gathered information to find better strategies for allocating the remaining shots.

For large M, asymptotes are recovered, which are considerably lower than the ones for $L = 1$. Importantly, the advantage from the adaptive nature of AEQuO comes for free, in the sense that it follows exclusively from a better allocation of the measurement budget and does not require extra resources (such as entangling gates for the simultaneous diagonalization of the Paulis). We investigate the improvement resulting from the adaptivity of AEQuO in more detail in Figure 4.4 and in Section 4.10, where we restrict all protocols considered in this section to bitwise commutation relations.

While the BF and ML subroutines yield comparable results, Figure 4.3(c) indicates that they outperform each other in different parameter regimes, depending on the problem at hand. This is a consequence of several factors. By construction, the BF picks the clique to be measured based on the gradient of $(\Delta\tilde{O})^2$. On the other hand, the ML subroutine (in this context) follows a stochastic gradient descent algorithm [112] that allows for exploring the error landscape. However, its performance depends on hyperparameters [108, 111, 113] that, in Figure 4.3, have not been optimized to keep low computational requirements.

As explained above, our protocol has three distinctive features. Besides the ability to monitor and determine the error $(\Delta\tilde{O})^2$ [see Eqs. (4.2)] that allows for better allocation of the measurements and assessment of the precision of the estimator, AEQuO can also exploit both non-bitwise commutation between Paulis[2] and cliques' overlaps, and it is adaptive. In the remainder of this section, we investigate the advantages provided by these last two features both with chemistry Hamiltonians and a family of 2D and 3D lattice models of interacting spin 1/2 particles (for details, see Section 4.11).

In Figure 4.4(a), we list averaged errors $(\Delta\tilde{O})^2$ for several chemistry Hamiltonians[3], with $M = 10^4$ and the ground state as input [22]. AEQuO is run with different settings, namely

[2] This ability is native to our LDF protocol as well.

[3] We show results relative to the BK encoding; the JW and Parity ones are qualitatively identical.

(not) allowing non-bitwise commutation relations between the Paulis, and with adaptive allocation turned either on ($L = 3$, $l = 4$) or off ($L = 1$). As expected, the smallest errors are obtained when AEQuO can learn from previous outcomes ($L = 3$, $l = 4$) and non-bitwise commuting Pauli operators can be simultaneously measured [indicated by GC in Figure 4.4(a)]. In the case of chemistry Hamiltonians with their ground state as input, the advantage is remarkable. If we compare AEQuO when restricted to bitwise commutation (BC) and $L = 1$ with the best results (bold numbers), we see that measurement budgets are required to be up to 14.4 times bigger [see numbers in square brackets in Figure 4.4(a)] in order to obtain the same estimator's precision.

For comparison, up to statistical noise, the Derand [37] and OGM [43] protocols yield results that are similar to AEQuO's when we restrict it to bitwise commutation and turn the adaptive allocation off (BC, $L = 1$). The direct comparison between the BC versions of AEQuO and all other protocols considered in this section is presented in Section 4.10.

A similar analysis is made in Figure 4.4(b-e), where we considered a generalized Hubbard model [118] in which qubits are located on edges of the 2D [panels (b) and (d)] and 3D [panels (c) and (e)] lattices drawn at the bottom of the figure. As explained in Section 4.11, they are allowed to hop with (many-body) flip-flop interactions, and we include energy shifts that depend on the states of qubits on neighbouring edges. We use the ground state as input, set $M = 10^4$, and run AEQuO with the same settings as in Figure 4.4(a): allowing for (squares) or forbidding (circles) non-bitwise commutation, and turning the adaptive allocation on (red and violet points) or off (blue and green points).

In panels (b-c), we show the relative increase of the error $(\Delta\tilde{O})^2$ if compared to $(\Delta\tilde{O})^2_{\text{opt}}$, that is the value obtained by AEQuO when allowing for non-bitwise commutation and providing prior knowledge of the input state. In practice, prior knowledge is given by initializing the covariance matrix $\tilde{Q}_{ij}$ in Eq. (4.2b) with $\text{cov}(\mathtt{P}_i, \mathtt{P}_j)$ for all $i, j = 1, \ldots, N$. Therefore, the data points in Figure 4.4(b-c) represent the relative increase of the measurement budget M that is required for reaching the error $(\Delta\tilde{O})^2_{\text{opt}}$. Values for $(\Delta\tilde{O})^2_{\text{opt}}$ correspond to the black squares in panels (d-e), depicting the errors $(\Delta\tilde{O})^2$ determined by AEQuO with prior knowledge of the state.

It is evident from Figure 4.4 that the advantages from the adaptive allocation and non-bitwise commutation are consistent, yet vary considerably between data points. All red squares in panels (b) and (d) are approximately zero, indicating that AEQuO learns in the process and for sufficiently large M it allocates the measurements *as if* it knew the input state *beforehand*. All other coloured data points lie above, suggesting that it is detrimental not to exploit non-bitwise commutation and adaptive allocation.

How advantageous these features are *highly* depends on the considered observable and a)

$10^{-4}\times$	**GC** L=3, l=4	**GC** L=1	**BC** L=3, l=4	**BC** L=1
H_2(8 BK)	**0.47(4)**	1.26(7) [2.7]	0.92(1) [2.0]	2.09(1) [4.4]
LiH(12 BK)	**0.24(1)**	0.81(6) [3.4]	1.12(1) [4.7]	1.78(2) [7.4]
BeH_2(14 BK)	**1.0(1)**	2.7(2) [2.7]	4.6(1) [4.6]	8.7(1) [8.7]
H_2O(14 BK)	**4.4(2)**	18.7(15) [4.3]	22.0(4) [5.0]	63.2(17) [14.4]

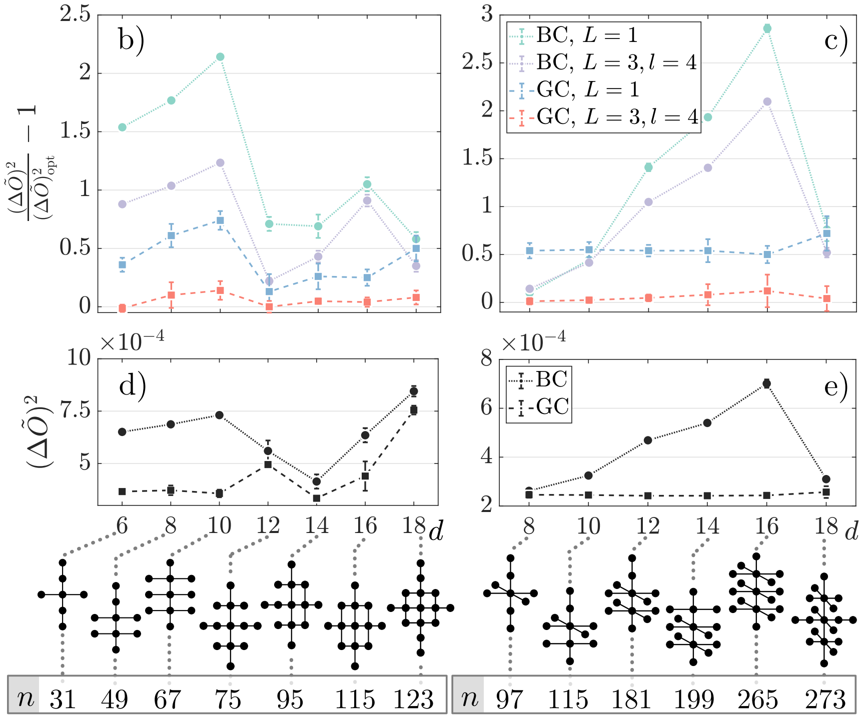


Figure 4.4: (a): Averaged errors $(\Delta\tilde{O})^2$ (standard deviations in brackets, bold indicates lowest) obtained by AEQuO for chemistry Hamiltonians and different values of L and l. BC (GC) refers to bitwise (general, i.e., non-bitwise) commutation. Relative overheads compared to the GC with $L = 3$, $l = 4$ are in square brackets. (b-c): Averaged relative errors $(\Delta\tilde{O})^2/(\Delta\tilde{O})^2_{\mathrm{opt}} - 1$ as a function of the number of qubits n for different 2D (b) and 3D (c) lattices (see main text and Section 4.11). Data are obtained with different values of L and l, and resorting to BC or GC. $(\Delta\tilde{O})^2_{\mathrm{opt}}$ corresponds to the black squares connected by dashed lines in panels (d-e), where we report averaged errors $(\Delta\tilde{O})^2$ obtained with prior knowledge of the state. For (b-e), the considered lattices with the associated N are at the bottom, error bars are standard deviations and we used the ground states of the models. For all panels, $M = 10^4$ and values are obtained by averaging 25 independent runs of AEQuO.

state. Empirically, non-bitwise commutation is more beneficial if the Hamiltonian is dominated by long-range many-qubits interactions, such that two commuting Paulis are more likely to commute non-bitwise. Chemistry Hamiltonians fall within this category. On the other hand, in most of the considered examples adaptive allocation resorted to precision improvements varying between 25% and 400%, with the tendency of increasing with the numbers of qubits n and Paulis N.

4.6 A minimal example

In this section we consider the example presented in Figure 4.1(b-e), consisting of an operator

$$O = \mathtt{XX} + \mathtt{ZZ} + \mathtt{IZ} = P_1 + P_2 + P_3. \tag{4.3}$$

For this operator O and a generic input state ρ_{in}, we are interested in finding the best possible estimator (viz. a cliques' cover), as well as the optimal allocation of the M measurements. In order to do so, we assume that the three Pauli operators $\mathtt{P}_i$ $(i = 1, 2, 3)$ have been previously probed infinitely many times, such that the values of $\tilde{\mathcal{Q}}_{ij} \to \text{Tr}\rho_{\text{in}}\mathtt{P}_i\mathtt{P}_j - \text{Tr}\rho_{\text{in}}\mathtt{P}_i\text{Tr}\rho_{\text{in}}\mathtt{P}_j$ are available beforehand. For the sake of clarity, we remark that this assumption was *not* taken when using AEQuO [see Figure 4.1(e)]. This demonstrates that our approach, for a generic input state, is more likely to provide lower errors $(\Delta\tilde{O})^2$ than any of the possible covers associated to the operator O in Eq. (4.3), despite the disadvantage of not knowing $\tilde{\mathcal{Q}}$ beforehand.

For a generic input state ρ_{in}, one out of the four covers presented in Figure 4.1(b-d) resorts to the smallest error $(\Delta\tilde{O})^2_{\text{opt}}$, provided the M measurements have been allocated optimally. The idea is to find this $(\Delta\tilde{O})^2_{\text{opt}}$ and compare it with the errors $(\Delta\tilde{O})^2$ determined with the different covers in Figure 4.1(b-d) and AEQuO in Figure 4.1(e). The relative difference $[(\Delta\tilde{O})^2 - (\Delta\tilde{O})^2_{\text{opt}}]/(\Delta\tilde{O})^2_{\text{opt}}$ plotted in the histograms is thus representative of how well the associated covers (or AEQuO) perform with respect to a wide variety of random input states ρ_{in}. Specifically, ρ_{in} correspond to pure states uniformly distributed on the Bloch sphere.

As a first step, we describe how to determine the optimal allocation of the M measurements for any clique cover. We start by rewriting the errors $(\Delta\tilde{O})^2$ associated to the four different covers as a sum of the elements in the matrices (recall $c_i = 1$ for all $i = 1, 2, 3$)

$$(\Delta\tilde{O})^2 = \sum \begin{bmatrix} \frac{\tilde{\mathcal{Q}}_{11}}{\xi_1} & 0 & 0 \\ 0 & \frac{\tilde{\mathcal{Q}}_{22}}{\xi_2} & 0 \\ 0 & 0 & \frac{\tilde{\mathcal{Q}}_{33}}{\xi_3} \end{bmatrix} \text{ for Figure 4.1(b),} \tag{4.4}$$

$$(\Delta\tilde{O})^2 = \begin{cases} \sum \begin{bmatrix} \frac{\tilde{\mathcal{Q}}_{11}}{\xi_1} & \frac{\tilde{\mathcal{Q}}_{12}}{\xi_1} & 0 \\ \frac{\tilde{\mathcal{Q}}_{12}}{\xi_1} & \frac{\tilde{\mathcal{Q}}_{22}}{\xi_1} & 0 \\ 0 & 0 & \frac{\tilde{\mathcal{Q}}_{33}}{\xi_2} \end{bmatrix} \\ \sum \begin{bmatrix} \frac{\tilde{\mathcal{Q}}_{11}}{\xi_1} & 0 & 0 \\ 0 & \frac{\tilde{\mathcal{Q}}_{22}}{\xi_2} & \frac{\tilde{\mathcal{Q}}_{23}}{\xi_2} \\ 0 & \frac{\tilde{\mathcal{Q}}_{23}}{\xi_2} & \frac{\tilde{\mathcal{Q}}_{33}}{\xi_2} \end{bmatrix} \end{cases} \text{ for Figure 4.1(c),} \tag{4.5}$$

$$(\Delta\tilde{O})^2 = \sum \begin{bmatrix} \frac{\tilde{\mathcal{Q}}_{11}}{\xi_1} & \frac{\tilde{\mathcal{Q}}_{12}}{\xi_1+\xi_2} & 0 \\ \frac{\tilde{\mathcal{Q}}_{12}}{\xi_1+\xi_2} & \frac{\tilde{\mathcal{Q}}_{22}}{\xi_1+\xi_2} & \frac{\tilde{\mathcal{Q}}_{23}}{\xi_1+\xi_2} \\ 0 & \frac{\tilde{\mathcal{Q}}_{23}}{\xi_1+\xi_2} & \frac{\tilde{\mathcal{Q}}_{33}}{\xi_2} \end{bmatrix} \text{ for Figure 4.1(d),} \tag{4.6}$$

where the two alternatives in the curly bracket in Eq. (4.5) are used to indicate the two possible cases where one clique is maximal and the other is not, as depicted in Figure 4.1(c). These matrices correspond to the ones pictorially represented in Figure 4.1(b-d), and fully determine the expected error $(\Delta\tilde{O})^2$ for any choice of ξ_1, ξ_2 and (for the cover without maximal cliques) ξ_3 [see Eqs.(4.2)]. These last parameters represent the number of times that the corresponding clique is measured, and can be obtained from m_i and m_{ij} (and vice versa, see Section 4.8). From Eqs. (4.4), (4.5) and (4.6) it is possible to determine the cliques in which the Paulis are by looking at the subscripts of the parameters ξ_j in the diagonal elements. If $\tilde{\mathcal{Q}}_{ii}$ is divided by ξ_1, ξ_2 or ξ_3, it means that the Pauli operator, $\mathtt{P}_i$, belongs to the first, second or third clique, respectively. Similarly, in Eq. (4.6), the element $\tilde{\mathcal{Q}}_{22}/(\xi_1 + \xi_2)$ indicates that the Pauli $\mathtt{P}_2$ is measured any time either clique is probed, in agreement with Figure 4.1(d).

In order to determine the optimal allocation of the M measurements for any of the covers in Figure 4.1(b-d), we minimize the corresponding error $(\Delta\tilde{O})^2$ in Eqs. (4.4), (4.5) or (4.6) with the additional constraints that $\sum_i \xi_i = M$ and that ξ_i are non-negative integers for all i. We note that a similar problem (without allowing for an overlap of cliques) has been studied in Ref. [119]. Since we are assuming that we know the exact values of $\tilde{\mathcal{Q}}_{ij}$, the errors $(\Delta\tilde{O})^2$ derived with the optimal ξ_i are the minima of the associated cover that can be achieved with the considered input state ρ_{in} and the measurement budget M. Furthermore, the four covers presented in Figure 4.1(b-d) are all non-redundant covers that can be found for the operator O in Eq. (4.3), meaning that out of these four $(\Delta\tilde{O})^2$, the smallest one is the minimal error $(\Delta\tilde{O})^2_{\text{opt}}$ that can be generally obtained when measuring O with M shots with respect to ρ_{in}.

Once the lowest possible errors associated to each cover have been determined, it is possible to test whether there is one cover always resorting to the minimal error $(\Delta\tilde{O})^2_{\text{opt}}$. One may think that the cover in Figure 4.1(d), consisting of maximal cliques only, always outperforms the others, since it probes more Paulis per shot. However, this is not the case due to possibly positive covariances $\tilde{\mathcal{Q}}_{ij}$ $(i \neq j)$ in the off-diagonal terms in Eq. (4.6)[4]. In fact, it turns out that the error from the cover made of maximal cliques only equals the minimal $(\Delta\tilde{O})^2_{\text{opt}} \sim 54\%$ of the times. The other $\sim 46\%$ of times $(\Delta\tilde{O})^2_{\text{opt}}$ corresponds to one of the two covers in Figure 4.1(c).

Despite not always being the optimal one, the cover made of maximal cliques does outperform all other covers on average. This motivated the choice of using maximal cliques only when probing the system, along with the post-processing unit afterwards for removing vertices from the cliques based on their covariances and their contribution to $(\Delta\tilde{O})^2$ within the considered clique. In fact, by comparing Figure 4.1(e) with Figure 4.1(b-d), we see that on average AEQuO provides smaller errors if compared to the theoretical minima that can be obtained with the four covers considered in the figure.

4.7 Bayesian estimation for the graph

In this section, we use Bayesian inference [120, 121] to estimate the averages, variances and covariances of the Paulis spanning O. We consider a pair of Pauli operators, $\mathtt{P}_j$ and $\mathtt{P}_i$, whose measurements are collected in the dataset D. From D, it is possible to obtain the following quantities. We denote by $s^{\pm}_{\lambda}$ (with $s^{+}_{\lambda} + s^{-}_{\lambda} = m_{\lambda}$) the total number of times an outcome ± 1 is collected from measurements of P_{λ} for $\lambda = i, j$. The associated underlying (but unknown) probability is denoted by $\theta_{\pm}(\lambda)$, where the argument λ will be dropped for clarity whenever it is clear from context. Similarly, $s^{\pm\pm}_{ij}$ (with $s^{++}_{ij} + s^{+-}_{ij} + s^{-+}_{ij} + s^{+-}_{ij} = m_{ij}$) is the number of times that $\mathtt{P}_i$ and $\mathtt{P}_j$ yielded the corresponding $\{\pm 1, \pm 1\}$ outcome, which is characterized by a probability $\theta_{\pm\pm}$. Finally, $w^{\pm}_{\lambda}$ is the number of occurrences of ± 1, that refers to the cases where $\mathtt{P}_i$ and $\mathtt{P}_j$ have been probed independently. It follows that $w^{+}_{i} + w^{-}_{i} = m_i - m_{ij}$ and $w^{+}_{j} + w^{-}_{j} = m_j - m_{ij}$.

[4] We remark that, in general, the best measurement strategy is the one measuring *all* Pauli operators at once. However, when some of the Paulis do not commute and as such their covariances cannot contribute to $(\Delta\tilde{O})^2$, it is sometimes advantageous *not* to measure a Pauli for the reasons explained in Section 4.3.

Using the collected data D, our goal is to estimate the posterior probability $P(\vec{\theta}|D)$ that best describes the parameters

$$\vec{\theta} = \{\theta_+(j), \theta_-(j), \theta_+(i), \theta_-(i), \theta_{++}, \theta_{+-}, \theta_{-+}, \theta_{--}\}$$

in our measurement process. By Bayes' Theorem,

$$P(\vec{\theta}|D) = \frac{P(D|\vec{\theta})P(\vec{\theta})}{P(D)}, \tag{4.7}$$

where the likelihood $P(D|\vec{\theta})$ is the probability of obtaining the dataset D given $\vec{\theta}$. The prior $P(\vec{\theta})$ is the probability of $\vec{\theta}$ before the current evidence or dataset D is observed. The probability $P(D)$ is also called the marginal likelihood.

Following standard procedures [120, 121], we consider $P(\vec{\theta}) = \mathrm{Dir}(K, \vec{a})$, where Dir indicates the Dirichlet distribution of order K and $\vec{a} = \{a_1, \ldots, a_K\}$ are its hyperparameters [120, 121]. Then $P(\vec{\theta}|D) = \mathrm{Dir}(K, \vec{c} + \vec{a})$, where $\vec{c} = \{c_1, \ldots, c_K\}$ is a vector giving the number of occurrences of each category in the dataset D.

The Dirichlet distribution of order K has the following probability density function:

$$P(\vec{\theta}) = P(\vec{\theta}; \vec{a}) = \frac{1}{B(\vec{a})} \prod_{i=1}^{K} \theta_i^{a_i - 1}, \tag{4.8a}$$

$$B(\vec{a}) = \frac{\prod_{i=1}^{K} \Gamma(a_i)}{\Gamma(\sum_{i=1}^{K} a_i)}, \tag{4.8b}$$

where $\Gamma(z) = \int_0^\infty x^{z-1} e^{-x} dx$ is the gamma function. The likelyhood function $P(D|\vec{\theta})$ then takes the form

$$P(D|\vec{\theta}) = \prod_{i=1}^{K} \theta_i^{c_i}, \tag{4.9}$$

which leads to the posterior

$$P(\vec{\theta}|D) = \frac{1}{B(\vec{a} + \vec{c})} \prod_{i=1}^{K} \theta_i^{c_i + a_i - 1}. \tag{4.10}$$

With the posterior probability $P(\vec{\theta}|D)$ in Eq. (4.10), we can determine the quantities of interest for the measurement protocol considered in this work by marginalization. In more detail, assume we are interested in estimating a certain quantity $f(\vec{\theta})$. We define its estimator $\tilde{f}(\vec{\theta})$ by

$$\tilde{f}(\vec{\theta}) = \int f(\vec{\theta}) P(\vec{\theta}|D) d\vec{\theta}. \tag{4.11}$$

This equation is used for determining $\tilde{\mathtt{P}_i}$ and the elements of the covariance matrix $\tilde{\mathcal{Q}}$ (see Section 4.3).

In the following, we explicitly derive $\tilde{\mathtt{P}_i}$ and the variance $\tilde{\mathcal{Q}}_{ii}$ of a single Pauli operator, $\mathtt{P}_i$, in Section 4.7.1. In Section 4.7.2, we consider the case in which two Paulis, $\mathtt{P}_i$ and $\mathtt{P}_j$, are measured together, and we estimate their covariance. As explained in Section 4.7.3, this estimate for their covariance may lead to wrong errors $(\Delta\tilde{O})^2$ when M is small. Then, we obtain a valid estimate for $\tilde{\mathcal{Q}}_{ij}$ $(i \neq j)$ by considering the case in which the Paulis may also have been measured independently from each other. Finally, we summarize the results and explain how Bayesian estimation is used in AEQuO in Section 4.7.4.

4.7.1 Single Pauli Operator

Whenever we measure a single Pauli operator, $\mathtt{P}_i$, we obtain a dataset D which is a collection of $+1$ and -1, from which s_i^+ and s_i^- can be found. In this case, $K = 2$, $\vec{\theta} = \{\theta_+, \theta_-\}$ with $\theta_- = 1 - \theta_+$, and for simplicity we set $D \mapsto \vec{c} = \{s_i^+, s_i^-\}$. From Eq. (4.10) we get the posterior probability

$$P(\{\theta_+, \theta_-\}|\{s_i^+, s_i^-\}) = \frac{\theta_+^{s_i^+}(1-\theta_+)^{s_i^-}}{B(\{s_i^+ + 1, s_i^- + 1\})}, \tag{4.13}$$

where we set $a_1 = a_2 = 1$. This choice for $\vec{a}$, corresponding to a uniformly distributed prior, is generally taken when no information is available prior to the measurement.

The exact values $\langle \mathtt{P}_i \rangle$ and $(\Delta \mathtt{P}_i)^2$ of the average and variance of the Pauli, $\mathtt{P}_i$, can be expressed in terms of the probabilities $\vec{\theta}$ as

$$\langle \mathtt{P}_i \rangle = \theta_+ - \theta_- = 2\theta_+ - 1, \tag{4.14a}$$

$$(\Delta \mathtt{P}_i)^2 = 4\theta_+\theta_- = 4\theta_+(1-\theta_+). \tag{4.14b}$$

Using these equations in place of $f(\vec{\theta})$ in Eq. (4.11), we obtain the following estimates $\tilde{\mathtt{P}_i}$

and $\tilde{\mathcal{Q}}_{ii}$:

$$\begin{aligned}\tilde{\mathtt{P}}_i &= \int_0^1 \langle \mathtt{P}_i \rangle P(\{\theta_+, \theta_-\} | \{s_i^+, s_i^-\}) d\theta_+ \\ &= \frac{s_i^+ - s_i^-}{s_i^+ + s_i^- + 2},\end{aligned} \tag{4.15a}$$

$$\begin{aligned}\tilde{\mathcal{Q}}_{ii} &= \int_0^1 (\Delta \mathtt{P}_i)^2 P(\{\theta_+, \theta_-\} | \{s_i^+, s_i^-\}) d\theta_+ \\ &= 4 \frac{\left(s_i^+ + 1\right)\left(s_i^- + 1\right)}{\left(s_i^+ + s_i^- + 2\right)\left(s_i^+ + s_i^- + 3\right)}.\end{aligned} \tag{4.15b}$$

These relations are used by our algorithm to update vertices and self-edges in the graph [see, e.g., Figure 4.1(a)].

4.7.2 Two Paulis always measured together

The procedure used in the previous section to determine $\tilde{\mathtt{P}}_i$ and $\tilde{\mathcal{Q}}_{ii}$ can be repeated here for the covariance $\tilde{\mathcal{Q}}_{ij}$ $(i \neq j)$, provided we have meaningful likelihood and prior functions. In this case, we deal with two Pauli operators, $\mathtt{P}_i$ and $\mathtt{P}_j$, and as such we have four possible outcomes described by the probabilities θ_{++}, θ_{+-}, θ_{-+} and $\theta_{--} = 1 - \theta_{++} - \theta_{+-} - \theta_{-+}$. Furthermore, we assume here that $\mathtt{P}_i$ and $\mathtt{P}_j$ are always measured together; the generalization is presented in Section 4.7.3. For the sake of clarity, we have $K = 4$, $\vec{\theta} = \{\theta_{++}, \theta_{+-}, \theta_{-+}, \theta_{--}\}$ and $D \mapsto \vec{c} = \{s_{ij}^{++}, s_{ij}^{+-}, s_{ij}^{-+}, s_{ij}^{--}\}$. From Eq. (4.10) we have the following posterior probability,

$$\begin{aligned}P(\vec{\theta}|D) = {}& \theta_{++}^{s_{ij}^{++} + a_{++} - 1} \theta_{+-}^{s_{ij}^{+-} + a_{+-} - 1} \theta_{-+}^{s_{ij}^{-+} + a_{-+} - 1} \\ & \times (1 - \theta_{++} - \theta_{+-} - \theta_{-+})^{s_{ij}^{--} + a_{--} - 1} \frac{1}{B(\vec{a} + \vec{c})},\end{aligned} \tag{4.17}$$

where we left the parameters $\vec{a} = \{a_{++}, a_{+-}, a_{-+}, a_{--}\}$ in their explicit form for reasons that will become clear in the following Section 4.7.3. Here, we assume $a_{++} = a_{+-} = a_{-+} = a_{--} = 1$ similar to the single Pauli, since there is no information available prior the measurements.

By expressing the exact covariance $\mathrm{cov}(\mathtt{P}_i, \mathtt{P}_j)$ in terms of the probabilities $\vec{\theta}$ as

$$\begin{aligned}\mathrm{cov}(\mathtt{P}_i, \mathtt{P}_j) &= 4(\theta_{++}\theta_{--} - \theta_{+-}\theta_{-+}) \\ &= 4\left[\theta_{++}(1 - \theta_{++} - \theta_{+-} - \theta_{-+}) - \theta_{+-}\theta_{-+}\right],\end{aligned} \tag{4.19}$$

and using Eqs. (4.11) and (4.17), we can determine a possible estimate $\tilde{\mathcal{Q}}_{ij}$ $(i \neq j)$,

$$\begin{aligned}\tilde{\mathcal{Q}}_{ij} &= \int_0^1 d\theta_{++} \int_0^{1-\theta_{++}} d\theta_{+-} \int_0^{1-\theta_{++}-\theta_{+-}} d\theta_{-+} \operatorname{cov}(\mathtt{P}_i, \mathtt{P}_j) P(\vec{\theta}|D) \\ &= 4\frac{(s_{ij}^{++}+1)(s_{ij}^{--}+1)-(s_{ij}^{+-}+1)(s_{ij}^{-+}+1)}{(s_{ij}^{++}+s_{ij}^{+-}+s_{ij}^{-+}+s_{ij}^{--}+4)(s_{ij}^{++}+s_{ij}^{+-}+s_{ij}^{-+}+s_{ij}^{--}+5)}.\end{aligned} \tag{4.20}$$

This equation, while representing a valid estimate for the covariance, is *not* used in our algorithm to obtain $\tilde{\mathcal{Q}}_{ij}$. Indeed, since we allow for overlap among the cliques, there are cases in which two Paulis are both measured individually and together, and Eq. (4.20) disregards all outcomes in which $\mathtt{P}_i$ and $\mathtt{P}_j$ are uncorrelated. In the following section, we study two possibilities on how to include these events into the estimation of $\tilde{\mathcal{Q}}_{ij}$, such that we do not neglect any collected information on the system and always obtain a meaningful estimate of the error $(\Delta\tilde{O})^2$.

4.7.3 Two Paulis measured together and individually

As anticipated at the end of the previous section, Eq. (4.20) ignores all events where $\mathtt{P}_i$ and $\mathtt{P}_j$ were not measured together. This has two implications. First, we are discarding available information that can be used for getting a more precise estimate for the covariance. Second, and more importantly, in case of scarce statistics Eqs. (4.20) and (4.15) may lead to a wrong estimate for the error $(\Delta\tilde{O})^2$. This is because the variances $\tilde{\mathcal{Q}}_{ii}$ and the covariances $\tilde{\mathcal{Q}}_{ij}$ $(i \neq j)$ are derived independently from each other, ignoring the fact that they are correlated. In fact, it is possible to express the probabilities $\theta_{\pm\pm}$ in terms of $\theta_\pm(i)$, $\theta_\pm(j)$, and the exact $\operatorname{cov}(\mathtt{P}_i, \mathtt{P}_j)$:

$$\theta_{++} = \theta_+(i)\theta_+(j) + \frac{\operatorname{cov}(\mathtt{P}_i, \mathtt{P}_j)}{4}, \tag{4.21a}$$

$$\theta_{+-} = \theta_+(i)\theta_-(j) - \frac{\operatorname{cov}(\mathtt{P}_i, \mathtt{P}_j)}{4}, \tag{4.21b}$$

$$\theta_{-+} = \theta_-(i)\theta_+(j) - \frac{\operatorname{cov}(\mathtt{P}_i, \mathtt{P}_j)}{4}, \tag{4.21c}$$

$$\theta_{--} = \theta_-(i)\theta_-(j) + \frac{\operatorname{cov}(\mathtt{P}_i, \mathtt{P}_j)}{4}, \tag{4.21d}$$

where we recall that the subscripts ± in $\theta_{\pm\pm}$ refer to Pauli operators $\mathtt{P}_i$ and $\mathtt{P}_j$, in this order. Since the relations in Eqs. (4.21) are not always satisfied by Eqs. (4.20) and (4.15), it may happen that in measuring an observable we obtain an estimate resulting in a wrong error $(\Delta\tilde{O})^2$. As an example, assume that we take 25 measurements of each of P_1 and P_2 independently, and always get $+1$ and -1 outcomes, respectively. Afterwards, we measure these Paulis together twice and get the two pairs $\{-1,+1\}$ and $\{+1,-1\}$. In this case, our estimates for the variances will be small, while we would get a comparatively large negative covariance. This leads to a negatively defined covariance matrix $\tilde{\mathcal{Q}}$ such that $\tilde{\mathcal{Q}}_{11}+\tilde{\mathcal{Q}}_{22}+2\tilde{\mathcal{Q}}_{12}=-0.07$, which is not physically allowed.

4.7.4 Summary

To summarize, we resorted to Bayesian theory to estimate averages, variances and covariances of Paulis being measured together and/or individually. This is crucial for allocating the measurement budget to different cliques, since our estimators yield valid results with scarce or even no statistics.

Another feature of our algorithm is that, *after* all measurements have been allocated and taken, it offers different options for calculating the estimator $\tilde{O}$ and its error $(\Delta\tilde{O})^2$. The user can substitute the Bayesian estimators for $\tilde{\mathtt{P}}_i$ and $\tilde{\mathcal{Q}}_{ij}$ in Eqs. (4.15), and the sample averages, variance and covariance, respectively, at the end of the algorithm [122]. This is a reasonable choice since these sample quantities are known to converge faster if compared to the Bayesian ones. We remark, however, that this substitution is reasonable if and only if there is a sufficiently large dataset for correctly estimating all contributions in Eq. (4.2b).

Yet another feature of our algorithm is that it allows, both when allocating the measurement budget and/or when outputting the final value for estimate and error, to use the exact values $\mathrm{cov}(\mathtt{P}_i,\mathtt{P}_j)$ for $\tilde{\mathcal{Q}}_{ij}$ in Eq. (4.2b). On the one hand, this allows for investigating the limits of our protocol. On the other hand, it is useful to determine the exact error (i.e., without statistical fluctuations) that is expected from a given measurement allocation, as we did for the values in squared brackets in Figure 4.3(a) and the points in Figure 4.3(c).

4.8 An efficient method to compute $(\Delta\tilde{O})^2$

In this section, we explain a computationally efficient method to derive the error $(\Delta\tilde{O})^2$. As evident from Eq. (4.2), this involves two objects, the matrix C (with elements $c_i c_j \tilde{\mathcal{Q}}_{ij}$) and the parameters m_{ij} $(i,j=1,\ldots,n)$. The first one can be obtained by following the procedure explained in Section 4.7. The latter one is found from the outputs of either the

BF or the ML subroutine, as explained in Section 4.4. Indeed, these subroutines provide an r-dimensional vector $\vec{\xi}$ whose elements ξ_j represent the number of times that the j-th clique is measured. To derive m_{ij} from $\vec{\xi}$, one may use the measurement characteristic matrix $\mathcal{E}$, which is an $(n \times r)$-matrix whose elements $\mathcal{E}_{ij}$ are equal to 1 if the Pauli string i belongs to the clique j, and zero otherwise. It then follows that m_{ij} is the (i, j)-element of the product $\mathcal{E}\Xi\mathcal{E}^\top$, where Ξ is a diagonal $(r \times r)$-matrix with $\vec{\xi}$ as the diagonal entries, and $^\top$ indicates transposition.

Given m_{ij}, the error $(\Delta\tilde{O})^2$ can be rewritten as a function of matrix operations that can be efficiently implemented numerically. Denoting the Hadamard product and Hadamard division [123] with $\circ$ and $\oslash$, respectively, we have

$$(\Delta\tilde{O})^2 = j^\top(C \circ \mathcal{E}\Xi\mathcal{E}^\top \oslash \mathcal{E}\Xi J \Xi \mathcal{E}^\top)j, \tag{4.22}$$

where J is the all-ones matrix of size $(r \times r)$, and j is the $(r \times 1)$-all-ones vector. Given C and $\vec{\xi}$, this equation is used in the BF and the ML subroutines of our algorithm to efficiently compute the error function. Notice that, compared to Eq. (4.2), the Kronecker delta does not appear in Eq. (4.22). This is not problematic since the BF [ML] subroutine considers the elements in $\vec{\xi}$ to be positive integers [real numbers that are then rounded to the nearest integer]. For the post-processing we directly use Eq. (4.2) to calculate $(\Delta\tilde{O})^2$.

4.9 Numerical results

In this section, we explain the numerical results in Figure 4.3 and 4.4. This includes details on the LDF algorithm that we have used, and an extended explanation about how the values in the tables, as well as the points in the plots, have been derived.

The numbers outside the square brackets in Figure 4.3(a) represent the standard deviations

$$\Sigma \equiv \sqrt{\frac{\sum_{j=1}^{R}(\tilde{O}_j - \langle O \rangle)^2}{R}}, \tag{4.23}$$

where $\tilde{O}_j$ is the j-th estimated average of the observable O under consideration, $\langle O \rangle$ is the exact average of O, and R is the total number of times the whole measurement procedure is repeated [$R = 40$ in Figure 4.3(a)]. For the Derand [37] and the APS [41, 42] methods, $\tilde{O}_j$ are determined with the algorithms reported in the associated references. The same holds for the OGM [43] method, where we resorted to version 2 of their algorithm and followed their sampling procedure.

As explained in Section 4.7.4 and in [108], one can choose different quantities to be returned by AEQuO. The $\tilde{O}_j$ in Eq. (4.23) for calculating the numbers outside the square brackets are obtained with the sample average [122] for estimating all Paulis, $\tilde{\mathtt{P}}_i$, within $\tilde{O}_j$. The numbers inside the square brackets in Figure 4.3(a), on the other side, are derived by averaging the values $\sqrt{(\Delta\tilde{O})^2}$ obtained in the R iterations of the algorithm. The associated errors in the parentheses are the sample root mean squares [122]. While AEQuO performs the measurement allocation without prior knowledge of the covariance matrix $\tilde{\mathcal{Q}}$, the values for $\sqrt{(\Delta\tilde{O})^2}$ that are returned at each j-th iteration are calculated with the exact $\tilde{\mathcal{Q}}_{ij} \to \mathrm{cov}(\mathtt{P}_i, \mathtt{P}_j)$. This allows to have a very precise estimate of the real error (hence the small uncertainties in the parentheses), even with only $R = 40$.

In panels (b) and (c) of Figure 4.3 we show the rescaled errors $M\Sigma^2/\langle O\rangle^2$ and $M(\Delta\tilde{O})^2/\langle O\rangle^2$, respectively. In (b), $M\Sigma^2/\langle O\rangle^2$ is chosen in order to compare AEQuO with other approaches that cannot directly estimate the variance of the considered observable O. In (c), we report $M(\Delta\tilde{O})^2/\langle O\rangle^2$, where $(\Delta\tilde{O})^2$ and the associated error bars are calculated in the same way as the values in the square brackets within Figure 4.3(a) (see previous paragraph). We have chosen $R = 100$ to reduce statistical fluctuations affecting Σ in (b), and $R = 40$ in (c). Furthermore, since the APS method and AEQuO with the BF subroutine allocate the measurement shots one at the time, when M is large they require longer runtimes than the other approaches. Therefore, they have been run up to $M = 6400$. Every time AEQuO is used to determine the $\tilde{\mathcal{Q}}$ used for measurement allocation, it resorts to Bayesian estimation with $\beta_1 = 0$ (see Section 4.7.4).

For observables characterized by small values of N, AEQuO can find all maximal cliques and feed them into either the BF or the ML subroutines for measurement allocation. This procedure led to the reported values in Figure 4.3(a) until (excluded) the LiH$_2$ Hamiltonian. Since the worst-case scaling of r is exponential in the number of vertices [124], resorting to all maximal cliques is not an option when N is large. We thus developed an algorithm that spans all vertices and, for each, finds a user-specified number of maximal cliques (see Section 4.4 and [108] for more informations). When building each maximal clique, this algorithm favours vertices with more edges and higher contributions to the cost function $(\Delta\tilde{O})^2$ in Eq. (4.2b). This creates a bias towards larger cliques, that in the context of measuring a quantum observable are highly desirable. In fact, we have tested AEQuO when using either all, or a subset of $r \lesssim N$ (large) maximal cliques. For the examples we considered, the errors yielded in these cases differed by 15% at most.

The values reported in Figure 4.3(a-b) for the LDF method have been determined with an algorithm based on Ref. [31] and integrated with the framework developed in this work. We first find a cover of the graph such that each vertex is contained in exactly one clique. To find this cover, we also prioritise larger cliques and vertices with higher $(\Delta\tilde{O})^2$ contributions, as we did for the maximal cliques' algorithm described in the previous paragraph. Once a cover is found, we use the BF subroutine to allocate all measurements, without ever updating the covariance matrix $\tilde{\mathcal{Q}}$, i.e., no adaptive features are used for the LDF method. We then perform the measurements, and resorting to the sample average we obtain $R = 40$ [$R = 100$] values $\tilde{O}_j$ that are used in Eq. (4.23) to determine the numbers reported in Figure 4.3(a) and 4.3(c) [Figure 4.3(b)].

The numerical results from the LDF method reported in Section 4.5 are lower if compared to the ones reported in Refs. [31, 34, 41, 43, 37]. This is a consequence of two facts. First, for each graph cover, our measurement allocation is optimal (as can be demonstrated by following the procedure in Section 4.6). Second, we allow for general commutation relations between Paulis, i.e., we group together two Pauli operators that commute, but do not necessarily bitwise commute. Indeed, when we restrict our LDF to bitwise commutation relations (as in Section 4.10), it yields results that are similar to the ones reported for the LDF algorithm in Refs. [31, 34, 41, 43, 37].

Our version of the LDF is representative of other measurement protocols (such as the ones in Refs. [31, 46, 47, 48, 45]) that also allow for non-bitwise commuting relations between Paulis. In fact, prior to this work two criteria were commonly used to collect Paulis. First, the magnitude of their coefficients and second, the total number of resulting groups. For building the groups of Paulis to be measured together, our LDF protocol employs the expected contributions of the Paulis to the error, that is available via Eqs. (4.2) and results in a strategy that is similar to the one used by AEQuO (described in Section 4.4 and [108]). Since having less groups and collecting together Paulis with large coefficients are highly correlated with having lower estimation errors, we expect the algorithms in Refs. [31, 46, 47, 48, 45] to achieve errors that are comparable to the numerical LDF results reported in Section 4.5.

In all panels of Figure 4.4, we report averaged values of $(\Delta\tilde{O})^2$ with their statistical errors [in parentheses in (a) and as error bars in (b-e)] and $R = 25$ [see above discussion about Figure 4.3(c)]. The same chemistry Hamiltonians previously utilized in Figure 4.3 have been used in Figure 4.4(a), while we resorted to the family of lattice models introduced in the main text and Section 4.11 for Figure 4.4(b-d). In (a), (b) and (d), the settings chosen for estimating $\tilde{\mathcal{Q}}$ are the same ones used in Figure 4.3. For (c) and (e) the covariance matrix is initialized to the exact values, as explained at the bottom of Section 4.7.4.

For all numerical values in Figure 4.3 where the ML subroutine has been utilized, we employed a stochastic gradient descent optimization algorithm [113] for the cost function $(\Delta\tilde{O})^2$ minimization. This algorithm's performance depends on hyperparameters [108, 111, 113] – the learning rate in particular – that have not been optimized to keep low computational requirements. On the other hand, the ML subroutine in Figure 4.4 employs the Limited-memory BFGS optimizer [125] that, belonging to the family of quasi-Newton methods [115], does not require choosing the learning rate.

4.10 Bitwise commutation comparison

In this section, we compare AEQuO and our LDF protocols both restricted to bitwise commutation (BC) relations with the APS [41, 42], the Derand [37] and the OGM [43] methods. In fact, as explained in Section 4.4, simultaneous measurements of non-bitwise commuting Pauli operators require entangling gates that, in the context of NISQ devices, generate errors that can jeopardize the result. As such, it is important to determine the advantage resulting from the adaptive nature of AEQuO alone.

In Figure 4.5 we present analogous numerical results as in Figure 4.3(a-b), with the additional constraint that all measurement schemes are restricted to bitwise commutation relations. From panel (a), it is possible to conclude that for the considered chemistry Hamiltonians the adaptive nature of AEQuO does provide a consistent advantage over all other schemes. Except for the water molecule, non-adaptive ($L = 1$, black dotted line) AEQuO, the OGM, and the Derand protocols yield results that are within statistical fluctuations of one another (in agreement with Figure 4.5(b) and Figs. 4.3 and 4.4). Furthermore, while the LDF protocol in Figure 4.3(a-b) was yielding results that, in several instances, were better than the OGM and the Derand, here the LDF always performs comparably or worse. We thus conclude that its advantage came from non-bitwise commutation between Paulis that, in this section, is not allowed.

The case of H_2O in Figure 4.5(a) presents an interesting feature. The OGM, while performing worse than adaptive ($L = 3$, $l = 4$) AEQuO, outperforms all other approaches. After a careful analysis, we identified the reason of this advantage in the fact that the OGM protocol, for small measurement budgets, does not measure the input observable. Instead, it removes Paulis with small coefficients from the Hamiltonian [43]. These terms, that are considered and measured by AEQuO, the LDF and the Derand protocols, introduce extra statistical fluctuations that increase the error. This is supported by Figure 4.5(b), where it is evident that the OGM (yellow downwards triangles) asymptotically reaches (within statistical fluctuations) the Derand (red upwards triangles) and non-adaptive AEQuO ($L = 1$, black dotted line) for large values of M, when all Paulis are considered by OGM.

To confirm that the OGM's enhanced performance for small M comes from neglecting Paulis that bare extra statistical errors, we investigated the performances of all protocols

a) $10^{-4}\times$	**LDF** **BC** [16]	**OGM** [27]	**APS** [25,26]	**Derand** [21]	**AEQuO** **BC**, L=1	**AEQuO** **BC**, L=3, l=4
H_2(8 BK)	5.6	1.6	53.8	3.3	2.2[2.1(0)]	**0.9**[0.9(0)]
LiH(12 BK)	7.2	1.2	39.1	2.3	1.5[1.8(0)]	**0.8**[1.1(0)]
BeH_2(14 BK)	16.0	7.1	232.6	11.7	7.3[8.7(1)]	**3.6**[4.6(1)]
H_2O(14 BK)	353	36	211	71	91 [63(0)]	**27** [22(0)]

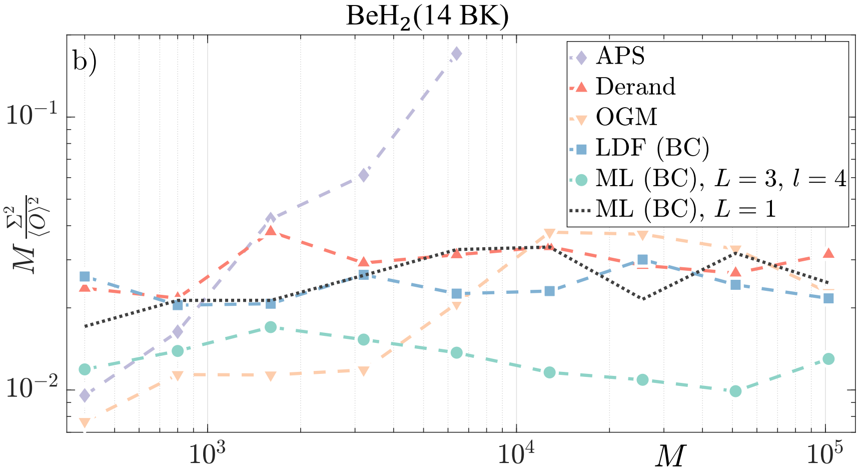


Figure 4.5: (a): Errors obtained with LDF [31], OGM [43], APS [41, 42], Derand [37], and AEQuO with the ML subroutine and $L = 1$ or $L = 3$ and $l = 4$. We consider chemistry Hamiltonians in the BK encoding [22]. Values (in bold the lowest) are variances Σ^2 with Σ as in Eq. (4.23). In square brackets, we report averages of $(\Delta\tilde{O})^2$ with their statistical error. The input is the ground state, $R = 25$, $M = 10^4$, and values are rescaled by 10^{-4}. (b): Relative errors $M\Sigma^2/\langle O\rangle^2$ as a function of M obtained considering the BeH$_2$ Hamiltonian with BK encoding and $R = 10^2$. In both panels, AEQuO and the LDF [31] protocol are restricted to bitwise commutation (BC) relations.

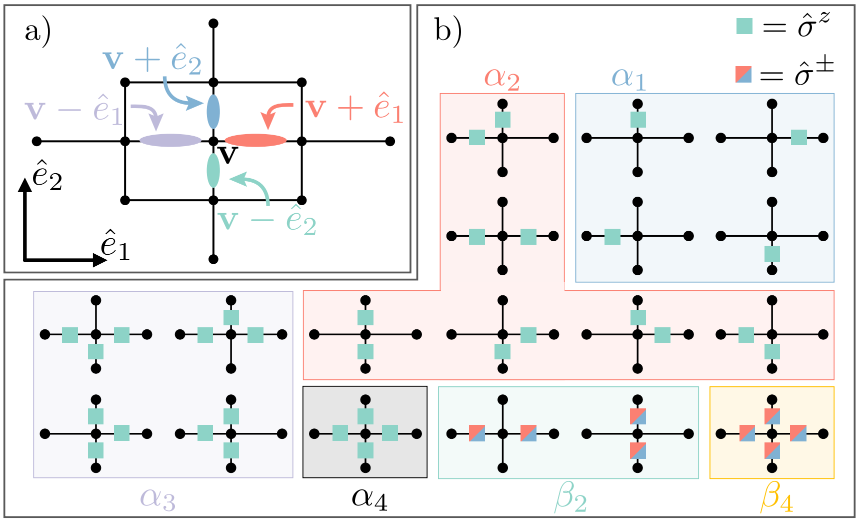


Figure 4.6: (a): Example of a lattice used in Figure 4.4(b-d). Here, we have $D = 2$ dimensions and $n = 16$. Vertices are indicated by their coordinates with vectors $\mathbf{v}$. Qubits lie on edges (coloured ellipses), and are identified by $\mathbf{v} \pm \hat{e}_i$ $(i = 1, \ldots, D)$ with $\hat{e}_i$ being unit versors defining the D-dimensional Cartesian plane. (b): All terms in the Hamiltonian describing the $D = 2$ dimensions model with $n = 4$ qubits. Green squares indicate Z interactions, while red/blue ones the flip-flop terms. We grouped together all interactions characterized by the same coefficients [α_k and β_{2k} in Eq. (4.25)].

with different input states. For small values of M, we have found several instances in which the OGM yields results that are comparable to the Derand and non-adaptive ($L = 1$) AEQuO.

4.11 Lattice Hamiltonians

For deriving the data in Figure 4.4(b-d), we considered different 2- and 3-D lattice models (see bottom of the figure), described by the Hamiltonians

$$\mathcal{H} = \sum_{\mathbf{v},i} \left[\sum_{k=1}^{2D} \frac{\alpha_k}{2} \prod_{\substack{S\subseteq\{\hat{\mathbf{e}},-\hat{\mathbf{e}}\} \\ \hat{\imath}\in S}}^{|S|=k} \hat{\sigma}^z_{\mathbf{v}+\hat{\imath}} + \sum_{k=1}^{D} \beta_{2k} \prod_{\substack{S\subseteq\{\hat{\mathbf{e}}\} \\ \hat{\imath}\in S}}^{|S|=k} \left(\hat{\sigma}^+_{\mathbf{v}+\hat{\imath}} \hat{\sigma}^-_{\mathbf{v}-\hat{\imath}} + \hat{\sigma}^-_{\mathbf{v}+\hat{\imath}} \hat{\sigma}^+_{\mathbf{v}-\hat{\imath}} \right) \right]. \tag{4.25}$$

To better explain this operator, we refer to Figure 4.6. In panel (a), we show an example of a lattice in $D = 2$ dimensions with $n = 16$ qubits lying on the edges. Vertices are indicated by vectors $\mathbf{v}$ and the Cartesian coordinate frame is characterized by versors $\{\hat{\mathbf{e}}\} = \{\hat{e}_1, \ldots, \hat{e}_D\}$. It follows that each qubit can be identified by $\mathbf{v} \pm \hat{e}_j$ $(j = 1, \ldots, D)$, where $\mathbf{v}$ is the vertex from which it can be reached by one of the versors (see Figure 4.1). In Eq. (4.25), α_k and β_{2k} are real constants, and the products run over all possible combinations of versors (S are subsets of k oriented versors). The raising and lowering operators are indicated with $\hat{\sigma}^{\pm} = (\mathtt{X} \pm i\mathtt{Y})/2$. In Figure 4.6(b), all elements of the Hamiltonian in Eq. (4.25) are represented for a $D = 2$ dimensional lattice with $n = 4$ qubits.

This class of physical models, describing many-body, next-neighbouring interacting spin-1/2 particles on a lattice is a generalization of the Hubbard model [118] with hopping multi-particle terms and energy shifts depending on the particles' state. For determining the data points in Figure 4.4(b-e), we set $\alpha_k = 1/k$ and $\beta_{2k} = 1/(2k)$.

4.12 Conclusions and outlook

We introduced AEQuO, an adaptive algorithm to estimate quantum observables expressed as a sum of Pauli operators. By allowing for both overlap between groups of Pauli operators that are simultaneously measured and general commutation relations, we gather more information per shot and introduce possibly negative covariances into Eqs. (4.2). AEQuO is capable of allocating the remaining measurement budget depending on previous outcomes, it post-processes the estimator to lower its error, and it gives estimates both for the average $\tilde{O}$ and the variance $(\Delta\tilde{O})^2$ of the considered observable. Our protocol is based on two routines which provide either near-optimal (BF) or computationally efficient (ML) allocation of the measurement budget. For the observables considered here, both subroutines result in similar errors $(\Delta\tilde{O})^2$.

We tested our algorithm on several Hamiltonians with different settings that include allowing or forbidding non-bitwise commutation between Paulis, and turning the adaptive allocation on $(L > 1)$ or off $(L = 1)$. Our protocol yields numerical results that outperform state-of-the-art estimators based on various grouping techniques [31, 34, 41, 43] and the classical shadow method [41, 37, 42]. We also studied the advantages our algorithm gains from adaptive allocation as well as non-bitwise commutation. We found that, while being highly problem dependent, these advantages are consistent.

There are different possibilities for improving our algorithm. Generating the list of all maximal cliques of the graph is computationally demanding for large problem instances. In this case, one could find better strategies to select a subset of (maximal) cliques, and this subset can in principle be updated while measurements are taken. Designing a better method of choosing suitable cliques could increase the performance of our algorithm without sacrificing the quality of the results. The ML-based subroutine could be further improved by employing graph neural networks to leverage the graph structure of the problem [117]. Another possibility is the extension of our adaptive algorithm to non-qubit-based hardware and, in view of the Bayesian framework underlying our estimation, the direct inclusion of the experimental characteristics of the considered platform via hierarchical modelling [126].

The algorithms developed in this chapter were coded in Python for a custom-built VQE measurement simulator and have been made available at [108].

Chapter 5

Minimizing Entangling Gates in Clifford Circuits

In the last chapter, we showed how to employ entangling Clifford gates to decrease the number of measurements in the VQE. A natural question about the viability of such approaches, though, is the problem of the fidelity of entangling gates on NISQ devices. In general, on NISQ devices, entangling gates are likely to have significantly lower fidelity than single-qubit gates. For this reason, it's important to weigh the tradeoffs of requiring fewer measurements against the errors introduced by using lower fidelity gates.

For any particular device and architecture, these tradeoffs will be weighed differently, which makes it impossible to choose an approach which is optimal in all circumstances. An approach which allows for a small number of entangling gates, limited by the architecture of the device and the fidelity of the gates, could likely prove optimal. For at least one error model and class of Hamiltonians, it has been shown that this tradeoff favors general commutation [127]. To avoid delving into the specifics of any current devices in this chapter, we will develop a framework for minimizing entangling gates in Clifford circuits which will be applicable to a variety of architectures, and can be used to help assess the aforementioned tradeoffs.

To begin, we will look at the theoretical question of the maximum number of entangling gates required in an n-qubit Clifford circuit. After proving a matching upper and lower bound for this result, we will develop a framework for constructing a Clifford circuit mapping between two specified stabilizer states using the minimum possible number of entangling gates. We will conclude with identifying a particularly interesting class of stabilizer states which can be constructed from the ground state using $\mathcal{O}(n \log n)$ entangling gates.

5.1 Matching Upper and Lower Bounds

The first direction we will consider is proving an upper bound on the number of entangling gates required to implement a specified Clifford circuit. This question was already considered and solved by Aaronson and Gottesman in their canonical form theorem.

Theorem 2 (Aaronson and Gottesman [78]) *Any Clifford circuit admits an equivalent Clifford circuit in canonical form. A Clifford circuit in canonical form consists of successive layers containing only* H, S, *and* CNOT *gates, which occur in the order:*

$$\mathrm{H} - \mathrm{CNOT} - \mathrm{S} - \mathrm{CNOT} - \mathrm{S} - \mathrm{CNOT} - \mathrm{H} - \mathrm{S} - \mathrm{CNOT} - \mathrm{S} - \mathrm{CNOT}. \qquad \square$$

As noted in Aaronson and Gottesman's article, a direct corollary of this is the upper bound we are seeking when we combine this result with a result from the field of linear reversible circuit synthesis.

Theorem 3 (Patel et al. [128]) *Any n-qubit* CNOT *circuit admits an equivalent* CNOT *circuit with at most $\mathcal{O}(n^2/\log n)$ gates.* □

Since there are a constant number of CNOT layers in Aaronson and Gottesman's canonical form, and each CNOT layer can be synthesized using at most $\mathcal{O}(n^2/\log n)$ CNOT gates, $\mathcal{O}(n^2/\log n)$ serves as an upper bound on the number of entangling gates required to implement an n-qubit Clifford circuit.

Having proved this upper bound, we move on to considering the lower bound. We will first go back to Patel, Markov, and Hayes's article on linear reversible circuit synthesis which provides a hint that the upper bound might be tight. After seeing this, we will prove this to be true

Theorem 4 (Patel et al. [128]) *There are n-qubit* CNOT *circuits which cannot be synthesized using fewer than $\Omega(n^2/\log n)$* CNOT *gates.* □

The proof for this theorem given by Patel, Markov, and Hayes is a counting argument proving that the number of distinct linear reversible circuits is large enough that some linear reversible circuits must require a large number of CNOT gates to be constructed. Moreover, Aaronson and Gottesman even remark on a similar counting argument showing that there exist n-qubit Clifford circuits requiring $\Omega(n^2/\log n)$ gates. Moreover, a minor extension of the above statements directly bounds the number of entangling gates.

Corollary 1 *There are n-qubit Clifford circuits which cannot be synthesized using fewer than $\Omega(n^2/\log n)$ entangling gates*[1]. □

Proof First we will count the number of unique stabilizer states on n qubits. This can be shown to be exactly

$$\prod_{i=0}^{n-1}(2^i + 1),$$

but for our purposes, it suffices to note that there are $2^{\Theta(n^2)}$, as is likewise noted by Aaronson and Gottesman.

On the other hand, we notice that there are a constant number of two-qubit Clifford circuits, since Clifford circuits can be viewed as permutations of Pauli operators. Some of these circuits are non-entangling (and one is the identity operation), which are included nonetheless. On an n-qubit system, therefore, there are $\mathcal{O}(n^2)$ unique two-qubit Clifford entangling gates.

The above implies that there are at most $\mathcal{O}(n^{2T})$ circuits with at most T entangling gates. Comparing this to the number of unique stabilizer states forces us to recognize that, in order for all stabilizer states to be realized by some ($\leq T$)-entangling gate Clifford circuit, T must satisfy:

$$\mathcal{O}(n^{2T}) \geq 2^{\Theta(n^2)},$$

or equivalently, $T = \Omega(n^2/\log n)$. ■

These matching upper and lower bounds tell us that there exist some Clifford circuits for which the canonical form will achieve a near-optimal entangling gate count. Finding the true optimal circuit remains open, and this search motivates our next section. Moreover, having a tight bound now opens the door to investigating families of circuits which do *not* saturate the bound, which will be investigated in Section 5.4.

5.2 Visualizing a True Optimum

For this section and the following one, we will re-focus our attention on the question of improving NISQ algorithms. In particular, the measurement scheme discussed in Chapter 4 required the repeated construction of Clifford circuits mapping a commuting Pauli

[1] During my thesis proposal, David Gosset outlined this proof.

operators to diagonal Pauli operators. A commuting set of Paulis exists in the span of a basis defining a stabilizer state, and diagonal Pauli operators exist in the span of the basis defining the ground state. So, in other words, we are synthesizing circuits which map specified stabilizer states to the ground state, $|0\rangle^{\otimes n}$.

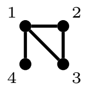
With this goal in mind, we will use the graph state formalism rather than the stabilizer formalism. The graph state formalism provides a visual representation of n-qubit stabilizer states using n-vertex simple graphs.

Definition 11 (Graph State [50]) Given a simple graph, $G = (V, E)$, the corresponding *graph state* is the stabilizer state,

$$|G\rangle = \left(\prod_{(i,\ j)\in E} CZ_{ij} \right) |+\rangle^{\otimes |V|} .$$

□

The set of Paulis which stabilize a given graph state admits a simple basis which can be constructed from the adjacency matrix of the graph. In the symplectic formalism, the X-part of the basis is the identity matrix and the Z-part of the basis is the adjacency matrix of the graph, as shown in the following example.



$$\left|\quad\right\rangle \mapsto \left(\begin{array}{c|c} \mathtt{1000} & \mathtt{0111} \\ \mathtt{0100} & \mathtt{1010} \\ \mathtt{0010} & \mathtt{1100} \\ \mathtt{0001} & \mathtt{1000} \end{array}\right) = \begin{array}{c} \mathtt{XZZZ} \\ \mathtt{ZXZI} \\ \mathtt{ZZXI} \\ \mathtt{ZIIX} \end{array}$$

It is not true that every stabilizer state is a graph state, but fortunately it *is* true that every stabilizer state is local-Clifford equivalent to a graph state. This can be seen by performing a modified Gaussian elimination algorithm on the symplectic form of a basis for the stabilizer state, which is equivalent to composing operators to choose a new basis. In this modified Gaussian elimination, if the i^{th} column is a pivot column in the X-part, then $(n+i)^{\text{th}}$ column cannot be a pivot column in the Z-part. The fact that the generators form a basis guarantees that either i or $n+i$ is a pivot column for all $i = 1, \ldots, n$.

If the X-part is full rank, then the resulting generators will take the form described above, where the fact that the Z-part is symmetric follows from commutation rules. Alternatively, if the X-part is not full rank, then Hadamard gates can map the pivot columns from the Z-part to the X-part until the X-part is indeed full rank.

Using this formalism obscures the total number of single-qubit Clifford operations, but since our goal is only to count entangling gates, the fact that many equivalent stabilizer states are equivalent to the same graph state actually simplifies our task.

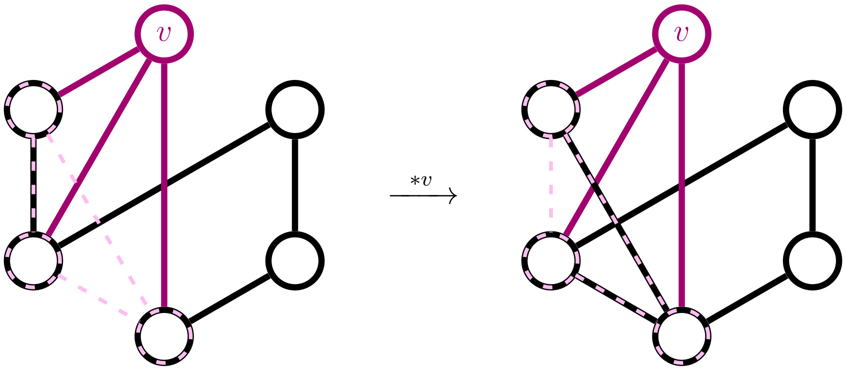


Figure 5.1: A local complement on vertex v. The dark pink edges are those with one endpoint at v. The light pink, dashed edges identify the clique on $N(v)$ on which the edges will be complemented.

To work within the graph state formalism, it is important to understand how Clifford operations act on graph states. In particular, we can actually classify all Clifford operations into three categories: SWAPs, single-qubit gates, and entangling gates. A sequence of SWAP operations is a relabelling of the vertices of the graph state, i.e. a mapping to an isomorphic graph. For the other two classes, we will see that there are simple graph operations which correspond to each.

Definition 12 (Local Complement) Given a graph, G, and a vertex, v, the local complement of G with respect to v (denoted $G * v$) is the graph obtained by complementing the edges of the subgraph of G induced by the neighborhood of v. An example is given in Figure 5.1. □

Definition 13 (Locally Equivalent) We say that two graphs, G and H, are *locally equivalent* (written $G \sim_L H$) if there exists a sequence of local complements mapping G to H. Since local complements are involutory, this can be checked to be an equivalence relation. □

Proposition 1 (Van den Nest et al. [129]) *Two stabilizer states, $|\phi\rangle$ and $|\psi\rangle$, with corresponding graphs, G_ϕ and G_ψ, are equivalent up to single-qubit Clifford gates if and only if $G_\phi \sim_L G_\psi$.* □

The above proposition shows that there is a single graph operation, the local complement, which can be used to generate all local-Clifford equivalent graph states. Moving on to entangling gates, we make use of the following proposition.

Proposition 2 (Grier and Schaeffer [130]) *Any 2-qubit Clifford entangling gate is equivalent up to local-Clifford gates to either a* CZ *gate or an* iSWAP *gate.* □

Proof This is a corollary of the result proved in Appendix E of Grier and Schaeffer's "The Classification of Clifford Gates over Qubits." An iSWAP gate is single-qubit Clifford equivalent to a CZ gate followed by a SWAP gate, and a CZ gate is equivalent to their generalized CNOT gate up to conjugation by single-qubit Cliffords. Therefore, if the SWAP is present, the circuit can be written as a collection of single-qubit Cliffords together with an iSWAP, and if it is not present, then a CZ is used instead. ■

This proposition implies that there are two unique graph operations which can be used to map graph states to one another. On a physical device, one might only have access to a native entangling gate which is single-qubit Clifford equivalent to either a CZ gate or an iSWAP gate, or perhaps both. For simplicity, we focus on the case where CZ gates are native, though a similar analysis would provide similar results for devices with native iSWAP gates (or even both). As a running example, we will look at a Clifford circuit with native CZ gates for generating the five-wheel graph, as shown in Figure 5.2

In their article, "Mapping graph state orbits under local complementation," Adcock et al. analyze local-complementation orbits (that is, sets of graphs which are locally equivalent). In this article, they categorize all local complementation orbits for all graphs on up to eight vertices. Adcock et al. note that the minimum number of edges in the local complementation orbit of a graph represents an upper bound on the number of CZ gates required to build the graph. We can see the improvement from this analysis in Figure 5.3, which constructs the five-wheel graph using one fewer CZ gate than the naive approach.

In the discussion section of their article, Adcock et al. mention a direction for future work in which one could draw a picture of the connections between orbits to help optimize Clifford circuits. They write:

> For a prescriptive method, the relationship between orbits by nonlocal CZ gates must be known. A complete map of this type would describe how all n-qubit graph states are related to one another, and provide a look up table for optimal transformations between them [5].

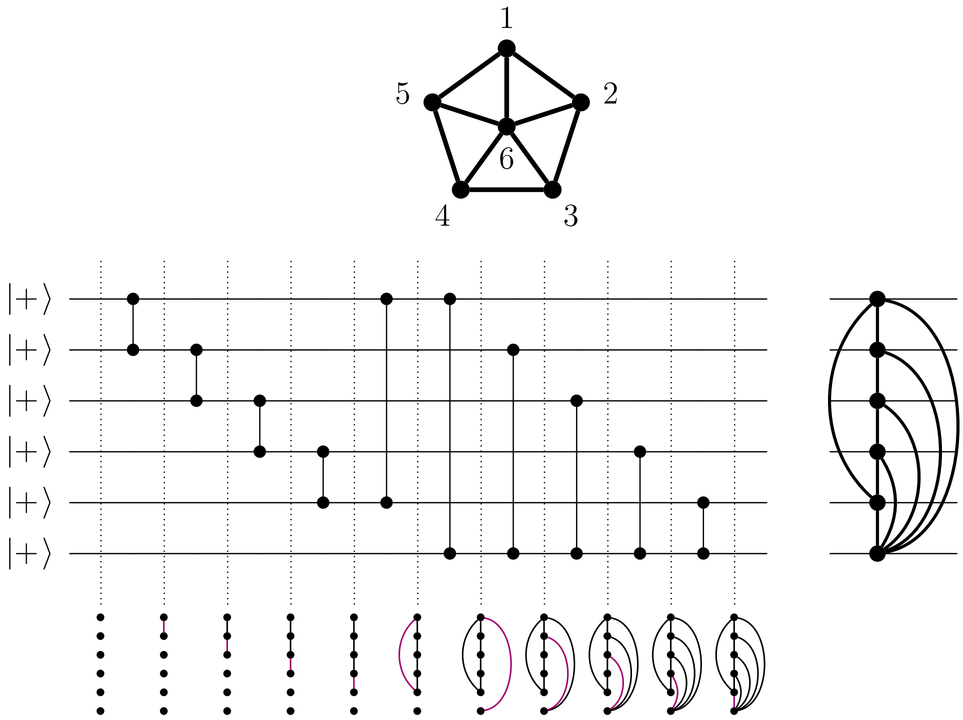


Figure 5.2: A Clifford circuit constructing the five-wheel graph using only CZ gates. All qubits begin in the $|+\rangle$ state and CZ gates are applied to every pair of qubits with an edge in the graph, represented above the circuit. After each layer in the circuit, the current graph state is drawn below.

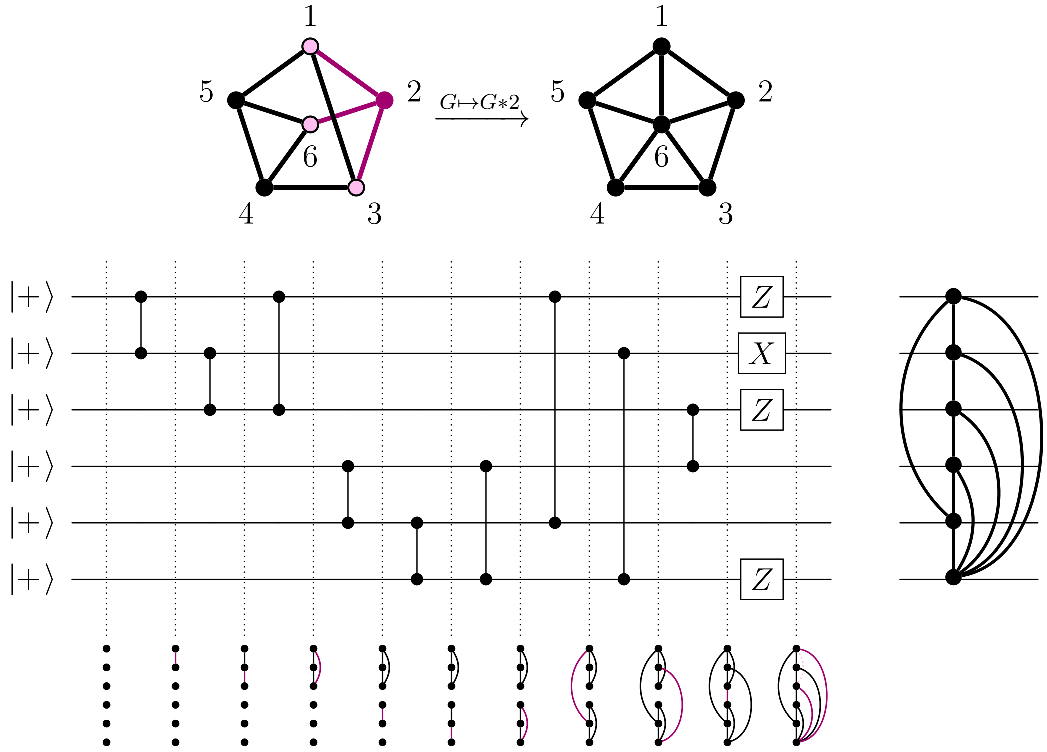


Figure 5.3: A Clifford circuit constructing the five-wheel graph by first constructing the graph in the local-complementation orbit with the fewest edges. The X and Z gates represent the single-qubit operations, $\sqrt{-i\mathtt{X}}$ and $\sqrt{-i\mathtt{Z}}$, which are, up to a constant, equivalent to sequences of the Clifford H and S gates [5].

Here, we will do just that, with the goal not just of creating a lookup table for small numbers of qubits, but also for understanding how to use such a framework to traverse the space of graph states to optimize larger circuits as well. To facilitate our investigation, we will define a distance function corresponding to the minimum number of CZ gates required to map from one graph state to another.

Definition 14 (Entanglement Distance) Given two graphs, G and H, both on n vertices, we define the *entanglement distance* between G and H (denoted $e(G, H)$) to be equal to the minimum k such that there is a sequence of graphs

$$G = G_0, G'_0, G_1, G'_1, \dots, G_k, G'_k = H$$

satisfying the following conditions:

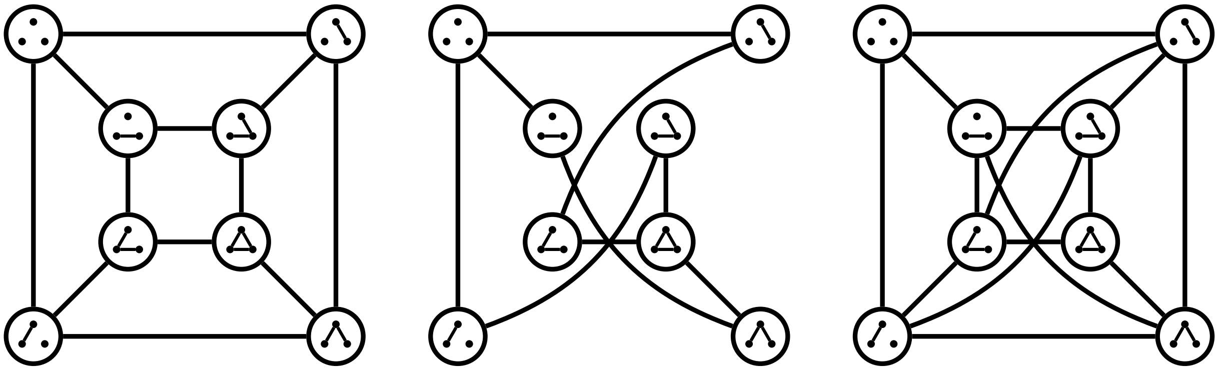
- for each $0 \leq i \leq k$, G_i is locally equivalent to G'_i, and
- for each $1 \leq i \leq k$, G_i is obtained from G'_{i-1} by adding or removing a single edge.

Since mapping to/from the empty graph is particularly motivated by the measurement scheme presented in Chapter 4, we will define $e(G)$ to be equal to the entanglement distance between G and the n-vertex empty graph.

We have called this a "distance," and we can notice that this does indeed satisfy the triangle inequality: for three graphs, F, G, H, we have that $e(F, H) \leq e(F, G) + e(G, H)$. In particular, we also have $e(G, H) \leq e(G) + e(H)$. □

Now, we will introduce a visualization for understanding these distances. We begin with a graph on $2^{\binom{n}{2}}$ vertices, with each vertex labelled by a unique simple n-vertex graphs. For native CZ gates, we know that two graph states can be mapped to one another if and only if they differ at a single edge. Adding an edge between such pairs of graphs results in the $\binom{n}{2}$-dimensional hypercube graph, written as $Q_{\binom{n}{2}}$. While it is not the focus of this section, it's good to note that when considering a device with native iSWAP gates, the result is similar to a hypercube since exactly one edge is added or deleted by each iSWAP gate, but the SWAPs change which graphs are adjacent.[2] Similarly, for a device with native CZ and iSWAP gates, the edges of the regular hypercube and twisted hypercube are both present. A 3-qubit example for each of these cases is shown in Figure 5.4.

[2] In Figure 5.4, instead of iSWAP gates we have drawn the edges for a combined CZ and SWAP gate. This changes the initial graph, but does not change the graph after quotienting out by the local equivalence relation.



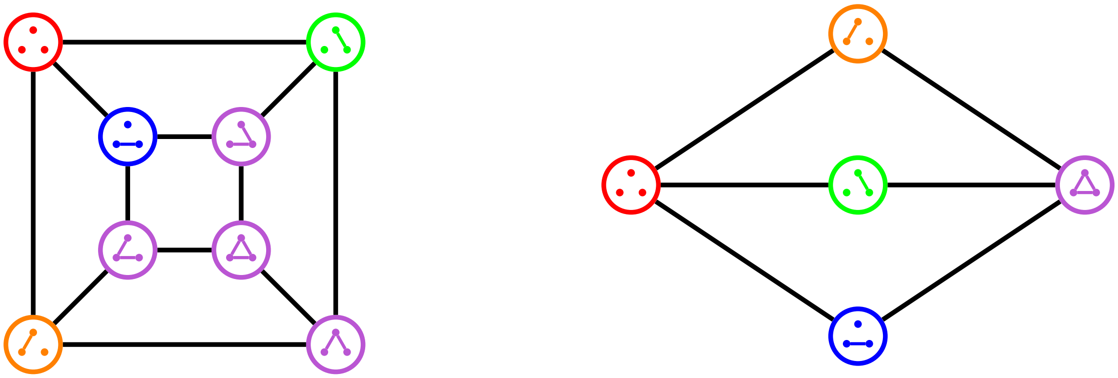
(a) CZ (b) iSWAP (c) CZ and iSWAP

Figure 5.4: Graph state adjacencies for the three native gate sets. CZ gates produce a hypercube graph, here the $\binom{3}{2} = 3$ dimensional one, which makes it simpler for algorithm descriptions. Importantly, the iSWAP and CZ + iSWAP cases are not more complicated to work with, but the graph (before identifying vertices) eschews simple description.

Assuming that the native entangling gates of our hypothetical n-qubit NISQ device are CZ gates, $Q_{\binom{n}{2}}$ captures the mappings between graph states. We will group all of the vertices into classes based on the local equivalence relation, which we now know corresponds to the single-qubit Clifford equivalence of graph states. Having determined this grouping, we compute the quotient graph, $Q_{\binom{n}{2}}/\sim_L$, which reduces each class to a single vertex and preserves edges between classes, as shown in Figure 5.5.

One way to think of this quotient graph is to imagine starting at a given stabilizer state. Every stabilizer state which is equivalent up to single-qubit Clifford circuits is identified to the same vertex in the graph. The native entangling gates are applied to every pair of qubits on every stabilizer state identified by this vertex, and edges are added to the classes of graph states which are reached in this way. If the native entangling gates are all single-qubit Clifford equivalent to the CZ gate, then the edges in the quotient graph capture every possible mapping to a non-local-Clifford equivalent stabilizer state.

The shortest path between two graph states in the quotient graph corresponds to a Clifford circuit mapping equivalent stabilizer states to one another using the minimal number of entangling gates. This quotient graph captures exactly the information which Adcock et al. alluded to. Applying this to our five-wheel example, the minimum number of required CZ gates is 7, which we provide an example of in Figure 5.6.



(a) Grouping into $\sim_L$ classes

(b) $Q_{\binom{3}{2}}/\sim_L$

Figure 5.5: Graph states in $Q_{\binom{3}{2}}$ grouped into local equivalence classes. Each of the distinct five colors represents a class of locally equivalent graphs. When constructing the quotient graph, if there exists an edge between a vertex of one color and a vertex of another color, then the quotient graph contains that edge.

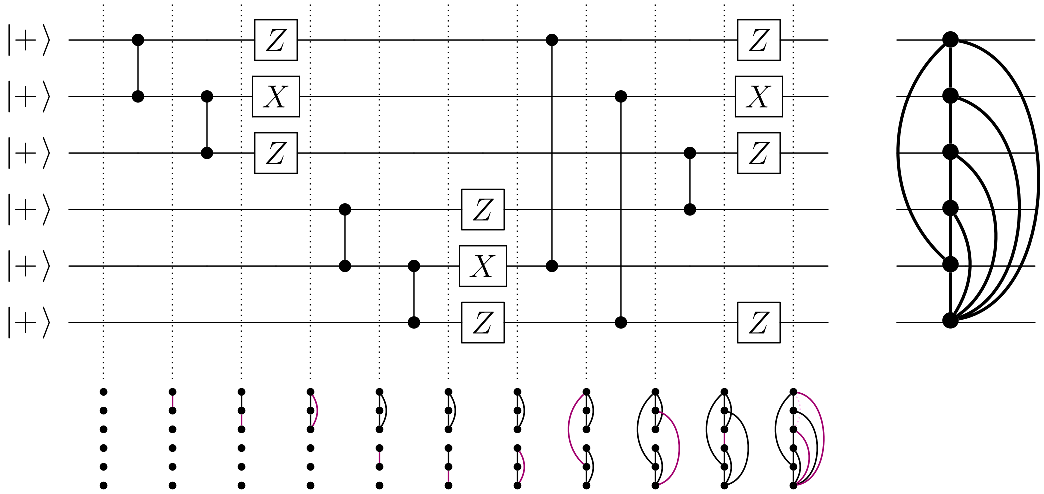


Figure 5.6: A Clifford circuit optimally constructing the five-wheel graph by interspersing local complements and CZ gates. Again, the X and Z gates represent the single-qubit operations, $\sqrt{-i\mathtt{X}}$ and $\sqrt{-i\mathtt{Z}}$, respectively.

5.3 Traversing the Quotient Graph

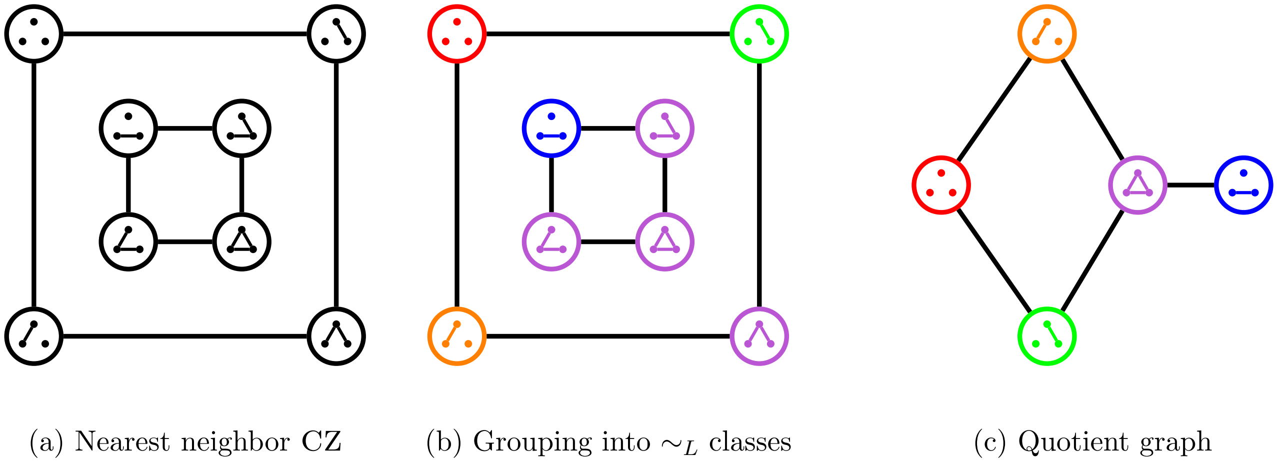
Now that we have a framework for understanding these optimal Clifford circuits, we can begin to ask questions about the structure of the graph and how to make use of it for circuit synthesis.

The first task for the purposes of justifying allowing entangling gates in a measurement protocol, is to determine the eccentricity of the empty graph vertex in the quotient graph. Eccentricity is a measure of the maximum distance between a fixed vertex and any other vertex in a graph, so the eccentricity of the empty graph is the maximum number of entangling gates required to diagonalize an n-qubit stabilizer state, or similarly to prepare an n-qubit graph state from the ground state. In other words, it is the maximum value of $e(G)$ over all n-vertex graphs, G.

Thanks to Theorem 1, we already have an asymptotic answer to this question: $\mathcal{O}(n^2/\log n)$. With the new framework, though, we can exactly calculate the eccentricity for small numbers of qubits. The eccentricities are included in the figures in Appendix B. Of particular note is the highlighted vertex in Figure 2d, which corresponds to the five-wheel graph. This is the smallest graph for which the minimum number of edges in every graph within its local-complementation orbit (9) is strictly greater than the minimum number of CZ gates required to prepare it (7).

Changing direction from a theoretical question to a practical question, one might wonder how to incorporate restricted architectures into this picture. For example, a physically-motivated architecture which is considered for circuit synthesis tasks is a linear nearest neighbor architecture, where entangling gates can only be applied on pairs of qubits which are adjacent in a line. Any architecture can be accounted for using this framework by simply deleting edges from the hypercube which correspond to qubits which cannot directly be entangled. We can similarly group classes of single-qubit equivalent stabilizer states using the same equivalence relation, since all such operations are local, and again compute the quotient graph. An example on 3 qubits is shown in Figure 5.7. For the sake of completeness, we have also found the minimal number of entangling gates on a linear nearest neighbor architecture for construction our running example of the five-wheel graph. Figure 5.8 shows the optimal 9-entangling gate circuit.

Observing this figure, we notice a perhaps unintuitive result. Given the task of entangling qubits 1 and 3 on a nearest neighbor architecture, it's tempting to first swap qubits 1 and 2 before performing a CZ gate on the new qubit 2 and 3. However, since non-native SWAP operations require three entangling gates, this incurs a cost of four entangling gates. Remarkably, we can actually entangle just qubits 1 and 3 using only three entangling gates



(a) Nearest neighbor CZ

(b) Grouping into $\sim_L$ classes

(c) Quotient graph

Figure 5.7: Graph states on a linear nearest neighbor architecture. Here, the bottom-left qubit is 1, the top qubit is 2, and the bottom-right qubit is 3, so a CZ gate can be performed between qubits 1 and 2 as well as qubits 2 and 3, but not between qubits 1 and 3. For any restricted architecture, the graph is the hypercube graph with edges removed along a given dimension.

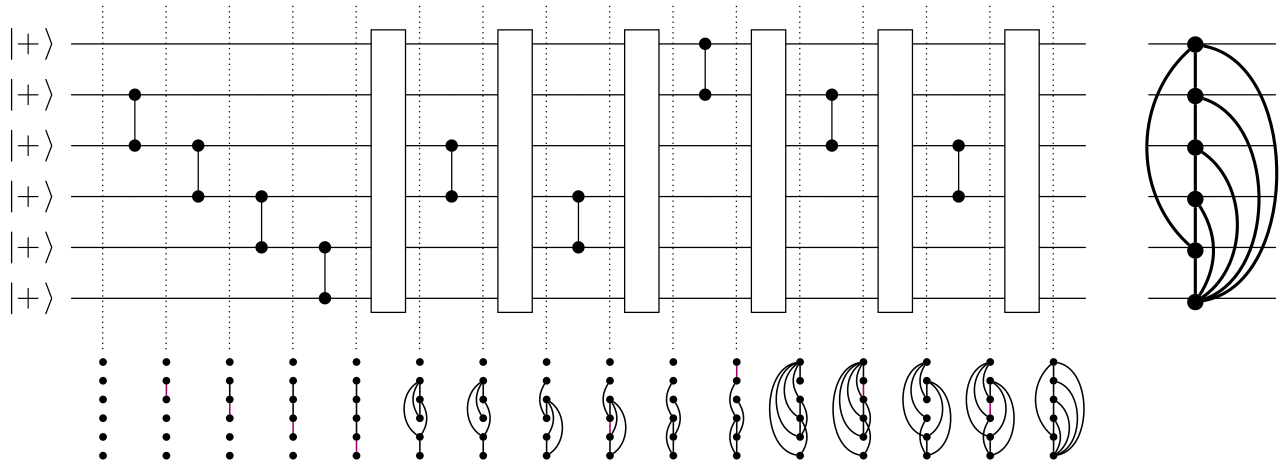


Figure 5.8: A Clifford circuit optimally constructing the five-wheel graph by interspersing local complements and CZ gates on a linear nearest neighbor architecture. The empty gates in the circuit represent sequences of single-qubit X and Z gates. The specific gates were excluded for brevity, but they can be computed by calculating the sequence of local complementations required to map the graph states to one another. Note that, on a restricted architecture, the choice of vertex labels on the graph state impacts the number of required CZ gates, so isomorphic graphs may prove easier or more difficult to construct.

by first creating the maximally entangled gate, applying single-qubit Clifford gates, and then deleting an edge in the graph state with one more CZ.

Restricted architectures result in disjoint hypercubes of lower dimensions, but the quotient operation collapses the layers into a connected graph as long as the architecture is fully connected. Using single-qubit Cliffords to jump between the disjoint hypercubes is a unique and valuable way of understanding Clifford circuit synthesis on restricted architectures. The eccentricities for a linear nearest neighbor architecture are shown in the figures in Appendix C. Again, the five-wheel graph is highlighted in the 6-qubit graph.

There are a multitude of other uses for this framework.

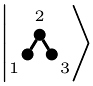
- Weighted edges corresponding to gate fidelities on NISQ devices. By assigning higher weights to the edges corresponding to lower-fidelity entangling gates, the quotient graph can be tailored to a specific device.

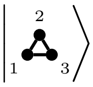
- Adding edges in the hypercube with unique weights corresponding to native SWAP gates. On devices with native SWAP operations, the cost might be more or less than a native entangling gate. After adding weighted edges to the hypercube corresponding to the allowed SWAPs on a device (so edges connecting certain isomorphic graphs), the shortest paths in the quotient graph will use the native SWAP gates if they would result in a more cost-efficient circuit.

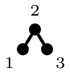
- Expanding the equivalence relation to include graph isomorphisms. For the measurement scheme of variational algorithms or for the preparation of graph states in MBQC, since the initial or final state is the empty graph, permutations of the qubits can be pulled to the beginning or end of the circuit and dealt with classically. Considering isomorphic graphs as being equivalent results in a significantly smaller quotient graph, which is perhaps easier to traverse.

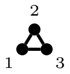
The above examples all represent promising avenues for applications of this framework to real-world Clifford circuit synthesis problems, but none of them are realistic unless we can say something about the efficiency of finding paths in this graph. Traversing the graph, that is visiting every vertex, is clearly inefficient, as the number of vertices grows exponentially in the number of qubits. However, perhaps we can construct paths efficiently without ever building the graph in the first place.

The first question is whether we can efficiently recognize when we have already visited a given vertex. For example,

CZ_{13} | 

⟩ = | 

⟩ and 

$\sim_L$ 

,

so applying a CZ gates (or even multiple CZ gates) does not necessarily map to a distinct class of graph states. For this reason, any algorithm which hopes to efficiently traverse the quotient graph must be able to compute equivalence efficiently. Fortunately, this is indeed possible, and it was proved both in the language of graphs as well as graph states.

Proposition 3 (Bouchet, Van den Nest et al. [51, 131]) *There exists an $\mathcal{O}(n^4)$ classical algorithm to determine whether two graphs are locally equivalent. This implies an efficient classical algorithm for determining whether two graph states are single-qubit Clifford equivalent.* □

Applying this to the problem of traversing the graph, we can efficiently recognize whenever applying an entangling gate has mapped us to a distinct class of graph states. However, even this is not enough to guarantee efficient traversal of the graph if calculating the neighborhood of a vertex in the quotient graph is inefficient.

At first glance, the fact that there exist local-complementation orbits with an exponential number of elements[3] seems to complicate this task. It is inefficient to enumerate all graphs in the local equivalence class and add/delete all $\binom{n}{2}$ edges to each graph to visit all its neighbors.

This perspective is a red herring, though, since we are not obliged to stay within the graph state formalism. As noted in the proof of Theorem 1, there are a constant number of 2-qubit Clifford circuits. Instead of adding/deleting edges to the graph, we can apply each of the 2-qubit Clifford circuits to every pair of qubits in the stabilizer state before mapping the results back to graph states to check for equivalence.

Using the above, we can traverse all vertices of the quotient graph for n-qubit graph states in time at most $\mathcal{O}(n^4|V|^2)$. It has already been mentioned, though, that $|V|$ is exponential in n, so it would be better to find paths directly without first constructing the graph. For this, we will employ the A* algorithm.

Definition 15 (A* Algorithm [132]) The *A* algorithm* is a path searching algorithm defined as follows. Given a graph, G, a source vertex, v_S and a target vertex, v_T, A* seeks to output the shortest path from v_S to v_T. To improve upon Dijkstra's algorithm, A* uses a heuristic function, which estimates the distance between each vertex to the target. At each step, A* chooses the vertex in the neighborhood of the set of already-visited vertices which minimizes the sum of the path length from the source together with the heuristic function's estimate for the remaining path length to the target. □

[3] Take, for example, the disjoint union of triangles on $3n$ vertices. Each triangle has 4 elements in its local-complementation orbit, so the orbit of the disjoint union of n triangles contains 4^n elements.

From this description of the A* algorithm, it's not clear whether it produces a good or even optimal result. Fortunately, there are conditions on the heuristic function which can guarantee the efficiency and optimality of the algorithm [133].

Beginning with optimality, Dijkstra's algorithm, a similar path searching algorithm which does not use a heuristic function, is guaranteed to be optimal. Since Dijkstra's algorithm is essentially A* with a heuristic function which is constant 0, it's clearly possible to guarantee that the output of A* is optimal. The condition which guarantees optimality is called admissibility, which means that the heuristic function always underestimates (or correctly estimates) the distance remaining to the target. The proof of optimality is, if A* returned a path of longer length, then the heuristic function on some vertex on the true shortest path must have overestimated the distance remaining, since otherwise there would've been a step at which it was chosen over a vertex on the longer path. This contradicts admissibility, so the returned path must have the shortest length.

Of course, heuristic functions which severely underestimate the distance remaining have a drawback, which is that they force the algorithm to explore more of the graph. Dijkstra's algorithm results in a breadth-first search for the target vertex, and given that there are an exponential number of vertices in our quotient graph, this is undesirable.

For the purpose of mapping to or from the empty graph, the problem which we keep coming back to, it is easy to imagine non-trivial heuristic functions which are admissible. The first which comes to mind is that a given graph, G, is at most $n - c(G)$ steps away from the empty graph, where $c(G)$ is the number of connected components of the graph state. Each CZ gate can split one component into at most two, so this is clearly admissible. However, beyond n steps from the empty graph, this heuristic function is constant, and the resulting algorithm will explore a large proportion of the graph, especially considering that some graph states are $\mathcal{O}(n^2/\log n)$ steps away.

An ideal heuristic function which returns the true distance to the empty graph, would immediately lead to an efficient classical algorithm for optimally minimizing entangling gates for diagonalizing stabilizer states and/or constructing graph states. While we search for this (or until we prove that the computation of such a function is NP-hard), it is likely best to instead use inadmissible heuristic functions.

A heuristic function which overestimates the distance to the empty graph might not result in the shortest path, but it will guarantee that the algorithm terminates quickly if desired. For example, using the number of edges in the graph as a heuristic function, we can clearly always delete an edge and move one step closer to the empty graph, guaranteeing an $\mathcal{O}(n^2)$ worst-case runtime. By exploring the neighborhoods of the visited vertices, though, we can ideally remove multiple edges in each step following the A* formula, and we are likely to achieve results significantly better than the naive approach.

We are considering a few different directions for investigating new and improved heuristic functions, but as of now, the problem of using A* to find truly optimal Clifford circuits is tantalizingly open.

5.4 An Efficient Class of Graphs

In the last section, we introduced ideas for an algorithm for finding the shortest path between two vertices in the quotient graph. In this section, we will be investigating the length of this shortest path, so we will call this the "entanglement distance" between G and H, which is defined formally as follows.

Definition 16 (Induced Subgraph) Given a graph, G, and a subset of its vertices, $S \subseteq V(G)$, the *induced subgraph* of G on S (denoted $G[S]$) is the graph with vertex set S and edge set $\{(u, v) \in E(G) : u \in S \wedge v \in S\}$. □

Definition 17 (Vertex-Minor) A graph, H, is said to be a *vertex-minor* of a graph, G, if it is an induced subgraph of some graph locally equivalent to G. □

Circle graphs are the most fundamental vertex-minor-closed class of graphs and are believed to play a role for vertex-minors that is analogous to that of planar graphs for graph minors (see [134, 135]). One striking result is an analogue of Kuratowski's theorem chacterising circle graphs by a list of three forbidden vertex-minors [136]. Because of this, circle graphs have been an active area of research for investigating properties of local complements and vertex-minors.

In this section, we will prove the following theorems, which show that the entanglement distance of circle graphs (and indeed a larger family of vertex-minor-closed classes of graphs) is bounded above by $\mathcal{O}(n \log n)$. Therefore, in contrast to the graphs with entanglement distance $\Theta(n^2/\log n)$, these classes of graphs require asymptotically fewer entangling gates, which marks them as especially interesting classes to study for applications to NISQ algorithms.

Theorem 5 *Let G be an n-vertex circle graph. Then $e(G) \leq 2n \log n + 3n$.* □

Theorem 6 *Assuming Geelen's weak vertex-minor structure conjecture, we have the following. Let $\mathcal{F}$ be a proper vertex-minor-closed class of graphs, and let G be an n-vertex graph contained in $\mathcal{F}$. Then $e(G) = \mathcal{O}(n \log n)$.* □

Although the weak vertex-minor structure theorem has not yet been proved, there has been significant progress towards obtaining this and the general structure theorem for vertex-minors (see [134, 135, 137]), and we are optimistic that it will be proved in the coming years. For a recent survey on vertex-minors, see [138].

5.4.1 Preliminaries

We will often use the following key observation which can easily be verified.

Lemma 1 *Let u, v be distinct vertices of a graph G, and let $H = (((G * v)\Delta\{uv\}) * v)\Delta\{uv\}$, then H is the graph obtained from G by removing all edges between u and $N(u) \cap N(v)$ and adding all edges between u and $N(v) \backslash (N(u) \cup \{u\})$. As a consequence, $e(G, H) \leq 2$. An example of this operation is given in Figure 5.9* □

5.4.2 Circle graphs

For a graph, G, and two disjoint vertex sets, $A, B \subseteq V(G)$, we let $G[A, B]$ be the bipartite subgraph of G on vertex set $A \cup B$ where xy is an edge of $G[A, B]$ if and only if xy is an edge of G, and $x \in A$, $y \in B$.

Definition 18 (Circle Graph) Given a chord diagram, which is a finite set of chords of a circle, the corresponding *circle graph*, G, is the graph with vertex set and edge set:

$$\begin{aligned} V(G) &= \{c : c \text{ is a chord}\} \\ E(G) &= \{(c_1, c_2) : c_1 \text{ intersects } c_2\} \end{aligned}$$

□

Two intervals I_1, I_2 in $\mathbb{R}$ "overlap" if they intersect and neither is contained in the other. For a collection of closed intervals $\mathcal{I}$ in $\mathbb{R}$, the *overlap graph*, $G(\mathcal{I})$, is the graph with vertex set $\mathcal{I}$ and edge set being the pairs of overlapping intervals in $\mathcal{I}$. It has been shown that every circle graph is an overlap graph of a collection of intervals in $\mathbb{R}$ such that no two share an endpoint [139].

Lemma 2 *Let G be a n-vertex circle graph with an isolated vertex u, let $A, B \subseteq V(G) \backslash \{u\}$ be disjoint, and let F be the edges between A and B in G. Let H be a graph on the same vertex set as G, with u an isolated vertex. Then $e(H, H\Delta F) \leq 2n - 2$.* □

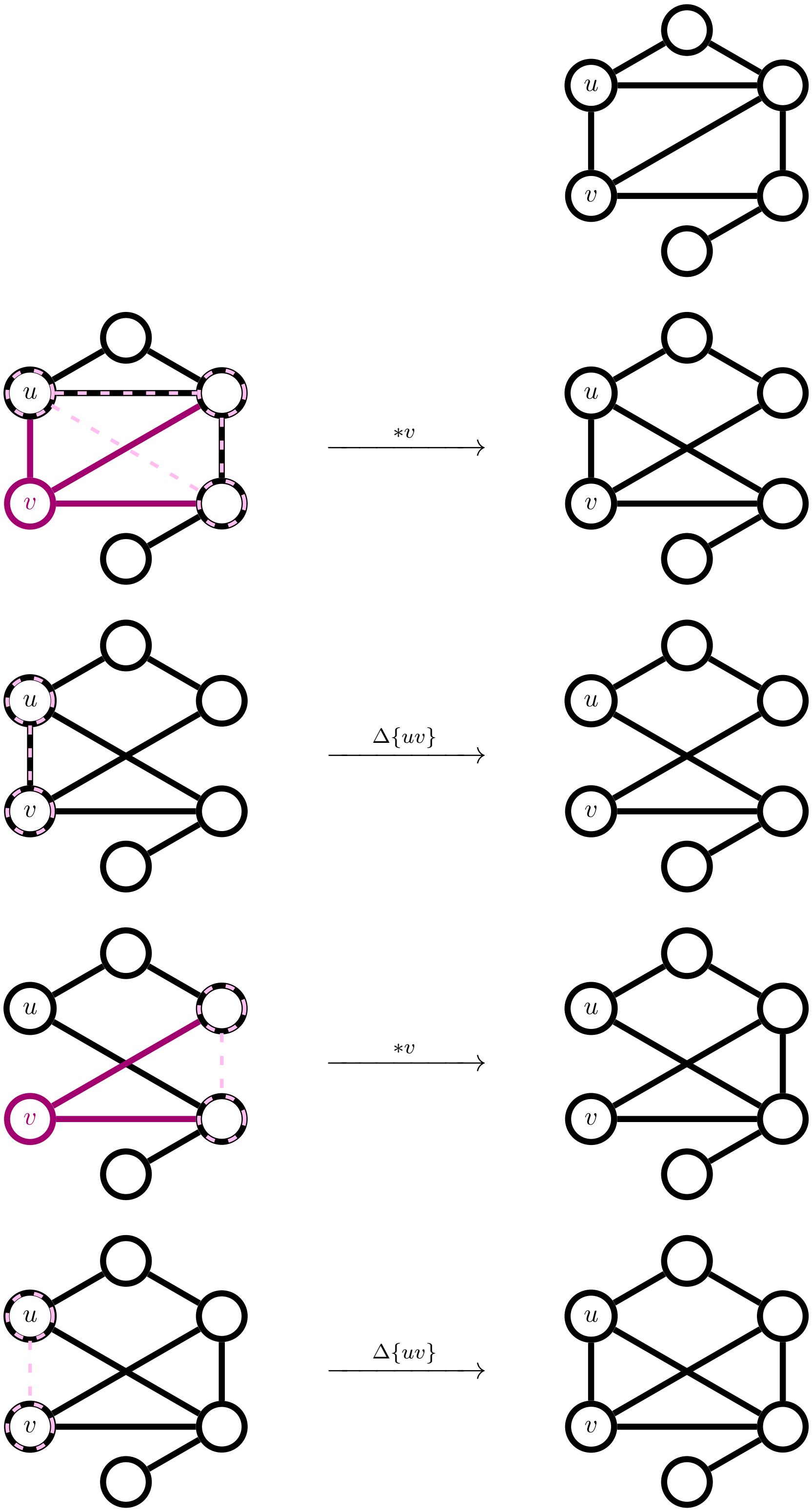


Figure 5.9: An example of the operation described in Lemma 1. $N(u) \mapsto (N(u)\Delta N(v))\setminus u$.

Proof Let $\mathcal{I}$ be a collection of closed intervals in $\mathbb{R}$ with no two sharing an endpoint such that $G\backslash\{u\} = G(\mathcal{I})$. Let $x_1 < \cdots < x_{2n-2}$ be the endpoints of $\mathcal{I}$, and for each x_i, let $I(x_i)$ be the interval of $\mathcal{I}$ that has x_i as an endpoint. For $I \in \mathcal{I}$, let $\ell(I)$ be such that $x_{\ell(I)}$ is the left endpoint of I, and let $r(I)$ be such that $x_{r(I)}$ is the right endpoint of I.

Let $H_0 = H$. Now, for each $1 \leq i \leq 2n-2$ in order,

- if $I(x_i) \in A$, then let $H_i = ((H_{i-1} * u)\Delta\{uI(x_i)\}) * u$,
- if $I(x_i) \in B$, then let $H_i = H_{i-1}\Delta\{uI(x_i)\}$, and
- otherwise, let $H_i = H_{i-1}$.

Clearly for each i, we have that $e(H_i, H_{i-1}) \leq 1$, so $e(H_{2n-2}, H) \leq 2n-2$. It remains to show that $H_{2n-2} = H\Delta F$. Observe that for each $0 \leq i \leq 2n-2$, we have that $N_{H_i}(u) = \{I \in A \cup B : x_i \in I\}$. So, $N_{H_{2n-2}}(u) = N_{H\Delta F}(u)$.

By Lemma 1, we have that if $I(x_i) \in A$, then $H_i\backslash\{u\} = (H_{i-1}\Delta\{wI(x_i) : w \in N_{H_{i-1}}(u)\})\backslash\{u\}$. Clearly if $I(x_i) \in B$, then $H_i\backslash\{u\} = H_{i-1}\backslash\{u\}$, and otherwise if $I(x_i) \notin A \cup B$, then $H_i = H_{i-1}$. It now follows that

$$\begin{aligned} H_{2n-2} &= H\Delta_{I\in A}\Big(\{Iw : w \in N_{H_{\ell(I)-1}}(u)\}\Delta\{Iw : w \in N_{H_{r(I)-1}}(u)\}\Big) \\ &= H\Delta_{I\in A}\big(\{IJ : x_{\ell(I)}J \in A\cup B\}\Delta\{IJ : x_{r(I)}J \in A \cup B\}\big) \\ &= H\Delta_{I\in A}\{IJ : J \in A \cup B \wedge IJ \in G(\mathcal{I})\} \\ &= H\Delta_{I\in A}\{IJ : J \in B \wedge IJ \in G(\mathcal{I})\} \\ &= H\Delta F, \end{aligned}$$

as desired. ■

With a divide and conquer strategy, we can extend this further to circle graphs with bounded chromatic number. A graph is *k-colorable* if there is an assignment of at most k colors to its vertices so that no two adjacent vertices are assigned the same color. The *chromatic number*, $\chi(G)$, of a graph, G, is equal to the minimum k such that G is k-colorable.

Lemma 3 *Let G be an n-vertex circle graph with chromatic number at most k and an isolated vertex u. Then $e(G) \leq (2n-2)\lceil \log k \rceil$.* □

Proof If $k = 1$, then the result is trivial, so we proceed inductively. Let $A, B \subseteq V(G)\backslash\{u\}$ be a partition such that $\chi(G[A]) = \lfloor \frac{k}{2} \rfloor$ and $\chi(G[B]) = \lceil \frac{k}{2} \rceil$.

By the inductive hypothesis, we have that

$$\begin{aligned} e(G[A] \cup G[B] \cup G[\{u\}]) &\leq e(G[A \cup \{u\}]) + e(G[B \cup \{u\}]) \\ &\leq 2|A| \left\lceil \log \left\lfloor \frac{k}{2} \right\rfloor \right\rceil + 2|B| \left\lceil \log \left\lceil \frac{k}{2} \right\rceil \right\rceil \\ &= (2n-2)(\lceil \log k \rceil - 1). \end{aligned}$$

By Lemma 2, we have that $e(G, G[A] \cup G[B] \cup G[\{u\}]) \leq 2n-2$. Therefore, $e(G) \leq e(G[A] \cup G[B] \cup G[\{u\}]) + e(G, G[A] \cup G[B] \cup G[\{u\}]) \leq (2n-2) \lceil \log k \rceil$, as desired. ■

Corollary 2 *Let G be an n-vertex circle graph with chromatic number at most k. Then $e(G) \leq (2n-2) \lceil \log k \rceil + n - 1$.* □

Proof Let u be a vertex of G. By Lemma 3, we have that $e(G\backslash\delta(u)) \leq (2n-2) \lceil \log k \rceil$. Clearly $e(G\backslash\delta(u), G) \leq |\delta(u)| \leq n-1$. Therefore, $e(G) \leq e(G\backslash\delta(u)) + e(G\backslash\delta(u), G) \leq (2n-2) \lceil \log k \rceil + n - 1$, as desired. ■

Since the chromatic number of an n-vertex graph is at most n, Theorem 5 follows from from Corollary 2.

5.4.3 Perturbations

A *rank-p perturbation* of a graph, G, is a graph whose adjacency matrix can be obtained from the adjancency matrix of G by adding (over the binary field) a symmetric matrix of rank at most p and changing all diagonal entries to be 0. For a graph, G, and a subset of its vertices, $X \subseteq V(G)$, we say that *complementing on X* is the act of obtaining a new graph H from G by replacing the induced subgraph of G on X by its complement.

Lemma 4 (Nguyen and Oum, McCarty [140, 135]) *Let G be an n-vertex graph and let H be a rank-p perturbation of G. Then G can be obtained from H by complementing on at most $\frac{3}{2}p$ sets of vertices.* □

Let us first examine the entanglement distance between one graph and the graph obtained by complementing on a set of vertices.

Lemma 5 *Let G be an n-vertex graph, and let H be obtained by complementing on a set $X \subseteq V(G)$. Then $e(G, H) \leq 2n - 2$.* □

PROOF Let u be a vertex of G, and let G_1 be the graph obtained from G by changing the neighborhood of u to be $X \backslash \{u\}$. Let $G_2 = G_1 * u$. Then H is obtained from G_2 by simply changing the neighborhood of u. Therefore $e(G, H) \leq e(G, G_1) + e(G_1, G_2) + e(G_2, H) \leq (n-1) + 0 + (n-1) = 2n - 2$, as desired. ■

Applying Lemma 4 and repeatedly applying Lemma 5 we now obtain the following.

Theorem 7 *Let G be an n-vertex graph and let H be a rank-p perturbation of G. Then $e(G, H) \leq 3pn - 3p$.* □

An immediate corollary of Theorem 5 and Theorem 7 is the following.

Corollary 3 *Let G be an n-vertex rank p-perturbation of a circle graph. Then $e(G) \leq 2n \log n + (3p + 3)n - 3p$.* □

5.4.4 Vertex-minors

The *cut-rank* of a set $X \subseteq V(G)$, denoted $\rho(X)$, is the rank of the submatrix of the adjacency matrix with rows X and columns $V(G) - X$. For $k \in \mathbb{N}$, a graph G is *k-rank-connected* if it has at least $2k$ vertices and $\rho(X) \geq \min(|X|, |V(G) - X|, k)$ for each $X \subseteq V(G)$.

Geelen's weak vertex-minor structure conjecture states that every graph in a proper vertex-minor-closed class of graphs with sufficiently high rank-connectivity is a bounded rank perturbation of a circle graph.

Conjecture 1 (Weak Structural Conjecture [135]) *For any proper vertex-minor-closed class of graphs $\mathcal{F}$, there exist $k, p \in \mathbb{N}$ so that each k-rank-connected graph in $\mathcal{F}$ is a rank-p perturbation of a circle graph.*

There is also a stronger vertex-minor structure conjecture,[135] however we shall not need this for our purposes. Assuming Conjecture 1 and by using Corollary 3, we can now derive a $\mathcal{O}(n \log n)$ bound for graphs forbidding a vertex-minor.

Theorem 8 *Let $\mathcal{F}$ be a proper vertex-minor-closed class of graphs, and let G be an n-vertex graph contained in $\mathcal{F}$. Then $e(G) = \mathcal{O}(n \log n)$.* □

Proof Let $k, p \in \mathbb{N}$ be as in Conjecture 1. We shall argue inductively on n that $e(G) \leq 12k^2 pn \log n$. We do not optimize the dependence on k and p, since they will likely be large.

If G is a rank-p perturbation of a circle graph, then by Corollary 3,

$$\begin{aligned} e(G) &\leq 2n \log n + (3p+3)n - 3p \\ &\leq 12pn \log n \\ &\leq 12k^2 pn \log n. \end{aligned}$$

So we may assume that G is not a rank-p perturbation of a circle graph. Therefore, G is not k-rank-connected.

If $n \leq 2k$, then clearly $e(G) \leq |E(G)| \leq \binom{n}{2} \leq \binom{2k}{2} \leq 2k^2 \leq 12k^2 pn \log n$, so we may assume that $n > 2k$. So, there exists some $X \subset V(G)$ with $\rho(X) < \min(|X|, |V(G) - X|, k)$. We may choose such a X so that $|X| \geq |V(G) \backslash X|$. Let $a_1, \ldots, a_{\rho_G(X)}$ be vertices of $V(G) \backslash X$ such that $\rho_G(X) = \rho_G(V(G) \backslash X) = \rho_{G[X \cup A]}(A)$, where $A = \{a_1, \ldots, a_{\rho_G(X)}\}$. Let $Y = V(G) \backslash (X \cup A)$. Then by the inductive hypothesis, $e(G[X \cup A]) \leq 12k^2 p|X \cup A| \log |X \cup A|$, and $e(G[Y]) \leq 12k^2 p|Y| \log |Y|$. Since $|Y| \leq n/2$, we therefore get that

$$\begin{aligned} e(G[X \cup A] \cup G[Y]) &\leq e(G[X \cup A]) + e(G[Y]) \\ &\leq 12k^2 p|X \cup A| \log |X \cup A| + 12k^2 p|Y| \log |Y| \\ &\leq 12k^2 p|X \cup A| \log n + 12k^2 p|Y| \log \frac{n}{2} \\ &= 12k^2 p|X \cup A| \log n + 12k^2 p|Y|(\log n - 1) \\ &= 12k^2 pn \log n - 12k^2 p|Y|. \end{aligned}$$

Let G_0 be the graph obtained from $G[X \cup A] \cup G[Y]$ by removing all edges between the vertices of A. Then, $e(G_0) \leq e(G[X \cup A] \cup G[Y]) + \frac{1}{2}k(k-1) \leq 12k^2 pn \log n - 12k^2 p|Y| + \frac{1}{2}k^2$ since $|A| = \rho_G(X) \leq k$.

Let G_1 be the graph obtained from G_0 by adding edges between X and Y so that $E_{G_1}(X, Y) = E_G(X, Y)$. For each vertex $y \in Y$, there exists some $A_y \subseteq A$ such that $N_G(b) \cap X = \Delta_{a \in A_y} N_{G_0}(a)$. Note that $|A_y| \leq |A| \leq k$ for each $y \in Y$. So, by repeatedly applying Lemma 1 a total of $|A_y|$ times for each vertex $y \in Y$, we have that

$$e(G_0, G_1) \leq \sum_{y \in Y} 2|A_y| \leq 2k|Y|.$$

Therefore $e(G_1) \leq 12k^2 pn \log n - 12k^2 p|Y| + \frac{1}{2}k^2 + 2k|Y| \leq 12k^2 pn \log n - \frac{1}{2}k^2$.

Now, G_1 and G differ only on the edges between vertices of A. So, $e(G_1, G) \leq \frac{1}{2}k(k-1) \leq \frac{1}{2}k^2$ since $|A| = \rho_G(X) \leq k$. Therefore

$$e(G) \leq e(G_1) + e(G_1, G) \leq 12k^2 pn \log n$$

as desired. ■

5.4.5 Open Problems

We have obtained $\mathcal{O}(n \log n)$ bounds for the entanglement distance of circle graphs and more generally (assuming Geelen's weak vertex-minor structure theorem) for any proper vertex-minor-closed class of graphs. This is a near linear improvement on the best possible bound of $\Theta(n^2/\log n)$ for the class of all graphs. On the other hand, we have the lower bound of $e(G) \geq n-1$ for any n-vertex connected graph G. Therefore, a logarithmic gap still remains, leading to the following problem.

Problem *Let $\mathcal{F}$ be a proper vertex-minor-closed class of graphs and let G be an n-vertex graph contained in $\mathcal{F}$. Is it true that $e(G) = \mathcal{O}(n)$?*

In response to this problem, we have the following conjecture.

Conjecture 2 *There are n vertex circle graphs G with $e(G) = \Omega(n \log n)$.*

Unfortunately, we have been unsuccessful modifying counting arguments, similar to those used in Theorem 4 and Theorem 1, to work in the case of circle graphs, so this conjecture remains open.

Chapter 6

Applications and Future Directions

Applications of the VQE were mentioned briefly in Chapter 4 during the discussion of the AEQuO measurement scheme, but it's important to note that physicists are, at this moment, using these and similar techniques on NISQ devices. In particular, my first proposed measurement scheme which derived from work in my masters thesis [31], was used to accurately estimate the ground state energy of an H_2 molecular Hamiltonian using Rigetti's Aspen-4-4Q-A quantum processing unit [32]. Similarly, the same measurement techniques were proposed for a VQE-based protocol for the study of magnetic field effects [21]. Later, we adapted the AEQuO measurement protocol and our simulation code to work over prime-dimensional finite fields, which contributed to a VQE simulation on a trapped-ion qudit quantum processor [141].

Finally, the combinatorial and graph-theoretic approaches described herewithin have a plethora of applications outside of those listed. In particular, to efficiently prepare the qubit coupled cluster Ansatz with involutory linear combinations of anticommuting Paulis, one needs to construct a large anticommuting set of Paulis with problem-specific constraints, which was solved using a similar graph partitioning approach [142]. Moreover, expanding outside of the NISQ regime, one recent approach to the optimization of Clifford+$\mathtt{T}$ circuits proposed by Litinski considers $\mathtt{T}$ gates (as well as arbitrary rotations in the $\mathtt{Z}$ basis), conjugated by Clifford circuits, as rotations in a specified Pauli basis [143]. The commutation relations between these Pauli bases determine the order in which the $\mathtt{T}$ gates can be implemented, and commuting sets can be implemented in parallel (given access to ancilla qubits).

Having gone over some past and present applications of our research, we can now discuss future directions. In Chapter 3, we investigated techniques for reducing the number of qubits required to represent local Hamiltonians. The ability to parallelize the computation of the ground state for any subset of the conditional qubits provides the ability to weigh a space-time tradeoff. Being able to avoid the exponentially costly number of repetitions, however, would make this technique significantly more valuable for real-world applications, especially for larger Hamiltonians.

Doing so requires identifying which sub-matrix in a block-diagonal matrix contains the lowest minimum eigenvalue, which is a difficult problem in general. However, for the classes of Hamiltonians which are of interest to researchers, there may be insights which make such a problem feasible. In particular, a deeper understanding of the symmetries of the system which allow for the mapping to a block-diagonal form could give insight into which block(s) could contain the ground state.

One key idea which underlies the results in Chapter 3 is the fact that commuting, diagonalizable matrices are simultaneously diagonalizable, so their eigenspaces coincide. Since non-identity Paulis have predictable eigenspaces in which the ±1 eigenvalues appear with multiplicity 2^{n-1}, finding a Pauli operator which commutes with the Hamiltonian allows us to search for a ground state separately in each of these two eigenspaces. From this point of view, while Pauli operators are the most obvious candidates to test for symmetries, they need not be the only ones. In particular, we can ask what would happen if we expanded our search to look for any other unitary symmetries with efficiently computable eigenspaces.

Even if we settle on a set of unitaries with that property, we still have to be able to test whether the Hamiltonian commutes with a unitary from this set. The Pauli operators commute if and only if they commute with each Pauli term in the local Hamiltonian, but general unitaries do not satisfy this condition. Therefore, searching for another class of symmetries greatly increases the difficulty of the problem.

We are having ongoing discussions about testing for Clifford symmetries for restricted classes of Clifford with easily-computable eigenspaces. Even if successful, proofs of optimality will not come easily with such an approach. Moreover, since it has been shown that determining the joint eigenvalues of a collection of low-weight tensor products of S gates is NP-complete, the sets of Clifford symmetries which allow for efficiently tapering qubits may prove to be a very restricted class [144].

In Chapter 4, we introduced AEQuO, an algorithm for adaptively choosing measurement bases in the VQE based on the collected measurement results. The error formula, when the covariance matrix is known, provides a way to compare the results of measurement schemes between different research groups without requiring hundreds or thousands of runs. Since this is only feasible for small Hamiltonians, the estimated error formula (which uses Bayesian estimates of the error) converged to the correct error using fewer repetitions than a statistical analysis on the outcomes would have provided.

However, we acknowledge that there are many areas where our algorithm falls short of optimality, which opens plenty of room for future work. In particular, the task of estimating a covariance matrix from measurement outcomes on subsets of the Paulis in the Hamiltonian results presents a difficult dilemma. This problem is one of missing data, and the fact that anticommuting Paulis cannot be measured simultaneously leads to likely inevitable biases in the estimation of the true covariance matrix. Our Bayesian approach converged on the expected result efficiently for the Hamiltonians we tested, but a proper statistical analysis of the problem at hand could lead to a provably unbiased estimator.

Continuing to look at AEQuO, we have also used a simple greedy approach to allocating measurements and a machine learning approach. While we were showed that these approaches, when combined with the covariance estimates, produced better results than the available literature on sample Hamiltonians, there is plenty of room for improvement. In particular, an approach which looks forward to future measurements to intelligently assign measurements in the next step would undoubtedly perform better than the greedy approach, as evidenced by the machine learning algorithm accompanying AEQuO. AEQuO struck a balance between efficiency and accuracy, but there are certainly a plethora of tradeoffs for future researchers to investigate.

In Chapter 5, we developed a framework for finding (near-)optimal Clifford circuits with as few entangling gates as possible. The biggest open question from this section, which was even pointed out within the text of the chapter, is the question of determining a good heuristic function for the A* algorithm to traverse the quotient graph.

While investigating the quotient graphs on small numbers of qubits, a pattern emerged in which well-known quantum error correcting codes, such as the perfect 5-qubit code [145], appear as subsets of bases of stabilizers which are further from the empty graph. The generators of the 5-qubit code combined with either of its logical operators forms a basis for the unique stabilizer state at a distance of 5 away from the empty graph in Figure 2c. Such a realization is perhaps not surprising, since quantum error correcting codes must, by definition, be highly entangled and resistant to being disentangled (i.e. mapped to the empty graph). This pattern has pushed us in the direction of investigating the connections between the distance from the empty graph and the distance of some underlying quantum error correcting code defined by the stabilizer state, and there are certainly other connections to be made and directions to be pushed while searching for this elusive heuristic function.

One avenue which was not explored in Chapter 5 comes from a reading Bouchet's paper in which he proved the efficient algorithm for recognizing locally equivalent graphs [51].

In this work, Bouchet, a matroid theorist, proves the result using a notion of isotropic systems, which he originally introduced a few year prior [6]. Decades later, Traldi defined an isotropic matroid, which is a binary matroid constructed from the adjacency matrix of a graph, and proved the following.

Proposition 4 ([146]) *Let G_1 and G_2 be simple graphs.*

$$G_1 \sim_L G_2 \text{ (up to isomorphism)} \iff M[IAS(G_1)] \cong M[IAS(G_2)]. \qquad \square$$

Here, $M[IAS(G)]$ is the "isotropic matroid" of the graph, G, and $\cong$ indicates a matroid isomorphism. As mentioned in Chapter 5, when we are mapping to or from the empty graph, SWAP operations can be handled classically. This, we noted, could allow us to expand our definition of local equivalence to include graph isomorphisms, which would result in smaller, easier to traverse graphs. Traldi's isotropic matroids directly capture this expanded equivalence relation.

Unfortunately, Traldi does not provide an algorithm for efficiently checking whether isotropic matroids are isomorphic, nor does he indicate whether it should be expected to be easy to check. Perhaps an isotropic matroid formalism for stabilizer states is the ideal formalism to use when tackling such circuit synthesis tasks, but efficient algorithms for identifying isomorphic isotropic matroids would have to be developed to run A*on these reduced quotient graphs.

NISQ era quantum computing, with local Hamiltonians, Pauli basis measurements, and Clifford circuits, provides fertile ground for applications of combinatorial techniques. I consider myself lucky to have found friends who were asking the right questions at the right time, which allowed us to uncover the novel results presented here. And considering the plethora of open problems and tantalizing directions for future research mentioned above, I eagerly look forward to our future investigations.

References

[1] Lane G. Gunderman, Andrew J. Jena, and Luca Dellantonio. *Occam's razor for quantum simulations.* arXiv:2308.01986. 2023.

[2] Sergey Bravyi et al. *Tapering off qubits to simulate fermionic Hamiltonians.* arXiv: 1701.08213. 2017.

[3] Lane G. Gunderman. "Transforming collections of Pauli operators into equivalent collections of Pauli operators over minimal registers". In: *Physical review. A* 107.6 (2023). ISSN: 2469-9926.

[4] Ariel Shlosberg et al. "Adaptive estimation of quantum observables". In: *Quantum (Vienna, Austria)* 7 (2023), pp. 906–. ISSN: 2521-327X.

[5] Jeremy C. Adcock et al. "Mapping graph state orbits under local complementation". In: *Quantum (Vienna, Austria)* 4 (2020), pp. 305–. ISSN: 2521-327X.

[6] André Bouchet. "Graphic presentations of isotropic systems". In: *Journal of combinatorial theory. Series B* 45.1 (1988), pp. 58–76. ISSN: 0095-8956.

[7] Daniel Gottesman. "The Heisenberg Representation of Quantum Computers". In: (1998).

[8] Michael A. Nielsen and Isaac L. Chuang. *Quantum computation and quantum information.* 10th anniversary ed. Cambridge ; Cambridge University Press, 2010. ISBN: 9781107002173.

[9] Phillip Kaye et al. *An introduction to quantum computing.* eng. Oxford ; Oxford University Press, 2007. ISBN: 9786610757619.

[10] John Watrous. *The Theory of Quantum Information.* Cambridge University Press, 2018. ISBN: 9781107180567.

[11] Julia Kempe, Alexei Kitaev, and Oded Regev. "The complexity of the local hamiltonian problem". In: *SIAM journal on computing* 35.5 (2006), pp. 1070–1097. ISSN: 0097-5397.

[12] Daniel Gottesman. "Surviving as a Quantum Computer in a Classical World". Unpublished manuscript. 2016.

[13] Tzu-Ching Yen and Artur F. Izmaylov. "Cartan Subalgebra Approach to Efficient Measurements of Quantum Observables". In: *PRX quantum* 2.4 (2021), pp. 040320–. ISSN: 2691-3399.

[14] Frank Arute et al. "Hartree-Fock on a superconducting qubit quantum computer". In: *Science* 369.6507 (2020), pp. 1084–1089.

[15] John Preskill. "Quantum Computing in the NISQ era and beyond". In: *Quantum (Vienna, Austria)* 2 (2018), pp. 79–. ISSN: 2521-327X.

[16] P.W. Shor. "Algorithms for quantum computation: discrete logarithms and factoring". In: *Proceedings 35th Annual Symposium on Foundations of Computer Science.* IEEE Comput. Soc. Press, 1994, pp. 124–134. ISBN: 0818665807.

[17] Antonio Acín et al. "The quantum technologies roadmap: a European community view". In: *New Journal of Physics* 20.8 (Aug. 2018), p. 080201.

[18] I.M. Georgescu, S. Ashhab, and Franco Nori. "Quantum simulation". In: *Reviews of modern physics* 86.1 (2014), pp. 153–185. ISSN: 0034-6861.

[19] Mari Carmen Bañuls et al. "Simulating lattice gauge theories within quantum technologies". In: *The European physical journal. D, Atomic, molecular, and optical physics* 74.8 (2020). ISSN: 1434-6060.

[20] Jan F. Haase et al. "A resource efficient approach for quantum and classical simulations of gauge theories in particle physics". In: *Quantum (Vienna, Austria)* 5 (2021), pp. 1–20. ISSN: 2521-327X.

[21] Danny Paulson et al. "Simulating 2D Effects in Lattice Gauge Theories on a Quantum Computer". In: *PRX quantum* 2.3 (2021), pp. 030334–. ISSN: 2691-3399.

[22] Yudong Cao et al. "Quantum Chemistry in the Age of Quantum Computing". In: *Chemical reviews* 119.19 (2019), pp. 10856–10915. ISSN: 0009-2665.

[23] Richard P. Feynman. "Simulating physics with computers". In: *International journal of theoretical physics* 21.6-7 (1982), pp. 467–488. ISSN: 0020-7748.

[24] John Watrous. "Quantum Computational Complexity". In: *Computational Complexity.* New York, NY: Springer New York, 2012, pp. 2361–2387. ISBN: 1461417996.

[25] Alán Aspuru-Guzik et al. "Simulated Quantum Computation of Molecular Energies". In: *Science (American Association for the Advancement of Science)* 309.5741 (2005), pp. 1704–1707. ISSN: 0036-8075.

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

[27] Andrew Tranter et al. "A Comparison of the Bravyi–Kitaev and Jordan–Wigner Transformations for the Quantum Simulation of Quantum Chemistry". In: *Journal of chemical theory and computation* 14.11 (2018), pp. 5617–5630. ISSN: 1549-9618.

[28] P. Jordan and E. P. Wigner. "Über das Paulische Äquivalenzverbot". In: *The Collected Works of Eugene Paul Wigner*. Springer Berlin Heidelberg, 1993. ISBN: 366202781X.

[29] Jacob T Seeley, Martin J Richard, and Peter J Love. "The Bravyi-Kitaev transformation for quantum computation of electronic structure". In: *The Journal of chemical physics* 137.22 (2012), pp. 224109–224109. ISSN: 0021-9606.

[30] Sergey B. Bravyi and Alexei Yu Kitaev. "Fermionic Quantum Computation". In: *Annals of physics* 298.1 (2002), pp. 210–226. ISSN: 0003-4916.

[31] Andrew Jena and Michele Mosca. *Partitioning Pauli Operators: in Theory and in Practice*. Waterloo, Ontario, Canada, 2019.

[32] Andrew Jena, Scott N. Genin, and Michele Mosca. "Optimization of variational-quantum-eigensolver measurement by partitioning Pauli operators using multiqubit Clifford gates on noisy intermediate-scale quantum hardware". In: *Physical review. A* 106.4 (2022). ISSN: 2469-9926.

[33] Jarrod R McClean et al. "The theory of variational hybrid quantum-classical algorithms". In: *New journal of physics* 18.2 (2016), pp. 23023–. ISSN: 1367-2630.

[34] 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* 152.12 (2020), pp. 124114–124114. ISSN: 0021-9606.

[35] Andrew Arrasmith et al. *Operator Sampling for Shot-frugal Optimization in Variational Algorithms*. arXiv:2004.06252. 2020.

[36] Ophelia Crawford et al. "Efficient quantum measurement of Pauli operators in the presence of finite sampling error". In: *Quantum (Vienna, Austria)* 5 (2021), pp. 385–. ISSN: 2521-327X.

[37] Hsin-Yuan Huang, Richard Kueng, and John Preskill. "Efficient Estimation of Pauli Observables by Derandomization". In: *Physical review letters* 127.3 (2021), pp. 1–030503. ISSN: 0031-9007.

[38] Giacomo Torlai et al. "Precise measurement of quantum observables with neural-network estimators". In: *Physical review research* 2.2 (2020). ISSN: 2643-1564.

[39] Stefan Hillmich et al. "Decision Diagrams for Quantum Measurements with Shallow Circuits". In: *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 2021, pp. 24–34. ISBN: 9781665416917.

[40] Hsin-Yuan Huang, Richard Kueng, and John Preskill. "Predicting many properties of a quantum system from very few measurements". In: *Nature physics* 16.10 (2020), pp. 1050–1057. ISSN: 1745-2473.

[41] Charles Hadfield et al. "Measurements of Quantum Hamiltonians with Locally-Biased Classical Shadows". In: *Communications in mathematical physics* 391.3 (2022), pp. 951–967. ISSN: 0010-3616.

[42] Charles Hadfield. *Adaptive Pauli Shadows for Energy Estimation.* arXiv:2105.12207. 2021.

[43] Bujiao Wu et al. "Overlapped grouping measurement: A unified framework for measuring quantum states". In: *Quantum (Vienna, Austria)* 7 (2023), pp. 896–. ISSN: 2521-327X.

[44] Masaya Kohda et al. "Quantum expectation-value estimation by computational basis sampling". In: *Physical review research* 4.3 (2022). ISSN: 2643-1564.

[45] Pranav Gokhale et al. *Minimizing State Preparations in Variational Quantum Eigensolver by Partitioning into Commuting Families.* arXiv:1907.13623. 2019.

[46] Ikko Hamamura and Takashi Imamichi. "Efficient evaluation of quantum observables using entangled measurements". In: *npj quantum information* 6.1 (2020). ISSN: 2056-6387.

[47] 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* 16.4 (2020), pp. 2400–2409. ISSN: 1549-9618.

[48] Artur F Izmaylov et al. "Unitary Partitioning Approach to the Measurement Problem in the Variational Quantum Eigensolver Method". In: *Journal of chemical theory and computation* 16.1 (2020), pp. 190–195. ISSN: 1549-9618.

[49] R Raussendorf and H J Briegel. "A one-way quantum computer". In: *Physical review letters* 86.22 (2001), pp. 5188–5191. ISSN: 0031-9007.

[50] M Hein et al. "Entanglement in Graph States and its Applications". In: (2006).

[51] André Bouchet. "Recognizing locally equivalent graphs". In: *Discrete mathematics* 114.1 (1993), pp. 75–86. ISSN: 0012-365X.

[52] Alberto Peruzzo et al. “A variational eigenvalue solver on a photonic quantum processor”. In: *Nature communications* 5.1 (2014), pp. 4213–4213. ISSN: 2041-1723.

[53] Albie Chan et al. *Hybrid variational quantum eigensolvers: merging computational models*. arXiv:2305.19200. 2023.

[54] R.R. Ferguson et al. “Measurement-Based Variational Quantum Eigensolver”. In: *Physical review letters* 126.22 (2021), pp. 1–220501. ISSN: 0031-9007.

[55] Abhinav Kandala et al. “Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets”. In: *Nature (London)* 549.7671 (2017), pp. 242–246. ISSN: 0028-0836.

[56] Guglielmo Mazzola. “Quantum computing for chemistry and physics applications from a Monte Carlo perspective”. In: (2023).

[57] Ryan V. Mishmash et al. “Hierarchical Clifford Transformations to Reduce Entanglement in Quantum Chemistry Wave Functions”. In: *Journal of chemical theory and computation* 19.11 (2023), pp. 3194–3208. ISSN: 1549-9618.

[58] Seonghoon Choi, Ignacio Loaiza, and Artur F. Izmaylov. “Fluid fermionic fragments for optimizing quantum measurements of electronic Hamiltonians in the variational quantum eigensolver”. In: *Quantum (Vienna, Austria)* 7 (2023), pp. 889–. ISSN: 2521-327X.

[59] Philip Richerme et al. “Quantum Computation of Hydrogen Bond Dynamics and Vibrational Spectra”. In: *The journal of physical chemistry letters* 14.32 (2023), pp. 7256–7263. ISSN: 1948-7185.

[60] Chris Sparrow et al. “Simulating the vibrational quantum dynamics of molecules using photonics”. In: *Nature (London)* 557.7707 (2018), pp. 660–667. ISSN: 0028-0836.

[61] Christopher S. Wang et al. “Efficient Multiphoton Sampling of Molecular Vibronic Spectra on a Superconducting Bosonic Processor”. In: *Physical review. X* 10.2 (2020), pp. 021060–. ISSN: 2160-3308.

[62] Navin Khaneja and Steffen J. Glaser. “Cartan decomposition of $SU(2^n)$ and control of spin systems”. In: *Chemical physics* 267.1 (2001), pp. 11–23. ISSN: 0301-0104.

[63] Simon V. Mathis, Guglielmo Mazzola, and Ivano Tavernelli. “Toward scalable simulations of lattice gauge theories on quantum computers”. In: *Physical review. D* 102.9 (2020), pp. 1–. ISSN: 2470-0010.

[64] Giulia Mazzola et al. “Gauge-invariant quantum circuits for $U(1)$ and Yang-Mills lattice gauge theories”. In: *Physical review research* 3.4 (2021). ISSN: 2643-1564.

[65] Lena Funcke, Karl Jansen, and Stefan Kühn. "Exploring the CP -violating Dashen phase in the Schwinger model with tensor networks". In: *Physical review. D* 108.1 (2023). ISSN: 2470-0010.

[66] Alessio Celi et al. "Emerging Two-Dimensional Gauge Theories in Rydberg Configurable Arrays". In: *Physical review. X* 10.2 (2020), pp. 021057–. ISSN: 2160-3308.

[67] Giacomo Torlai et al. "Neural-network quantum state tomography". In: *Nature physics* 14.5 (2018), pp. 447–450. ISSN: 1745-2473.

[68] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. "The quest for a Quantum Neural Network". In: *Quantum information processing* 13.11 (2014), pp. 2567–2586. ISSN: 1570-0755.

[69] S. K. Jeswal and S. Chakraverty. "Recent Developments and Applications in Quantum Neural Network: A Review". In: *Archives of computational methods in engineering* 26.4 (2019), pp. 793–807. ISSN: 1134-3060.

[70] Kerstin Beer et al. "Training deep quantum neural networks". In: *Nature communications* 11.1 (2020), pp. 808–808. ISSN: 2041-1723.

[71] Madelyn Cain et al. *Quantum speedup for combinatorial optimization with flat energy landscapes.* delocalarXiv: 2306.13123v2. 2023.

[72] Nikolaj Moll et al. "Quantum optimization using variational algorithms on near-term quantum devices". In: *Quantum Science and Technology* 3.3 (2018), pp. 30503–. ISSN: 2058-9565.

[73] Jorge Miguel-Ramiro, Alexander Pirker, and Wolfgang Dür. "Optimized Quantum Networks". In: *Quantum (Vienna, Austria)* 7 (2023), pp. 919–. ISSN: 2521-327X.

[74] Jorge Miguel-Ramiro, Ferran Riera-Sàbat, and Wolfgang Dür. "Collective Operations Can Exponentially Enhance Quantum State Verification". In: *Physical review letters* 129.19 (2022), pp. 190504–190504. ISSN: 0031-9007.

[75] L. Tagliacozzo, A. Celi, and M. Lewenstein. "Tensor networks for lattice gauge theories with continuous groups". In: *Physical review. X* 4.4 (2014), pp. 041024–. ISSN: 2160-3308.

[76] Sukhwinder Singh, Robert N. C. Pfeifer, and Guifré Vidal. "Tensor network decompositions in the presence of a global symmetry". In: *Physical review. A, Atomic, molecular, and optical physics* 82.5 (2010). ISSN: 1050-2947.

[77] Román Orús. "Tensor networks for complex quantum systems". In: *Nature reviews physics* 1.9 (2019), pp. 538–550. ISSN: 2522-5820.

[78] Scott Aaronson and Daniel Gottesman. “Improved simulation of stabilizer circuits”. In: *Physical review. A, Atomic, molecular, and optical physics* 70.5 (2004). ISSN: 1050-2947.

[79] Chanchal K. Majumdar and Dipan K. Ghosh. “On Next-Nearest-Neighbor Interaction in Linear Chain. I”. In: *Journal of mathematical physics* 10.8 (1969), pp. 1388–1398. ISSN: 0022-2488.

[80] Chanchal K. Majumdar and Dipan K. Ghosh. “On Next-Nearest-Neighbor Interaction in Linear Chain. II”. In: *Journal of mathematical physics* 10.8 (1969), pp. 1399–1402. ISSN: 0022-2488.

[81] Brent Harrison et al. *Reducing the qubit requirement of Jordan-Wigner encodings of N-mode, K-fermion systems from N to $\lceil \log_2 \binom{N}{K} \rceil$*. arXiv:2211.04501. 2022.

[82] Alexey V. Gorshkov et al. “Universal approach to optimal photon storage in atomic media”. In: *Physical review letters* 98.12 (2007), pp. 123601–123601. ISSN: 0031-9007.

[83] Jess Banks et al. “Pseudospectral Shattering, the Sign Function, and Diagonalization in Nearly Matrix Multiplication Time”. In: *Foundations of computational mathematics* 23.6 (2023), pp. 1959–2047. ISSN: 1615-3375.

[84] Josh Alman and Virginia Vassilevska Williams. “A refined laser method and faster matrix multiplication”. In: *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms*. 2021, pp. 522–539. ISBN: 9781611976465.

[85] Diego Armentano et al. *A stable, polynomial-time algorithm for the eigenpair problem*. arXiv:1505.03290. 2015.

[86] D. A. (Daniel A.) Lidar and Todd A. Brun. *Quantum error correction*. Cambridge: Cambridge University Press, 2013. ISBN: 1-107-42383-X.

[87] Lane G. Gunderman. “Local-dimension-invariant qudit stabilizer codes”. In: *Physical review. A* 101.5 (2020). ISSN: 2469-9926.

[88] A. Ketkar et al. “Nonbinary Stabilizer Codes Over Finite Fields”. In: *IEEE transactions on information theory* 52.11 (2006), pp. 4892–4914. ISSN: 0018-9448.

[89] Stasja Stanisic et al. “Observing ground-state properties of the Fermi-Hubbard model using a scalable algorithm on a quantum computer”. In: *Nature communications* 13.1 (2022), pp. 5743–5743. ISSN: 2041-1723.

[90] Loren Williams. 2023. URL: https://williams.chemistry.gatech.edu/structure/small_molecules/.

[91] John P. Lowe and Kirk A. Peterson. *Quantum chemistry.* 3rd ed. / John P. Lowe, Kirk A. Peterson. Burlington, MA: Elsevier Academic Press, 2006. ISBN: 012457551X.

[92] Cornelius Hempel et al. "Quantum Chemistry Calculations on a Trapped-Ion Quantum Simulator". In: *Physical review. X* 8.3 (2018). ISSN: 2160-3308.

[93] T. E. O'Brien et al. "Purification-based quantum error mitigation of pair-correlated electron simulations". In: *Nature Physics* 19.12 (2023), pp. 1787–1792. ISSN: 1745-2481.

[94] Vincent E. Elfving et al. "Simulating quantum chemistry in the seniority-zero space on qubit-based quantum computers". In: *Physical review. A* 103.3 (2021). ISSN: 2469-9926.

[95] Jarrod R McClean et al. "OpenFermion: the electronic structure package for quantum computers". In: *Quantum Science and Technology* 5.3 (2020), pp. 34014–. ISSN: 2058-9565.

[96] D. Horn, M. Weinstein, and S. Yankielowicz. "Hamiltonian approach to Z(N) lattice gauge theories". In: *Physical review. D, Particles and fields* 19.12 (1979), pp. 3715–3731. ISSN: 0556-2821.

[97] Tatiana A Bespalova and Oleksandr Kyriienko. *Quantum simulation and ground state preparation for the honeycomb Kitaev model.* arXiv:2109.13883. 2021.

[98] Alexei Kitaev. "Anyons in an exactly solved model and beyond". In: *Annals of physics* 321.1 (2006), pp. 2–111. ISSN: 0003-4916.

[99] George C. Knee and William J. Munro. "Optimal Trotterization in universal quantum simulators under faulty control". In: *Physical review. A, Atomic, molecular, and optical physics* 91.5 (2015). ISSN: 1050-2947.

[100] Conor McKeever and Michael Lubasch. "Classically optimized Hamiltonian simulation". In: *Physical review research* 5.2 (2023). ISSN: 2643-1564.

[101] Beatrice Nash, Vlad Gheorghiu, and Michele Mosca. "Quantum circuit optimizations for NISQ architectures". In: *Quantum Science and Technology* 5.2 (2020), pp. 25010–. ISSN: 2058-9565.

[102] Lena Funcke et al. "Dimensional Expressivity Analysis of Parametric Quantum Circuits". In: *Quantum (Vienna, Austria)* 5 (2021), pp. 422–. ISSN: 2521-327X.

[103] Heinz-Peter Breuer and F. (Francesco) Petruccione. *The theory of open quantum systems.* Oxford ; Oxford University Press, 2002. ISBN: 0198520638.

[104] Y. Cao, J. Romero, and A. Aspuru-Guzik. “Potential of quantum computing for drug discovery”. In: *IBM journal of research and development* 62.6 (2018), 6–6:20. ISSN: 0018-8646.

[105] W.M. Itano et al. “Quantum projection noise: Population fluctuations in two-level systems”. In: *Physical review. A, Atomic, molecular, and optical physics* 47.5 (1993), pp. 3554–3570. ISSN: 1050-2947.

[106] M. Cerezo et al. “Variational quantum algorithms”. In: *Nature reviews physics* 3.9 (2021), pp. 625–644. ISSN: 2522-5820.

[107] Cambyse Rouzé and Daniel Stilck França. *Learning quantum many-body systems from a few copies.* arXiv:2107.03333. 2023.

[108] Andrew Jena. *GitHub – VQE Measurement Optimization.* Dec. 2021. URL: https://github.com/AndrewJena/VQE_measurement_optimization.

[109] Coen Bron and Joep Kerbosch. *Algorithm 457: finding all cliques of an undirected graph.* New York, NY, USA, 1973.

[110] Thomas H. Cormen. *Introduction to algorithms.* 3rd ed. Cambridge, Mass: MIT Press, 2009. ISBN: 9780262033848.

[111] Stephan Hoyer, Jascha Sohl-Dickstein, and Sam Greydanus. *Neural reparameterization improves structural optimization.* arXiv:1909.04240. 2019.

[112] Herbert Robbins and Sutton Monro. “A Stochastic Approximation Method”. In: *Ann. Math. Statist. 22, no. 3 (1951), 400-407.* Vol. 22. Institute of Mathematical Statistics, 1951, pp. 400–407. ISBN: 1461295688.

[113] Diederik P Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization.* arXiv:1412.6980. 2017.

[114] Jorge. Nocedal and Stephen J. Wright. *Numerical optimization.* Springer series in operations research. New York: Springer, 1999. ISBN: 0387987932.

[115] P. E. Gill and W. Murray. “Quasi-Newton Methods for Unconstrained Optimization”. In: *IMA journal of applied mathematics* 9.1 (1972), pp. 91–108. ISSN: 0272-4960.

[116] Chigozie Nwankpa et al. *Activation Functions: Comparison of trends in Practice and Research for Deep Learning.* arXiv:1811.03378. 2018.

[117] Zonghan Wu et al. “A Comprehensive Survey on Graph Neural Networks”. In: *IEEE transaction on neural networks and learning systems* 32.1 (2021), pp. 4–24. ISSN: 2162-237X.

[118] Fabian H. L. Essler et al. *The One-Dimensional Hubbard Model*. 1st ed. Cambridge: Cambridge University Press, 2005. ISBN: 0521802628.

[119] Nicholas C Rubin, Ryan Babbush, and Jarrod McClean. “Application of fermionic marginal constraints to hybrid quantum algorithms”. In: *New journal of physics* 20.5 (2018), pp. 53020–. ISSN: 1367-2630.

[120] John Kruschke. “Tutorial: Doing Bayesian Data Analysis with R and BUGS”. In: (2011).

[121] Andrew Gelman et al. *Bayesian data analysis*. Third edition. Chapman & Hall/CRC texts in statistical science. Boca Raton: CRC Press, 2014. ISBN: 9781439840955.

[122] Paolo Fornasini. *The Uncertainty in Physical Measurements: An Introduction to Data Analysis in the Physics Laboratory*. 1. Aufl. New York, NY: Springer-Verlag, 2009. ISBN: 9780387786490.

[123] Roger A. Horn and Charles R. Johnson. *Matrix analysis*. 2nd ed. New York: Cambridge University Press, 2013. ISBN: 9780521548236.

[124] J.W. Moon and L. Moser. “On cliques in graphs”. In: *Israel journal of mathematics* 3.1 (1965), pp. 23–28. ISSN: 0021-2172.

[125] Dong C. Liu and Jorge Nocedal. “On the limited memory BFGS method for large scale optimization”. In: *Mathematical programming* 45.3 (1989), pp. 503–528. ISSN: 0025-5610.

[126] J.F. Haase et al. “Controllable Non-Markovianity for a Spin Qubit in Diamond”. In: *Physical review letters* 121.6 (2018), pp. 060401–060401. ISSN: 0031-9007.

[127] Zachary Pierce Bansingh et al. “Fidelity Overhead for Nonlocal Measurements in Variational Quantum Algorithms”. In: *The journal of physical chemistry. A* 126.39 (2022), pp. 7007–7012. ISSN: 1089-5639.

[128] Ketan N. Patel, Igor L. Markov, and John P. Hayes. “Optimal synthesis of linear reversible circuits”. In: *Quantum Inf. Comput.* 8 (2008), pp. 282–294.

[129] Maarten Van den Nest, Jeroen Dehaene, and Bart De Moor. “Graphical description of the action of local Clifford transformations on graph states”. In: *Phys. Rev. A* 69 (2 2004), p. 022316.

[130] Daniel Grier and Luke Schaeffer. “The Classification of Clifford Gates over Qubits”. In: *Quantum (Vienna, Austria)* 6 (2022), pp. 734–. ISSN: 2521-327X.

[131] Maarten Van den Nest, Jeroen Dehaene, and Bart De Moor. “Efficient algorithm to recognize the local Clifford equivalence of graph states”. In: *Physical review. A, Atomic, molecular, and optical physics* 70.3 (2004). ISSN: 1050-2947.

[132] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. "A Formal Basis for the Heuristic Determination of Minimum Cost Paths". In: *IEEE transactions on systems science and cybernetics* 4.2 (1968), pp. 100–107. ISSN: 0536-1567.

[133] Rina Dechter and Judea Pearl. "Generalized best-first search strategies and the optimality af A*". In: *Journal of the ACM* 32.3 (1985), pp. 505–536. ISSN: 0004-5411.

[134] Jim Geelen et al. "The Grid Theorem for vertex-minors". In: *Journal of combinatorial theory. Series B* 158 (2023), pp. 93–116. ISSN: 0095-8956.

[135] Rose McCarty. *Local Structure for Vertex-Minors.* 2021.

[136] André Bouchet. "Circle Graph Obstructions". In: *Journal of combinatorial theory. Series B* 60.1 (1994), pp. 107–144. ISSN: 0095-8956.

[137] Rose McCarty. "Decomposing a signed graph into rooted circuits". In: (2023).

[138] Donggyu Kim and Sang-il Oum. "Vertex-minors of graphs". In: (2023).

[139] James Davies and Rose McCarty. "Circle graphs are quadratically χ-bounded". In: *The Bulletin of the London Mathematical Society* 53.3 (2021), pp. 673–679. ISSN: 0024-6093.

[140] Huy-Tung Nguyen and Sang-il Oum. "The average cut-rank of graphs". In: *European journal of combinatorics* 90 (2020), pp. 103183–. ISSN: 0195-6698.

[141] Michael Meth et al. *Simulating 2D lattice gauge theories on a qudit quantum computer.* arXiv:2310.12110. 2023.

[142] Ilya G. Ryabinkin, Andrew J. Jena, and Scott N. Genin. "Efficient Construction of Involutory Linear Combinations of Anticommuting Pauli Generators for Large-Scale Iterative Qubit Coupled Cluster Calculations". In: *Journal of chemical theory and computation* 19.6 (2023), pp. 1722–1733. ISSN: 1549-9618.

[143] Daniel Litinski. "A Game of Surface Codes: Large-Scale Quantum Computing with Lattice Surgery". In: *Quantum (Vienna, Austria)* 3 (2019), pp. 128–. ISSN: 2521-327X.

[144] Xiaotong Ni, Oliver Buerschaper, and Maarten Van den Nest. "A non-commuting stabilizer formalism". In: *Journal of mathematical physics* 56.5 (2015), pp. 1–. ISSN: 0022-2488.

[145] Raymond Laflamme et al. "Perfect Quantum Error Correcting Code". In: *Phys. Rev. Lett.* 77 (1 1996), pp. 198–201.

[146] Lorenzo Traldi. “Binary matroids and local complementation”. In: *European journal of combinatorics* 45 (2015), pp. 21–40. ISSN: 0195-6698.

Appendix A

In this appendix, we provide additional details about the physical models employed in the plot of Fig. 3.3. We remark that the symmetries that our method finds is entirely independent on the values of the coefficients within the following Hamiltonians.

$\mathbb{Z}_2$ model

The $\mathbb{Z}_2$ model [54, 53, 20, 96] is a pure (without matter) lattice gauge theory (LGT) possessing a $\mathbb{Z}_2$ symmetry. As shown in Fig. 1, we consider an $N \times N$ lattice (2×2 in the figure) with qubits lying on edges. The electric field of the LGT is described by individual Pauli operators $\mathtt{Z}$ acting over each edge. The magnetic field, on the other side, is characterized by the electric field circulating into each plaquette (i.e., the smallest cell of the lattice). The Hamiltonian is therefore

$$\hat{\mathcal{H}} = -\sum_{p=1}^{n_p} \hat{P}_p + \xi \sum_{q=1}^{n} \hat{Z}_q; \quad \hat{P}_p = \prod_{i \in \square_p} \hat{X}_i, \tag{1}$$

where n_p is the number of plaquettes $\square_p$ ($n_p = 9$ in Fig. 1) and ξ is the coupling constant of the theory [21]. For clarity, we included suffices to the Pauli operators to indicate to which qubit they act (e.g., X_3 indicates $\mathtt{X}$ acts on the third qubit).

Kitaev honeycomb model

As shown in Fig. 1, the Kitaev honeycomb model employs a hexagonal lattice on a torus. Following Ref. [97], the Hamiltonian of the system is

$$\mathcal{H} = J_{\mathrm{x}} \sum_{\langle i,j \rangle \in \mathcal{X}} \mathtt{X}_i \mathtt{X}_j + J_{\mathrm{y}} \sum_{\langle i,j \rangle \in \mathcal{Y}} \mathtt{Y}_i \mathtt{Y}_j + J_{\mathrm{z}} \sum_{\langle i,j \rangle \in \mathcal{Z}} \mathtt{Z}_i \mathtt{Z}_j + \sum_{i=1}^{n} h_{\mathrm{z},i} \mathtt{Z}_i, \tag{3}$$

where $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$ refer to edges aligned to the three defining directions of an hexagon: in the figure, the orange, green and blue, respectively. The last term in Eq. (3) describes $\vec{B}$, such that when all coefficients $h_{\mathrm{z},i}$ are equal to (different from) zero, the model is with (without) magnetic field. The remaining coefficients J_{x}, J_{y} and J_{z} characterize the neighbour-neighbour interactions of the model.

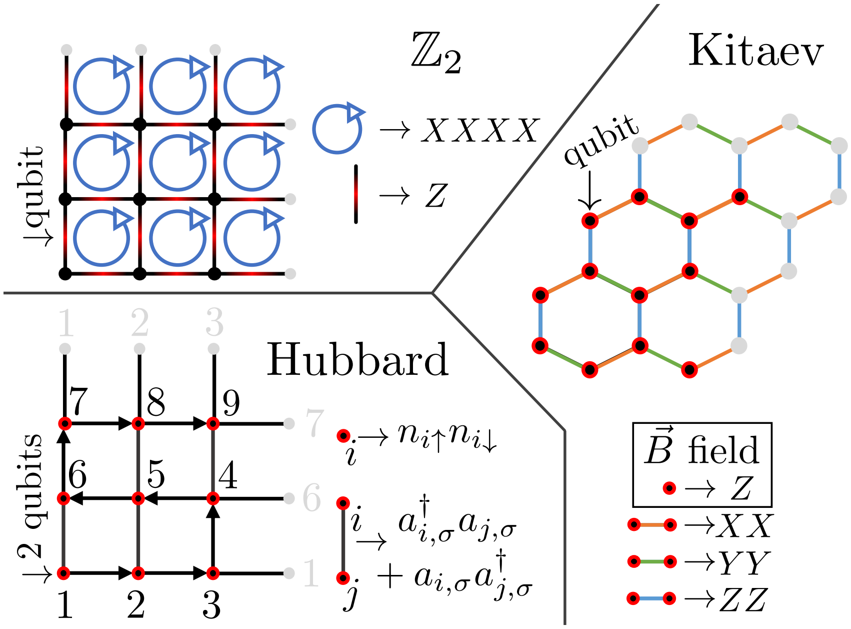


Figure 1: Physical models considered for the plot in Fig. 3.3. *(top left)*: For the $\mathbb{Z}_2$ model [54, 53, 20, 96] qubits lie on the edges of a square lattice. Each qubit individually is acted upon a `Z` operator, and `XXXX` operators act on plaquettes, as indicated by the blue circular arrows. *(right)*: In the Kitaev hexagonal lattice [97, 98], qubits lie on vertices. Depending on the direction of the connection with the neighbouring vertex, the interaction is `XX` (orange), `YY` (green) or `ZZ` (blue). When the magnetic field $\vec{B}$ is present, additional `Z` operators act on each individual qubit. *(bottom left)*: In the Hubbard model [89], each vertex `I` can host two fermions with opposite spin $\sigma \in \{\uparrow, \downarrow\}$, that interact together via $n_{i,\uparrow}n_{i,\downarrow}$. Furthermore, each fermionic species can jump between connected vertices via the flip-flop interactions. As explained in Ref. [89] and in App. 6, to simulate the Hubbard model with a quantum computer, fermions are encoded into qubits via the Jordan-Wigner [28, 21] transformation, resulting in the Hamiltonian obtained from Eqs. (4) and (5) All lattices presented here are the minimal instances considered in Sec. 3.2 Larger ones are obtained by increasing both dimensions equally. Furthermore, gray dots indicated periodic boundary condtions (as explicitly indicated in the Hubbard model).

Hubbard model

As another concrete example we consider the Hubbard model on $n/2$ sites (such that the number of qubits is n). In this case there is an analytic solution for the dimension of the isotropic space, and hence the parallelization size. The Hamiltonian is given by

$$H = -\sum_{\substack{\langle i,j\rangle \\ \sigma\in\{\uparrow,\downarrow\}}} \left(a^{\dagger}_{i\sigma}a_{j\sigma} + a^{\dagger}_{j\sigma}a_{i\sigma}\right) + U\sum_{i=1}^{n/2} n_{i\uparrow}n_{i\downarrow}, \tag{4}$$

where these operators describe two species ($\sigma \in \{\uparrow,\downarrow\}$) of fermions residing at each lattice site (see Fig. 1 and Ref. [89]). The first (second) summation describes interactions between fermions of the same species residing on neighbouring vertices i, j (fermions of different species residing on the same vertex $\mathtt{I}$). U is a parameter describing the theory.

We will assume that there is a single connected lattice. If there is more than one lattice then the savings will be the resultant amount for each lattice. Beyond this, there are no restrictions placed on the connectivity of the spins. This Hamiltonian is translated into qubit operators via the Jordan-Wigner transformation [28]

$$\left(a^{\dagger}_{i\sigma}a_{j\sigma} + a^{\dagger}_{j\sigma}a_{i\sigma}\right) \mapsto (\mathtt{X}_i\mathtt{X}_j + \mathtt{Y}_i\mathtt{Y}_j)\bigotimes_{k=i+1}^{j-1}\mathtt{Z}_k, \tag{5a}$$

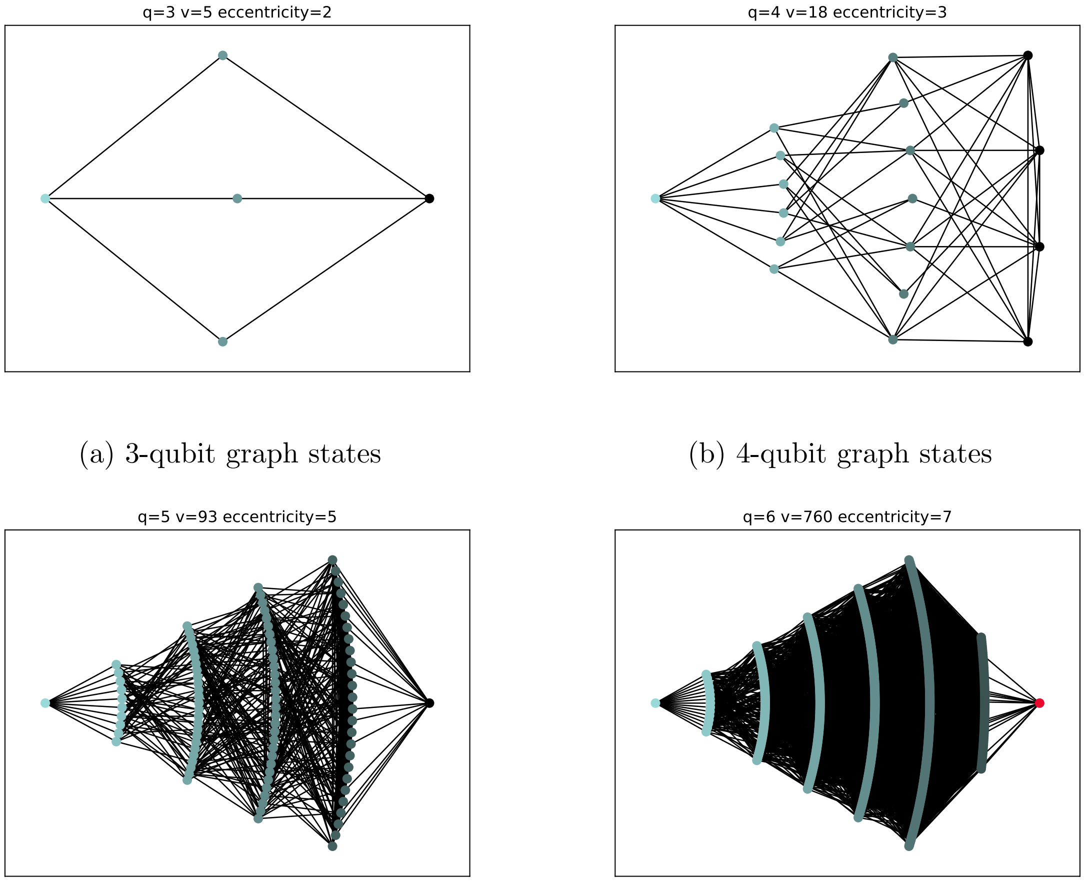
$$n_{i\uparrow}n_{i\downarrow} \mapsto \frac{(\mathtt{I}-\mathtt{Z}_{i\uparrow})(\mathtt{I}-\mathtt{Z}_{i\downarrow})}{4}, \tag{5b}$$

where the numbering follows the "snake pattern" shown in Fig. 1 and described in Ref. [89]. Since we only need a compositionally independent set the latter equation generates $\mathtt{Z}$ operators on each qubit in the ↑ and ↓ lattices. Then the excitation preserving operators may have their $\mathtt{Z}$ operator terms removed without changing our final result. Following this, for each $\mathtt{X}_i\mathtt{X}_j$ we may remove the corresponding $\mathtt{Y}_i\mathtt{Y}_j$ operator since these only differ by $\mathtt{Z}$ operators.

Next, we may cancel out some of the $\mathtt{X}_i\mathtt{X}_j$ terms since they are not all independent. For any loop the $\mathtt{X}_i\mathtt{X}_j$ terms cancel out, and so we may remove a generator to break the loop and all terms will then be independent. Since this holds for all loops, we may repeat this process until the connectivity is a tree of n vertices, and so $n-1$ independent $\mathtt{X}_i\mathtt{X}_j$ Pauli operators will exist for each of the spin lattices. The total number of independent Pauli operators is then $n-2+n$.

To find the rank of the anticommutation matrix, we first note that the spin direction lattices are uncoupled for this computation. The simplest way to find the rank in this case is then to note that for each Z Pauli operator there are a unique pair of X_iX_j operators which do not commute with it, and the converse is also true. Then the anticommutation matrix will be as full rank as possible, which given that the dimension is odd means that the rank will be $dim(M) - 1$, and so the isotropic space is of size 1 for each of the two spin direction lattices. Therefore the total isotropic subspace size is 2, meaning that 2 qubits may be saved through this technique.

Appendix B



q=3 v=5 eccentricity=2

q=4 v=18 eccentricity=3

q=5 v=93 eccentricity=5

q=6 v=760 eccentricity=7

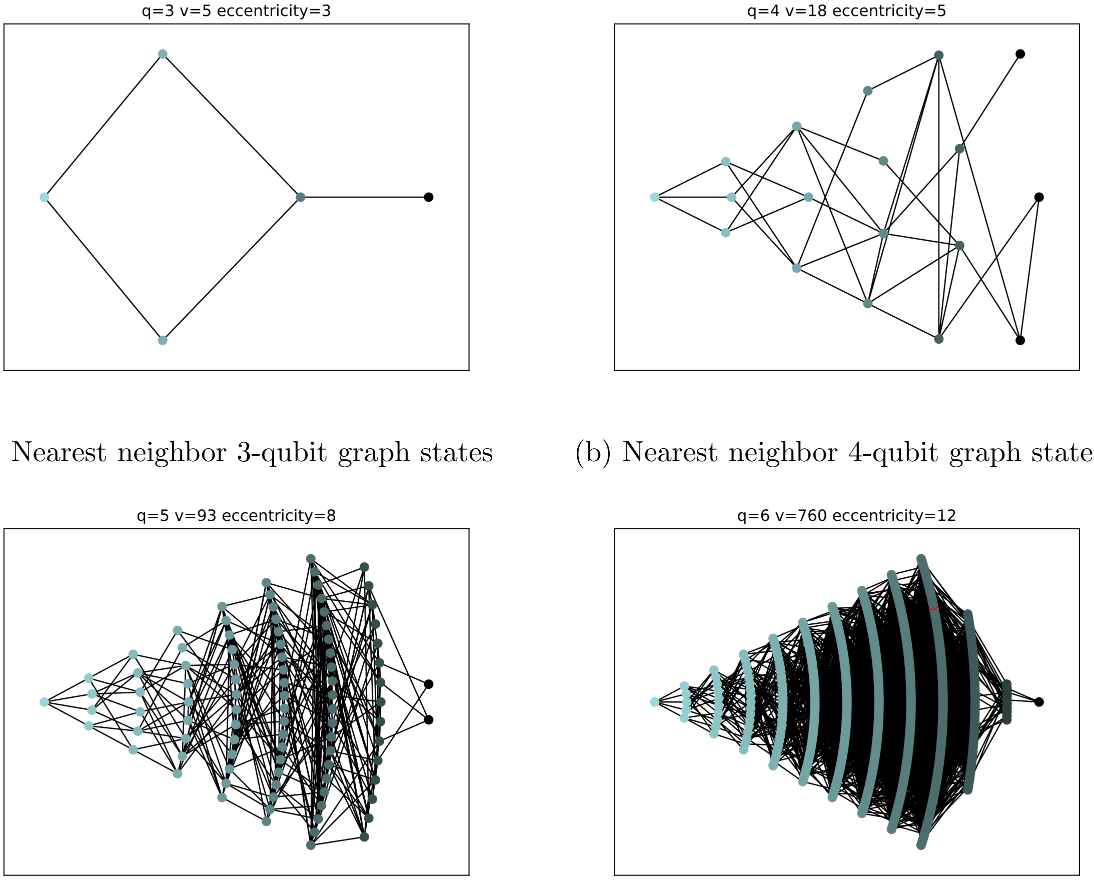
(a) 3-qubit graph states

(b) 4-qubit graph states

(c) 5-qubit graph states

(d) 6-qubit graph states

Appendix C



(a) Nearest neighbor 3-qubit graph states

(b) Nearest neighbor 4-qubit graph states

(c) Nearest neighbor 5-qubit graph states

(d) Nearest neighbor 6-qubit graph states