

Anders Håøy Rokne

Quantum Error Mitigation for near-future Quantum Computation

Master's thesis in Applied Physics

Supervisor: Franz Georg Fuchs

Co-supervisor: Jeroen Danon, Knut-Andreas Lie

June 2021

Anders Håøy Rokne

Quantum Error Mitigation for near-future Quantum Computation

Master's thesis in Applied Physics
Supervisor: Franz Georg Fuchs
Co-supervisor: Jeroen Danon, Knut-Andreas Lie
June 2021

Norwegian University of Science and Technology
Faculty of Natural Sciences
Department of Physics



Abstract

Quantum computing has the potential to solve specific problems exponentially faster than any classical computer. In the short term, however, quantum hardware will be limited both in the number of qubits, the basic unit of quantum information, and the length of useful computations due to the influence of noise. In the long-term, quantum error correction promises to allow for corrections to arbitrary precision over long calculations, but they require lots of ancillary qubits. Our attention therefore turns to quantum error mitigation, which encompasses a range of techniques that aim to reduce the impact of noise in quantum computations.

In this thesis, we study techniques of quantum error mitigation for near-term applications of quantum computers. We aim to assess and compare different techniques for error mitigation, and we propose a variant of the zero-noise extrapolation technique with a novel scheme for circuit-level noise amplification. We applied the zero-noise extrapolation to a representative trial circuit and achieved a relative error of 3.3% in simulations with applied noise models, compared to 30% in the non-mitigated results, in the best case. We also found our noise amplification scheme to outperform an existing scheme for different noise models. However, we found the zero-noise extrapolation technique to come with a considerable computational cost, especially for larger numbers of noise amplified terms. For 2 and 3 terms we still a significant improvement in the error to 14% and 8.3% respectively, yet with a required shot count within one order of magnitude.

We further explored the potential in combining this variant of the zero-noise extrapolation with the error detection scheme. We applied the scheme to the variational quantum eigensolver on the H_2 molecule, using a noisy simulator. The combined scheme outperformed both the extrapolation and the detection schemes on their own. For an error rate of 0.1%, the combined scheme achieved an error within the important benchmark of the chemical accuracy. The error detection scheme on its own did not achieve the chemical accuracy for any cases, however, it increased required circuit shots by less than one order of magnitude in all cases. The scheme thus comes at a much lesser computational cost than the zero-noise extrapolation scheme, although it requires some additional ancillary qubits. Combining the two techniques thus showed potential for greater noise-mitigation whilst still remaining viable for near-term applications.

Sammendrag

Kvantedatamaskiner har potensiale til å kunne løse spesifikke beregningsproblemer eksponentielt raskere enn noen klassisk datamaskin er i stand til. I nærliggende fremtid er derimot kvantedatamaskiner begrensede når det kommer til antallet tilgjengelige qubiter, og i lengden av mulige kvanteberegninger grunnet mye støy. På sikt vil kvantefeilkorreksjon kunne korrigere feil til vilkårlig presisjon over lange kvanteberegninger, men de krever et stort antall ekstra qubiter. Vi fokuserer derfor i stedet på dempe effekten av kvantefeil, så kalt kvantefeilmitigering.

I denne oppgaven studerer vi ulike teknikker for å dempe kvantefeil i anvendelser av kvantedatamaskiner i nær fremtid. Målet vårt er å undersøke og sammenligne ulike slike teknikker, og vi foreslår en variant av nullstøysekstrapoleringsteknikken men en ny metode for støyforsterkning på kvantekrets-nivå. Vi anvendte nullstøysekstrapoleringsteknikken på en representativ testkrets og oppnådde en relativ feil på 3.3%, betydelig bedre enn den opprinnelige feilen i kretsen på 30%, i det beste testede tilfellet. Vår metode gav også bedre resultater enn en eksisterende støyforsterkningsmetode for flere ulike støymodeller. Vi så derimot at denne teknikken ga en betydelig økning i beregningskostnad, spesielt når vi inkluderer mange støyforsterkningsledd. For kun 2 og 3 ledd fant vi fortsatt en betydelig forbedring i den relative feilen, til henholdsvis 14% og 8.3%, men i disse tilfellene økte beregningskostnaden med under én størrelsesorden.

Videre utforsket vi potensialet i å kombinere to ulike teknikker for støydemping. Vi kombinerte vår variant av nullstøysekstrapoleringen men en teknikk for støydemping ved feildeteksjon. Dette anvendte vi på en algoritme for simuleringer av H_2 -molekylet på en simulator med en enkel støymodell. Den kombinerte metoden gav bedre resultater enn både nullstøysekstrapolerings- og feildeteksjons-teknikkene gav for seg selv. Med en feilrate på 0.1% oppnådde den kombinerte metoden en feil under den kjemiske nøyaktigheten. Teknikken med feildeteksjon oppnådde ikke dette for noen av de testede tilfellene, men teknikken kom med en betydelig lavere beregningskostnad enn nullstøysekstrapolering, men den krever noen ekstra qubiter. Å kombinere de to støymitigerings-teknikkene viser dermed potensiale til å kunne oppnå en større grad av støydemping.

Acknowledgements

I would like to thank my supervisor Franz G. Fuchs for much help and great supervision throughout this year, and for giving me this opportunity. I would also like to thank my co-supervisors, Jeroen Danon and Knut-Andreas Lie, for great advice and thorough proof-reading.

I would like thank my mum and dad, Liv Berit Håøy and Knut Inge Rokne, for all love and support throughout my years, as well as my family at large. Also, a huge thank you to all my friends for making these last 5 years such a great time.

Finally, I want to thank Plancks constant, \hbar , for allowing us all to exist.

Contents

Abstract	iii
Sammendrag	v
Acknowledgements	vii
Contents	ix
1 Introduction	1
2 Theory	7
2.1 Linear algebra	7
2.1.1 Inner Product Spaces	7
2.1.2 Homomorphisms	8
2.2 Quantum Mechanics	9
2.2.1 The State Space	9
2.2.2 Time Evolution	9
2.2.3 Measurements	10
2.3 Density Operators and Quantum Channels	10
2.3.1 The Density Operator	10
2.3.2 Trace and Partial Trace	12
2.3.3 The Bloch Sphere	12
2.3.4 Quantum Channels	13
2.3.5 Operator-Sum Representation	14
2.4 Quantum Computation	15
2.4.1 The Quantum Bit	15
2.4.2 Multi-Qubit States	16
2.4.3 Quantum Computations and Circuit Notation	17
2.4.4 Quantum Gates	18
2.4.5 Controlled Gates	20
2.4.6 Gate Identities	21
2.4.7 Basis Sets of Gates	22
2.4.8 Qubit Connectivity and Virtual Qubits	23
2.4.9 Measurements of Quantum Circuits	26
2.4.10 Expectation Values	27
2.4.11 Mean Square Error and Error Estimation	28
3 Error Models for Quantum Computation	31
3.1 Noise as Quantum Channels	31
3.2 Coherent Noise	32

3.3	Incoherent Noise	32
3.3.1	Stochastic Pauli-Noise	32
3.3.2	Depolarizing Noise	33
3.3.3	Amplitude Damping	33
3.3.4	Phase Damping	34
4	Hybrid Algorithms	35
4.1	Variational Quantum Methods	36
4.2	Variational Quantum Eigensolver	36
4.2.1	The Molecular Hamiltonian	37
4.2.2	Jordan–Wigner transformation	38
4.2.3	The UCCSD ansatz	39
4.2.4	Hardware-efficient ansatze	40
4.2.5	Symmetry-preserving ansatze	41
5	Quantum Error Mitigation	43
5.1	Zero-Noise Extrapolation	43
5.1.1	Richardson Extrapolation	44
5.1.2	Noise Amplification by Random Pauli-Gate Sampling	45
5.1.3	Noise Amplification by Repeating CNOT-Gates	46
5.1.4	Variance Scaling	46
5.2	Error Detection	48
6	Experiments and Results	51
6.1	Zero-Noise Extrapolation for CNOT-Gates	52
6.1.1	The SWAP-test Circuit	52
6.1.2	Experiments	54
6.2	Zero-Noise Extrapolation with Error Detection	60
6.2.1	VQE for the Hydrogen Molecule	60
6.2.2	UCCSD Ansatz Circuit with Error Detection	61
6.2.3	Experiments	62
7	Conclusion	67
7.1	Conclusions	67
7.2	Future Work	69
	Bibliography	71
A	The H₂ Molecule	75
B	Pauli-Twirling	79
C	Quantum Error Correction	81
C.1	The Classical Repetition code	81
C.2	The Quantum Repetition code	82
C.3	Digitization of errors	84
C.4	Error Thresholds	84

Chapter 1

Introduction

The onset of quantum computation promises to achieve exponential speed up for specific computational tasks when compared to known classical algorithms. As first proposed by Richard Feynman in the context of simulating quantum physics, exploiting the distinct physical properties of quantum mechanical systems may enable quantum computers to surpass the capabilities of classical computers for certain applications. As recently as 2019, a research team at Google announced that they had achieved *quantum advantage*, or *quantum supremacy*, on a programmable quantum computer by computing a specific task faster than what is thought possible to accomplish on any classical supercomputer [1]. This represents an important milestone on the path to practically applicable quantum computation.

The main idea in quantum computation is to encode information into *qubits* that have quantum mechanical properties. In classical computation the fundamental unit of information is the bit, which have a value of either 0 or 1. The analogue in quantum computation is the qubit. The qubit is physically realized in a quantum mechanical system where the equivalents of the bit-values 0 and 1 are encoded as *quantum states* [2]. For example, a spin up/down particle, where we encode 0 as spin up, and 1 as spin down. Now the qubit may be in the state 0, in the state 1, or in any superposition of the two. Quantum computations are most often formulated as quantum circuits. In general, we first initialize some quantum state on a set of qubits. This quantum state is evolved in time by applying a sequence of quantum operations, or *quantum gates*, analogous to logical gates of classical logical circuits. Lastly, a set of quantum measurements are performed on the culminating final state, resulting in a set of recorded *measurement outcomes*. As quantum measurements are probabilistic, we usually repeat this experiment several times, before interpreting the measurement outcomes by use of classical post-processing to produce the final algorithm output. Often, this involves the estimation of one or more expectation values of some quantum operators with respect to the final quantum state.

The advantages of quantum computation, in very oversimplified terms, can be understood as follows; on a classical computer with n bits of memory, we can only store and operate on a single bit-string of length n at a time. On a quantum computer with n qubits, however, we can create superpositions of all the 2^n basis states corresponding to each of the n -bit bit-strings and do operations on all of them at once. This is sometimes referred to as *quantum parallelism*. Such a state would in general require in the order of 2^n bits to represent on a classical computer, i.e., exponentially increasing with n . The reality is more complex of course. Yet, this offers a rough idea in layman terms of how quantum computers offer something fundamentally distinct from the capabilities of classical computers, with the potential to obtain exponential speed up in certain circumstances.

A first possible application for a quantum computer are simulations of quantum mechanical systems. Intuitively, simulating quantum physics on a machine that possesses the same quantum mechanical properties, such as superpositions and entanglement, can be done more efficiently than on a classical computer. Efficient simulations of such systems could further a variety of fields, such as material science, solid state physics and quantum chemistry. The most prominent quantum algorithm for this task is the *variational quantum eigensolver* (VQE). The VQE algorithm finds ground state energies of molecules by preparing parametrized quantum trial states, or *ansatze*, then minimizing the energy of said state ansatze with respect to the variational parameters. The minimization is done by an optimization algorithm run on a classical computer, which makes this a hybrid quantum-classical algorithm [3–6].

Further advancements in quantum hardware are nonetheless required before quantum computers may become applicable to real-world tasks. There are two main limitations imposed by current and near-future quantum hardware. Firstly, quantum computers are afflicted by noise, as quantum systems are notoriously difficult to keep stable and coherent for longer periods of time. Secondly, they have limited scale by which we mean the number of available and connected qubits. The near-term era of quantum computation has therefore been coined the *noisy intermediate-scale quantum* (NISQ) era.

For the long-term, quantum error correction by quantum error correcting codes promises to provide means to repeatedly detect and correct arbitrary errors throughout the run of a quantum computation [7, 8]. This can be done to arbitrary precision, under the assumption that the error rates are below certain code-specific thresholds and given that enough ancillary qubits are available. This gives promise for the long-term viability of quantum computation, but for such codes the overhead in terms of ancillary qubit requirements is large.

For the short-term, quantum error correcting codes are indeed far too demanding in their qubit requirements. This brings us on to the second main limitation of NISQ-era devices; their limited scale. The state-of-the-art quantum processor used to achieve quantum advantage had 53 connected qubits, and in the near-future, in the range of hundreds to thousands of qubits is considered

to be realistically obtainable. In contrast, consider the Shor’s factorization algorithm, which efficiently factorizes integers. It is perhaps the most well-known quantum algorithm and may have far-reaching consequences as some modern encryption schemes relies on the assumption that integer factorization is hard. However, as a long-depth quantum algorithm, it requires quantum error correction to obtain useful results on a noisy quantum computer. The qubit requirements of Shor’s algorithm have been estimated to be over 20 million qubits [9], far beyond the capabilities of near-future quantum hardware.

Because of the restrictions posed by NISQ-era quantum hardware, the near-term applications for quantum computers are hybrid quantum-classical algorithms. Such algorithms consist of a short-depth quantum computation combined with classical computation. The aforementioned VQE algorithm for molecular simulations is an example of such a hybrid algorithm. Hybrid algorithms also include the quantum approximate optimization algorithm (QAOA), that solves the classical optimization problem known as the max-cut problem, and others [10–12].

To combat noise in NISQ-era quantum devices, for which quantum error correcting codes are not viable, we turn instead to *quantum error mitigation*. With quantum error mitigation we mean techniques that aim to reduce the impact of noise. This is in contrast to quantum error correction that targets exact correction for certain sets of errors. Error mitigation can often be achieved without the large ancillary qubit requirements posed by quantum error correction and may thus be feasible for near-term quantum computers.

There are several quantum error mitigation schemes that have been proposed in recent years [13–22]. One notable technique is the zero-noise extrapolation. The idea is to amplify the noise in our quantum circuits by a set of different known factors, then compute the corresponding noise amplified expectation values. From this, we can then extrapolate the expectation value to the zero-noise case. Another recently proposed technique does error mitigation based on error detection. The error detection scheme borrows ideas from quantum error correction, but instead of identifying the exact errors that have occurred in order to correct them, we simply attempt to detect if some error has occurred. If so, we discard that circuit execution.

In this thesis, we will examine quantum error mitigation techniques with applications to quantum algorithms viable on near-future quantum hardware. We aim to explore how quantum error mitigation may combat noise in NISQ-era quantum hardware, how it may extend the range of viability for certain hybrid algorithms, and what are their drawbacks. In particular, we explore the zero-noise extrapolation technique for error mitigation, and we propose a variation of this technique with a specific noise-amplification scheme. We further examine the error detection scheme and aim to understand the advantages gained from combining both error mitigation schemes. We apply the aforementioned error mitigation techniques to a simple trial quantum circuit, known as the SWAP-test circuit, as well as on the VQE algorithm.

For the zero-noise extrapolation technique, we propose a scheme for amplifying noise in the multi-qubit CNOT quantum operation. The idea is to exploit that the CNOT-gate is its own inverse. By repeating an odd number of CNOT-gates after one another, the action in the ideal case will be equal to a single CNOT-gate, but on a real quantum computer the noise on each CNOT-gate will compound. This noise amplification scheme has, to the best of our knowledge, not been researched previously.

This thesis is laid out in the following way; first, Chapter 2 presents the preliminary theory and prerequisites needed to understand quantum computation, and the further concepts of hybrid quantum-classical algorithms and quantum error mitigation that will be presented. This includes the basics of quantum mechanics, leading up to the density operator and quantum channel formalism for quantum mechanics that are commonly used within quantum information theory and quantum computation, as well as the basics of quantum computation in terms of quantum circuits and quantum gates. Chapter 3 gives a framework in which to model errors and noise in quantum computers in terms of quantum channels operating on density operators and defines some particular useful error models.

To understand the possible near-future applications for quantum computers, Chapter 4 presents the main aspects of hybrid algorithms. The chapter describes the general concepts of variational quantum methods. Several prominent hybrid algorithms, including the aforementioned VQE and QAOA algorithms, are examples of such methods. We further focus on specifically the VQE algorithm, including three different quantum state ansatz. The unitary coupled clusters single-double excitation (UCCSD) ansatz, the hardware efficient ansatz, and a recently proposed symmetry-preserving ansatz.

Chapter 5 gives the general concept of quantum error mitigation techniques that are hoped to be applicable to the aforementioned near-future hybrid algorithms. We focus on the zero-noise extrapolation and the error detection techniques. For the former, we present our proposed scheme for noise amplification by repeating CNOT-gates, as well as an existing noise amplification scheme that uses sampling of random Pauli-gates [23].

Chapter 6 presents the details of the experiments that we performed on the error mitigation techniques, and both presents and discusses the results from said experiments. The zero-noise extrapolation technique was employed to the SWAP-test trial circuit, in order to explore the efficiency of the technique with our proposed noise amplification scheme, as well as to compare this scheme with existing state-of-the-art amplification schemes. To explore the potential in combining this technique with the error detection scheme, as well as to examine how these error mitigation schemes might extend the range of viability for useful near-future hybrid algorithms, we trialed these techniques on the VQE algorithm applied to the H_2 molecule. The code for these experiments is made available at https://github.com/AndersHR/qem__master__thesis. The thesis and results are then summarized in Chapter 7.

Aside from the main part of the thesis, Appendix A presents some specifics for the VQE algorithm on the H_2 molecule. This includes the specific qubit Hamiltonian, as transformed from the fermionic Hamiltonian of H_2 by use of the Jordan–Wigner transformation, as well as the full UCCSD ansatz for this problem. Appendix B presents the concept of Pauli-twirling, a technique that transforms general error channels into stochastic Pauli-noise channels. Appendix C gives an introduction to the concepts of quantum error correction by stabilizer codes and explains why they are useful for the long-term but perhaps not feasible for the near-future. These concepts are included for the interested reader, but are not mandatory in order to understand the main subjects of this thesis.

Chapter 2

Theory

This chapter presents and explains the preliminary theory that is needed to understand quantum computation in general, as well as the concepts that will be presented later. We define a few key concepts of linear algebra, which forms the mathematical foundation of quantum mechanics, and present the basics of quantum mechanics. We formulate quantum information theory within the density operator and quantum channel formalism. This formalism generalizes the concepts of quantum mechanics to open systems and probabilistic ensembles of states. The formalism as presented is commonly used within the field of quantum computation, and will prove especially useful when modelling noise in real quantum computers.

The theory presented here is well-established. The interested reader can find the subjects explained and explored more in detail in textbooks such as *Quantum Computation and Quantum Information* by Nielsen and Chuang, and others [2, 24–26].

2.1 Linear algebra

The basis of quantum mechanics is that of linear algebra. This section presents some concepts that will be important when defining the postulates of quantum mechanics, and for the density operator and quantum channel formulation of quantum mechanics. We define the inner product space, which will be crucial for the concept of quantum-mechanical state vectors, and the homomorphism. The latter will become important when defining the quantum density operator, and the quantum channel, which are important concepts in quantum information theory and quantum computation.

2.1.1 Inner Product Spaces

An *inner product space* is a vector space V over a field \mathbb{F} (e.g., the real numbers \mathbb{R} or the complex numbers \mathbb{C}), equipped with an inner product

$$\langle \cdot, \cdot \rangle : V \times V \rightarrow \mathbb{F} \tag{2.1}$$

which is a map that for all $x, y, z \in V$ and $\alpha \in \mathbb{F}$ fulfills the following properties:

1. Linearity:

$$\langle \alpha x, y \rangle = \alpha \langle x, y \rangle, \quad (2.2)$$

$$\langle x + z, y \rangle = \langle x, y \rangle + \langle z, y \rangle. \quad (2.3)$$

2. Conjugate symmetry:

$$\overline{\langle x, y \rangle} = \langle y, x \rangle. \quad (2.4)$$

3. Positive semi definite:

$$\langle x, x \rangle > 0 \Leftrightarrow x \neq 0. \quad (2.5)$$

The inner product $\langle \cdot, \cdot \rangle$ of an inner-product space V induces a norm $\|\cdot\|$; that is, a map

$$\|\cdot\| : V \rightarrow \mathbb{R}, \quad (2.6)$$

defined for all $x \in V$ as

$$\|x\| = \sqrt{\langle x, x \rangle}. \quad (2.7)$$

2.1.2 Homomorphisms

A *homomorphism* between two inner product spaces A and B is a linear map $F : A \rightarrow B$ that preserves the inner product. By this we mean the following: denote $\langle \cdot, \cdot \rangle_A$ as the inner product on A , and $\langle \cdot, \cdot \rangle_B$ as the inner product on B . Then

$$\langle F(x), F(y) \rangle_B = \langle x, y \rangle_A \quad (2.8)$$

holds for all $x, y \in A$.

Denote the space of homomorphisms between inner product spaces A and B as $\text{Hom}(A, B)$. A homomorphism from a space V onto itself is called an *endomorphism*, and the space of endomorphisms on an inner product space V is denoted $\text{End}(V)$; i.e., $\text{End}(V) = \text{Hom}(V, V)$.

For a homomorphism $S \in \text{Hom}(A, B)$ on inner product spaces A and B with inner products $\langle \cdot, \cdot \rangle_A$ and $\langle \cdot, \cdot \rangle_B$, respectively, the *adjoint* or *Hermitian conjugate* of S , denoted S^\dagger , is the unique operator such that

$$\langle S(a), b \rangle_B = \langle a, S^\dagger(b) \rangle_A \quad (2.9)$$

for all $a \in A$ and $b \in B$. Note that $(S^\dagger)^\dagger = S$. Furthermore, S is called

- normal if $SS^\dagger = S^\dagger S$,
- unitary if $SS^\dagger = S^\dagger S = \mathbb{I}$, where \mathbb{I} is the identity operator,
- Hermitian if $S^\dagger = S$.

For S being an endomorphism on an inner product space A , S is said to be *positive* if $0 \leq \langle a, S(a) \rangle_A$ for all $a \in A$.

2.2 Quantum Mechanics

The postulates of quantum mechanics provide a mathematical framework for quantum physics. Note that the postulates in themselves do not offer any physics, rather a foundation in which to model quantum-mechanical systems. There exist several, equivalent formulations of the postulates. We present here an algebraic formulation that most straight-forwardly leads up to the density operator formalism, which is the commonly used formalism in quantum computation and quantum information theory [2, 25].

2.2.1 The State Space

Postulate 1. For every isolated physical system, there is associated a complex Hilbert space H , known as the *state space*. The state of the system is completely described by a vector $|\psi\rangle \in H$, the *state vector*, with norm equal to 1.

The state vector is unique up to a global phase factor, $|\psi\rangle \rightarrow e^{i\alpha} |\psi\rangle$ for any $\alpha \in \mathbb{R}$. This will become clear when we consider quantum measurements later in Section 2.2.3. For every state vector $|\psi\rangle$, which is called a *ket* vector, there exists a corresponding *bra* vector $\langle\psi|$ that is a vector in its dual space. The inner product between two state vectors $|\psi\rangle$ and $|\phi\rangle$ is denoted $\langle\phi, \psi\rangle = \langle\phi| \cdot |\psi\rangle = \langle\phi|\psi\rangle$. This notation is referred to as the *Bra-Ket notation*.

2.2.2 Time Evolution

Postulate 2. Time evolution of a closed quantum system in state $|\psi\rangle \in H$ is described by a unitary operator $U : H \rightarrow H$ derived from the time-dependent Schrödinger equation

$$i\hbar \frac{d}{dt} |\psi\rangle = \mathcal{H} |\psi\rangle, \quad (2.10)$$

where \mathcal{H} is the Hamiltonian of the system and \hbar is the Heisenberg's reduced constant.

The Hamiltonian is a Hermitian operator, whose eigenvalues are energies $\{E_i\}_i$ with eigenvectors $\{|i\rangle\}_i$. The eigenvectors form a complete basis for the Hilbert space, and the state vector can thus be expressed as a linear combination $|\psi\rangle = \sum_i c_i |i\rangle$ for a set of complex coefficients c_i .

The time evolution of a state vector $|\psi\rangle$ from time t' to t , given a time-independent Hamiltonian $\mathcal{H}(t) = \mathcal{H}$, is thus given as

$$U(t, t') |\psi\rangle = e^{-ih(t-t')\mathcal{H}} |\psi\rangle = \sum_i c_i e^{-ihE_i(t-t')} |i\rangle. \quad (2.11)$$

It is common practice in literature to set $\hbar = 1$ by rescaling the units, in order to ease notation.

2.2.3 Measurements

Postulate 3. For every measurable physical quantity of a system, there is associated an Hermitian operator A with eigenvalues a and eigenvectors $|a\rangle$. A quantum measurement will probabilistically return a measurement result equal to one of the eigenvalues a . As A is Hermitian operator there exist a spectral decomposition

$$A = \sum_a a P_a \quad (2.12)$$

where P_a is the projection operator onto the eigenspace associated with the eigenvalue a . If the system is in state $|\psi\rangle$, then the probability of a measurement obtaining the measurement result a is

$$p_a = \langle \psi | P_a | \psi \rangle, \quad (2.13)$$

and the state will afterwards be left in the post-measurement state

$$|\psi'\rangle = \frac{U P_a |\psi\rangle}{\sqrt{\langle \psi | P_a^\dagger P_a | \psi \rangle}}, \quad (2.14)$$

where U is some possible time evolution operator. An equivalent formulation of quantum measurements can be written in terms of the set of measurement operators $\{M_a\}_a$, where $M_a = U P_a$.

2.3 Density Operators and Quantum Channels

Until now we have considered the description of a quantum system in terms of a wave function $|\psi\rangle$ as a vector in a Hilbert space H . In Section 2.3.1 we introduce the density operator formalism, where the quantum system is described by a density operator ρ , that offers a more general description. This formalism also let us describe *open* quantum systems, quantum *subsystem*, and *ensembles* of state.

In Section 2.3.4, we further define *quantum channels* as means to describe general quantum operations on density operators. In the quantum channel formalism, we define general maps between density operators in terms of *completely-positive trace-preserving maps*. The operations we are interested in may for example be controlled operations in a quantum computer, as a part of a quantum computation, or it may be uncontrollable noise that is present in physical quantum devices.

2.3.1 The Density Operator

The *density operator* gives us a more flexible way of defining quantum states. The concept will let us describe systems where we do not have perfect information, for example subsystems or open systems.

The density operator also referred to as a *density matrix*, is an operator $\rho \in \text{End}(H)$ for a Hilbert space H such that

$$\rho = \sum_k p_k |\psi_k\rangle \langle \psi_k|, \quad (2.15)$$

where $|\psi_k\rangle \in H$ are orthogonal quantum states and the coefficients p_k form a probability distribution; i.e., $p_k \geq 0$ for all k and

$$\sum_k p_k = 1. \quad (2.16)$$

Note that the term *state* is often used interchangeably for quantum states $|\psi\rangle$ and density operators ρ . Whether we mean one or the other becomes clear from the context. A density operator is said to be *pure* if it can be written as

$$\rho = |\psi\rangle \langle \psi|, \quad (2.17)$$

for some state vector $|\psi\rangle$. A density operator which is not pure is said to be *mixed*.

A closed, isolated quantum system described by the quantum state $|\psi\rangle$ is in the density-operator formalism described by the pure density operator $|\psi\rangle \langle \psi|$. As an example, take the composite state $|\psi_{AB}\rangle \in H_A \otimes H_B$ given by

$$|\psi_{AB}\rangle = \frac{1}{\sqrt{2}} (|0\rangle_A \otimes |0\rangle_B + |1\rangle_A \otimes |1\rangle_B) \equiv \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle), \quad (2.18)$$

that gives the corresponding density operator as the pure density operator

$$\rho_{AB} = |\psi_{AB}\rangle \langle \psi_{AB}| = \frac{1}{2} (|00\rangle \langle 00| + |00\rangle \langle 11| + |11\rangle \langle 00| + |11\rangle \langle 11|) \quad (2.19)$$

for $|0\rangle$ and $|1\rangle$ being two orthonormal basis states for H_A and H_B . Compare this to the following example of a mixed state

$$\rho' = \frac{1}{2} (|00\rangle \langle 00| + |11\rangle \langle 11|). \quad (2.20)$$

Density operators are particularly useful for describing ensembles of states. As an example, assume that we start out with a quantum state $|\psi\rangle$ and an error occurs with probability p that transforms the state into $|\psi'\rangle = M|\psi\rangle$. We do not know if an error has occurred, only the probability that it has done so. The state afterwards can then be described by the mixed state

$$\rho' = (1-p) |\psi\rangle \langle \psi| + p |\psi'\rangle \langle \psi'|. \quad (2.21)$$

2.3.2 Trace and Partial Trace

The *trace* of a density operator, and later the *partial trace*, are concepts that will become useful for both analysis, and for later definitions. We define the trace of a density operator $\rho \in \text{End}(H)$ as

$$\text{Tr}[\rho] = \sum_l \langle e_l | \rho | e_l \rangle \quad (2.22)$$

for $\{|e_L\rangle\}_l$ being some orthonormal basis of H . The requirement given in Equation (2.16) is then equivalent to requiring

$$\text{Tr}[\rho] = 1. \quad (2.23)$$

This becomes evident by choosing a basis such that $|e_l\rangle = |\psi_l\rangle$. Furthermore, an operator ρ is pure if and only if $\text{Tr}[\rho^2] = 1$.

For a composite system AB and $\rho \in \text{End}(H_A \otimes H_B)$, define the *partial trace* of ρ onto A as

$$\text{Tr}_B[\rho] = \sum_l \langle e_l | \rho | e_l \rangle \quad (2.24)$$

for $\{|e_L\rangle\}_l$ being some orthonormal basis of H_B . The resulting operator $\rho_A \equiv \text{Tr}_B[\rho]$ is a density operator in $\text{End}(H_A)$.

For any density operator $\rho_A \in \text{End}(H_A)$ one can always find a density operator $\rho_{AB} \in \text{End}(H_{AB})$ on some larger composite system AB such that ρ_{AB} is pure and $\text{Tr}_B[\rho_{AB}] = \rho_A$. ρ_{AB} is then called a *purification* of ρ_A .

As an example, let us look at the pure state $\rho_{AB} = |\psi_{AB}\rangle \langle \psi_{AB}|$ as given in Equation (2.19). Taking the partial trace of ρ_{AB} onto A we can describe the state on

$$\rho_A = \text{Tr}_B[|\psi_{AB}\rangle \langle \psi_{AB}|] = \frac{1}{2}(|0\rangle \langle 0|_A + |1\rangle \langle 1|_A). \quad (2.25)$$

The expectation value of an observable M with respect to some density operator ρ is given as

$$\langle M \rangle_\rho = \text{Tr}[M\rho]. \quad (2.26)$$

2.3.3 The Bloch Sphere

The *Bloch sphere* representation provides a geometrical representation for density operators on a certain class of quantum-mechanical systems; quantum systems of dimension equal to 2, e.g., a qubit system.

Theorem 2.1. Any density operator ρ on a quantum system of dimension 2 can be written on the form

$$\rho = \frac{1}{2}(\mathbb{I} + \vec{r} \cdot \vec{\sigma}) \quad (2.27)$$

for a real vector $\vec{r} = (r_x, r_y, r_z)$, such that $\|\vec{r}\| \leq 1$ and $\vec{\sigma} = (\sigma_x, \sigma_y, \sigma_z)$, where $\{\sigma_i\}_{i=x,y,z}$ are the Pauli matrices defined as

$$\sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad \sigma_y = \begin{bmatrix} 0 & i \\ -i & 0 \end{bmatrix}, \quad \sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \quad (2.28)$$

Thus the Bloch sphere representation gives a geometrical representation for general density operators ρ on 2-dimensional quantum systems, in terms of the vector $\vec{r} \in \mathbb{R}^3$. The vector \vec{r} is referred to as the *Bloch vector* of ρ . This can be represented as in Figure 2.1.

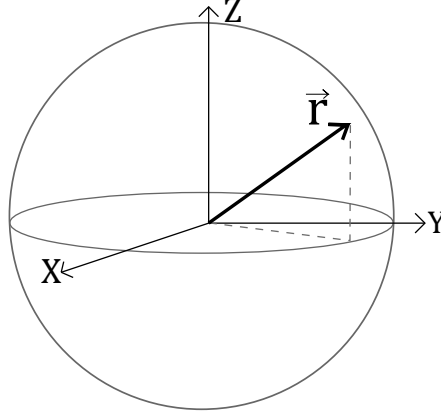


Figure 2.1: A geometrical representation of a density operator $\rho = \frac{1}{2}(\mathbb{I} + \vec{r} \cdot \vec{\sigma})$ by its Bloch vector \vec{r} on the Bloch sphere.

2.3.4 Quantum Channels

So far, we have introduced the density operator as a general mean to describe states of quantum systems. We further need a formalism to describe changes occurring to such systems, as they are evolved in time, i.e., quantum operations. As the state of the system at all times may be described by a density operator the minimal requirement is that the quantum operation maps density operators to density operators. The approach of *quantum channels* is to recognise the general properties of density operators and define operations as maps that preserve these properties.

We define a *quantum channel*, or *quantum operation*, \mathcal{E} as a *trace-preserving completely positive linear map* from density operators $\rho \in \text{End}(H_A)$ to density operators $\rho' \in \text{End}(H_B)$,

$$\mathcal{E} : \text{End}(H_A) \rightarrow \text{End}(H_B). \quad (2.29)$$

With trace-preserving we mean that the trace of ρ is preserved under \mathcal{E} , i.e.,

$$\text{Tr}[\mathcal{E}(\rho)] = \text{Tr}[\rho]. \quad (2.30)$$

A map $\mathcal{E} : \text{End}(H_A) \rightarrow \text{End}(H_B)$ is said to be *positive* if it fulfills the property that if $\rho \in \text{End}(H_A)$ is a positive operator, then $\mathcal{E}(\rho) \in \text{End}(H_B)$ is also a positive operator.

A map $\mathcal{E} : \text{End}(H_A) \rightarrow \text{End}(H_B)$ is said to be *completely positive* if it fulfills the slightly more restrictive restriction that if $\rho \in \text{End}(H_R \otimes H_A)$ a positive operator, then $(\mathbb{I} \otimes \mathcal{E})(\rho) \in \text{End}(H_R \otimes H_B)$ a positive operator.

The motivation behind defining quantum channels as trace-preserving and completely positive maps is to preserve the properties of quantum density operators, which are positive operators with trace equal to one. The stricter requirement of complete positivity ensures that a quantum channel acting locally on a part of a system still preserves positivity of the total, composite state.

A special class of quantum channels are the unitary channels $\mathcal{E}(\rho) = U\rho U^\dagger$ for U a unitary operator. The unitary evolution of a quantum state vector $|\psi\rangle$ by a unitary operator U , as described in Section 2.2.2, can be written within the density operator formalism as the unitary quantum channel $\mathcal{E}(|\psi\rangle\langle\psi|) = U|\psi\rangle\langle\psi|U^\dagger$ on the pure state $|\psi\rangle\langle\psi|$. Note that a unitary channel preserves purity, as

$$\rho' = \mathcal{E}(|\psi\rangle\langle\psi|) = U|\psi\rangle\langle\psi|U^\dagger = |\psi'\rangle\langle\psi'|, \quad (2.31)$$

where we have defined $|\psi'\rangle = U|\psi\rangle$.

Theorem 2.2 (The Stinespring Dilation). For any quantum channel $\mathcal{E} : \text{End}(H_A) \rightarrow \text{End}(H_B)$ there exists a unitary $U \in \text{Hom}(\text{End}(H_A), \text{End}(H_B) \otimes \text{End}(H_C))$, and a unitary $\tilde{U} \in \text{Hom}(\text{End}(H_A) \otimes \text{End}(H_C), \text{End}(H_B) \otimes \text{End}(H_C))$, such that

$$\mathcal{E}(\rho) = \text{Tr}_C[\tilde{U}(\rho \otimes |e\rangle\langle e|_C)\tilde{U}^\dagger] = \text{Tr}_C[U\rho U^\dagger] \quad (2.32)$$

for any $\rho \in \text{End}(H_A)$; $|e\rangle_C$ is a unit vector in an ancillary Hilbert space H_C .

The Stinespring dilation gives the interpretation that any quantum channel \mathcal{E} can be seen as a unitary channel, with unitary \tilde{U} , on a larger composite system $\rho \otimes |e\rangle\langle e|_C \in \text{End}(H_A \otimes H_C)$. The channel acting locally on the subsystem of interest is found by tracing out the system C .

Furthermore, we define compositions of quantum channels through the \circ operator, defined as

$$\tilde{\mathcal{N}}(\rho) = \mathcal{E} \circ \mathcal{N}(\rho) = \mathcal{E}(\mathcal{N}(\rho)), \quad (2.33)$$

where we have defined the quantum channel $\tilde{\mathcal{N}} = \mathcal{E} \circ \mathcal{N}$.

2.3.5 Operator-Sum Representation

An efficient representation of general quantum channels is the *operator-sum representation*. By Equation (2.32) we can write any arbitrary quantum channel \mathcal{E} as

$$\mathcal{E}(\rho) = \text{Tr}_C[U\rho U^\dagger] = \sum_i \langle e_i|U\rho U^\dagger|e_i\rangle_C = \sum_i E_i\rho E_i^\dagger, \quad (2.34)$$

where $\{|e_i\rangle_C\}_i$ is some basis for the ancillary Hilbert space H_C , and we have defined the operators $E_i \equiv U|e_i\rangle_C$. The operators $\{E_i\}_i$ are called the *Kraus operators* of \mathcal{E} .

Note the following restriction, which must hold for any valid set of Kraus operators describing a quantum channel,

$$\sum_i E_i^\dagger E_i = \sum_i U^\dagger |e_i\rangle \langle e_i|_C U = U^\dagger \left(\sum_i |e_i\rangle \langle e_i|_C \right) U = U^\dagger \mathbb{I} U = \mathbb{I}. \quad (2.35)$$

We have used that $\sum_i |e_i\rangle \langle e_i| = \mathbb{I}$, for $\{|e_i\rangle\}_i$ being a complete, orthonormal basis, and \mathbb{I} being the identity operator, and that U is unitary. Note that the freedom in choosing the basis $\{|e_i\rangle\}_i$ implies that the Kraus operators are not uniquely defined.

As an example, take the error channel as described in Equation (2.21), where the pure state $|\psi\rangle \langle\psi|$ is transformed into a mixed state after an error M may have occurred with probability p . This case can be described as a quantum channel by defining the Kraus operators

$$E_0 = \sqrt{1-p} \mathbb{I}, \quad E_1 = \sqrt{p} M. \quad (2.36)$$

These completely describe the given quantum error channel \mathcal{E} , as

$$\begin{aligned} \mathcal{E}(\rho) &= E_0 |\psi\rangle \langle\psi| E_0^\dagger + E_1 |\psi\rangle \langle\psi| E_1^\dagger \\ &= (1-p) \mathbb{I} |\psi\rangle \langle\psi| \mathbb{I} + p M |\psi\rangle \langle\psi| M^\dagger \\ &= (1-p) |\psi\rangle \langle\psi| + p |\psi'\rangle \langle\psi'| = \rho'. \end{aligned} \quad (2.37)$$

They also fulfill the requirement given in Equation (2.35), as

$$E_0^\dagger E_0 + E_1^\dagger E_1 = (1-p) \mathbb{I} + p M^\dagger M = (1-p+p) \mathbb{I} = \mathbb{I}, \quad (2.38)$$

where we use that M is unitary.

2.4 Quantum Computation

So far, we have presented the basics of quantum mechanics in Section 2.2, and the density operator and quantum channel formalism in Section 2.3. This chapter employs the aforementioned theory to build a basis in which to model *quantum computations*.

2.4.1 The Quantum Bit

In classical computation the fundamental unit of information is the bit, which has a value of either 0 or 1. Analogously, the quantum bit, or qubit, forms the most fundamental unit of information for quantum computation. The qubit is encoded into some quantum system, taking advantage of the distinct quantum-mechanical properties of superposition and entanglement. An arbitrary qubit state may be written

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle, \quad (2.39)$$

for $\alpha, \beta \in \mathbb{C}$ and $\{|0\rangle, |1\rangle\}$ being the *computational basis*, which is some orthonormal basis spanning the Hilbert space $H \cong \mathbb{C}^2$ of the qubit system. When measuring the state $|\psi\rangle$ in the computational basis, we get a measurement result of 0 with probability $|\langle 0|\psi\rangle|^2 = |\alpha|^2$ and 1 with probability $|\langle 1|\psi\rangle|^2 = |\beta|^2$. Normalization of the state $|\psi\rangle$ requires that $|\alpha|^2 + |\beta|^2 = 1$.

Note that there is nothing inherently special about the computational basis and other bases may be constructed. For example the $\{|+\rangle, |-\rangle\}$ basis can be constructed as

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \quad |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle), \quad (2.40)$$

which is an equivalent orthonormal, complete basis for the same Hilbert space. Still, it is useful to define a canonical basis for which to write out qubit states.

2.4.2 Multi-Qubit States

To construct the space of multi qubit states, we take the tensor product of single-qubit state spaces. For a 2-qubit system with qubits A and B, we take the Hilbert space of the combined system to be

$$H_{AB} = H_A \otimes H_B, \quad (2.41)$$

on which a general state can be written as

$$|\psi\rangle = \sum_{x \in \{0,1\}} \sum_{y \in \{0,1\}} \alpha_{xy} |xy\rangle. \quad (2.42)$$

It is common in literature to write the tensor product of two qubit states $|x\rangle \otimes |y\rangle$ as $|xy\rangle$. A complete, orthonormal basis for the 2-qubit space is then $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$.

We extend this to an arbitrary number of qubits. An n -qubit system will be of dimension 2^n , and we denote the computational basis states as $|z\rangle$, where $z \in \{0, 1\}^n$ are the n -bit bitstrings, which can be written $z = z_0 z_1 z_2 \dots z_{2^n-1}$ for $z_k \in \{0, 1\}$. An arbitrary state $|\psi\rangle$ on an n -qubit system is thus written

$$|\psi\rangle = \sum_{z \in \{0,1\}^n} c_z |z\rangle \quad (2.43)$$

with coefficients c_z such that $\sum_z |c_z|^2 = 1$. A measurement in the computational basis gives outcome z with probability $|c_z|^2$.

An alternative way to represent n -qubit states is by integers $i = 0, 1, \dots, 2^n - 1$, such that

$$|z\rangle = |z_0 z_1 \dots z_{2^n-1}\rangle = |i\rangle, \quad i = \sum_{k=0}^{2^n-1} z_k \cdot 2^k, \quad (2.44)$$

where z is a binary string $z = z_0 z_1 \dots z_{2^n-1}$, $z_k \in \{0, 1\} \forall k$. This eases notation in some circumstances. We will write multi-qubit states in the bit string representation, unless otherwise specified.

2.4.3 Quantum Computations and Circuit Notation

A *quantum computation*, or *quantum algorithm*, consists of the preparation of a multi-qubit initial state $|\psi\rangle_0$, the evolution of this state by a sequence of unitary operators acting on one or more of the qubits at a time, and projective measurements of some or all of the qubits.

We will write quantum computations in terms of *quantum circuits*. The notation is analogous to that of classical *logical circuits*. Logical circuits model classical computation on the lowest level of abstractions, in terms of wires, representing bits, and logical operators acting on them. Some examples of classical logical operators are AND-gates and OR-gates.

At the leftmost end of a quantum circuit, we write the initial state of each qubit. Each qubit is assigned a wire. We follow the evolution of a qubit state by following its wire from left to right. Thus, the circuit

$$|\psi\rangle \text{ —————} \quad (2.45)$$

represents a single qubit initialized in state $|\psi\rangle$.

The states are evolved by applying *quantum gates*, representing unitary operators acting on one or several qubits, in sequence. These are applied in time in order from left to right. A bare wire represents no evolution, or equivalently an application of the identity operator \mathbb{I} . The quantum circuit

$$\begin{array}{c} |\psi\rangle \text{ —} \boxed{U} \text{ —} \\ |\phi\rangle \text{ —————} \end{array} \boxed{V} \text{ —} \quad , \quad (2.46)$$

is thus equivalent to acting on the state $|\psi\rangle \otimes |\phi\rangle$ with the operator $V_{12}(U_1 \otimes \mathbb{I}_2)$.

Measurements are represented by a special class of gates, denoted by a meter. Doubled wires represents classical bits. The measurement of a single qubit is represented by the circuit

$$\text{—————} \boxed{\text{meter}} = x, \quad (2.47)$$

where the measurement outcome is stored in the classical bit x . All measurements are performed in the computational basis $\{|0\rangle, |1\rangle\}$, unless otherwise specified.

As an example, combining all the aspects of quantum circuits that we have defined, take the quantum circuit

$$\begin{array}{c} |q_0\rangle \text{ —————} \\ |q_1\rangle \text{ —————} \\ |q_2\rangle \text{ —} \boxed{V} \text{ —————} \end{array} \boxed{U} \text{ —} \boxed{M} \text{ —} \boxed{\text{meter}} = a \quad (2.48)$$

$$\text{—————} \boxed{E} \text{ —} \boxed{\text{meter}} = b.$$

This circuit is equivalent the unitary evolution of the initial state $|q_0q_1q_2\rangle = |q_0\rangle \otimes |q_1\rangle \otimes |q_2\rangle$, to the state $|\psi\rangle = \mathcal{U} |q_0q_1q_2\rangle$. The operator \mathcal{U} is written as

$$|\psi\rangle = \mathcal{U} |q_0q_1q_2\rangle = (M \otimes \mathbb{I} \otimes E)(U \otimes \mathbb{I})(\mathbb{I} \otimes \mathbb{I} \otimes V) |q_0q_1q_2\rangle. \quad (2.49)$$

Then followed by a measurement of qubits 0 and 2, in the computational basis $\{|0\rangle, |1\rangle\}$, for which the measured values, 0 or 1, are stored in the classical bits a and b .

Note that, whereas a quantum gate normally represents a unitary operator, sometimes it will be convenient to write general quantum channels, acting on the qubit states, in the same quantum circuit notation, for example, in the case of a noise-afflicted gate that is not necessarily unitary anymore. When doing so, this will be clearly specified, and the following analysis will be done within the density operator formalism.

When quantifying the complexity of a quantum algorithm, a common metric is the circuit *depth*. This is defined as the longest path from initialisation to the circuit end, moving in time along the qubit wires. Each gate execution gives a contribution of one. This is roughly equivalent to the number of time steps needed for the circuit execution to terminate. As an example, take the circuit in Equation (2.48). Counting the measurement gates, this circuit has a depth of four.

2.4.4 Quantum Gates

We will now have a closer look at quantum gates and define some useful examples of quantum gates. A *quantum gate* is a unitary operation, acting on one or several qubits. We write the computational basis states of a qubit system as vectors in \mathbb{C}^2 .

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \quad (2.50)$$

By taking the tensor product of the single-qubit computational basis states, the computational basis state for an n -qubit system can analogously be written as unit vectors in \mathbb{C}^{2^n} . As an example, take an arbitrary state $|\psi\rangle$ on a 2-qubit system. We now write $|\psi\rangle$ as

$$|\psi\rangle = \sum_{z \in \{0,1\}^2} c_z |z\rangle = [c_{00} \ c_{10} \ c_{01} \ c_{11}]^T. \quad (2.51)$$

Any unitary operator on an n -qubit system will now map 2^n -dimensional complex vectors to 2^n -dimensional complex vectors and can thus be expressed as a $2^n \times 2^n$ matrix.

An example of an often-used quantum single-qubit gate is the Hadamard gate, defined as

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \boxed{H}. \quad (2.52)$$

Notably, the Hadamard acts on the single-qubit computational basis states as $H|0\rangle = |+\rangle$ and $H|1\rangle = |-\rangle$. The Hadamard thus gives a basis transformation, from the computational basis $\{|0\rangle, |1\rangle\}$, to the basis $\{|+\rangle, |-\rangle\}$ given by $|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$.

Furthermore, consider an n -qubit system starting out in the 0-state, i.e., in $|00\dots 0\rangle$. If we apply a single Hadamard gate to each of the n -qubits, we obtain an equal superposition of all the computational basis states, as

$$H^{\otimes n} |00\dots 0\rangle = \frac{1}{\sqrt{2^n}} \sum_{z \in \{0,1\}^n} |z\rangle. \quad (2.53)$$

Here, we have used $H^{\otimes n}$ to denote the tensor product $H \otimes \dots \otimes H$, of a total of n Hadamard gates.

The T -gate, and its inverse, the T^\dagger -gate, are two other useful quantum gates. These are defined as

$$T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} = \boxed{T}, \quad T^\dagger = \begin{bmatrix} 1 & 0 \\ 0 & e^{-i\pi/4} \end{bmatrix} = \boxed{T^\dagger}. \quad (2.54)$$

Furthermore, we define the phase gate S , and its inverse S^\dagger , as

$$S = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/2} \end{bmatrix} = \boxed{S}, \quad S^\dagger = \begin{bmatrix} 1 & 0 \\ 0 & e^{-i\pi/2} \end{bmatrix} = \boxed{S^\dagger}, \quad (2.55)$$

and the Pauli gates $X = \sigma_x$, $Z = \sigma_z$, and $Y = \sigma_y = iXZ$, which are defined as

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \boxed{X}, \quad (2.56)$$

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \boxed{Z}, \quad (2.57)$$

$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} = \boxed{Y}. \quad (2.58)$$

The Pauli gates further define the *rotation gates*, $R_x(\theta)$, $R_z(\theta)$, and $R_y(\theta)$, all parametrized by an angle θ , as

$$R_x(\theta) = e^{i\theta X/2} = \begin{bmatrix} \cos(\frac{\theta}{2}) & -i \sin(\frac{\theta}{2}) \\ -i \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{bmatrix} = \boxed{R_x(\theta)}, \quad (2.59)$$

$$R_z(\theta) = e^{i\theta Z/2} = \begin{bmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{bmatrix} = \boxed{R_z(\theta)}, \quad (2.60)$$

$$R_y(\theta) = e^{i\theta Y/2} = \begin{bmatrix} \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{bmatrix} = \boxed{R_y(\theta)}. \quad (2.61)$$

Representing an arbitrary single-qubit state in the Bloch sphere representation, given in Equation (2.27), these gates have the following interpretation: The gates $R_x(\theta)$, $R_z(\theta)$, and $R_y(\theta)$ act on $\rho = \frac{1}{2}(\mathbb{I} + \vec{r} \cdot \vec{\sigma})$ by rotating the Bloch vector by θ degrees around the x , z , or y axis, respectively.

In fact, any unitary mapping between single-qubit pure quantum states can be decomposed into a sequence of the three rotation operators, possibly

multiplied by a global phase factor. This follows from the rotation operators giving three independent rotations of a three-dimensional unit vector, i.e., the Bloch vector. Thus, any arbitrary unitary operator U can be written as

$$U = e^{i\alpha} R_x(\theta) R_z(\phi) R_y(\lambda). \quad (2.62)$$

An important set of single-qubit gates, sometimes referred to as the *physical gates*, are the U_1 , U_2 , and U_3 gates. These are the physically realized single-qubit gates in the quantum computers publicly available, at present time, through the IBM Quantum Experience [27]. The $U_3(\theta, \phi, \lambda)$ gate, parameterized by the angles θ , ϕ , and λ , is given as

$$U_3(\theta, \phi, \lambda) = \begin{bmatrix} \cos(\theta/2) & -e^{i\lambda} \sin(\theta/2) \\ e^{i\phi} \sin(\theta/2) & e^{i\lambda+i\phi} \cos(\theta/2) \end{bmatrix} = \boxed{U_3(\theta, \phi, \lambda)}. \quad (2.63)$$

With three independent angles, the U_3 gate can generate any arbitrary rotation of the Bloch vector, and thus any arbitrary single-qubit unitary. The U_1 and U_2 gates are derived from the U_3 gate, as $U_1(\lambda) = U_3(0, 0, \lambda)$ and $U_2(\phi, \lambda) = U_3(\frac{\pi}{2}, \phi, \lambda)$.

2.4.5 Controlled Gates

An important class of multi-qubit gates are the *controlled* gates. Focusing on the two-qubit case, a controlled gate acts on one qubit called the *control*, and one qubit called the *target*. In general, a controlled gate may act on multiple control and target qubits. In circuit notation, a two-qubit *controlled-U* gate, denoted cU , is defined as

$$cU = \begin{array}{c} \bullet \\ | \\ \boxed{U} \end{array}, \quad (2.64)$$

for U being a unitary operator. The black dot denotes the control, and the other qubit is the target. The action of a controlled-U gate on a state $|x\rangle|y\rangle$, where $|x\rangle$ is the control and $|y\rangle$ is the target, is written as a quantum operator as

$$cU = |0\rangle\langle 0| \otimes \mathbb{I} + |1\rangle\langle 1| \otimes U. \quad (2.65)$$

A controlled gate of high practical importance is the *controlled-NOT* gate, or *controlled-X* gate, denoted CNOT or cX . The CNOT-gate is in circuit notation denoted as

$$\text{CNOT} = \begin{array}{c} \bullet \\ | \\ \oplus \end{array}. \quad (2.66)$$

Note the notation on the lower qubit, the target qubit. Writing out the quantum operator, and representing the two-qubit state as a vector in \mathbb{C}^4 , as given in Equation (2.51), the CNOT-gate can be expressed as a 4×4 matrix as

$$\text{CNOT} = |0\rangle\langle 0| \otimes \mathbb{I} + |1\rangle\langle 1| \otimes X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (2.67)$$

The reason for the *controlled-NOT* term becomes apparent when inspecting the action of the CNOT on computational basis states $|x\rangle, |y\rangle \in \{|0\rangle, |1\rangle\}$. We get

$$\text{CNOT } |x\rangle |y\rangle = |x\rangle |y \oplus x\rangle, \quad (2.68)$$

where \oplus is addition modulo 2. For $|x\rangle, |y\rangle$ computational basis states, the CNOT acts as a logical NOT gate on y if $x = 1$, and as the identity gate otherwise.

2.4.6 Gate Identities

The Pauli-gates interact with the CNOT-gates to give effective, total operators according to

$$\text{CNOT } (X \otimes \mathbb{I}) \text{CNOT} = X \otimes X, \quad \text{CNOT } (\mathbb{I} \otimes X) \text{CNOT} = \mathbb{I} \otimes X, \quad (2.69)$$

and

$$\text{CNOT } (Z \otimes \mathbb{I}) \text{CNOT} = Z \otimes \mathbb{I}, \quad \text{CNOT } (\mathbb{I} \otimes Z) \text{CNOT} = Z \otimes Z, \quad (2.70)$$

where the operators act on 2-qubit states where the first qubit is the control, and the second is the target. In terms of quantum circuits, the Pauli- X and Z gates are said to be "propagated" over a CNOT-gate, according to the following identities

$$\begin{array}{c} \boxed{X} \text{---} \bullet \text{---} \\ | \\ \oplus \text{---} \end{array} = \begin{array}{c} \bullet \text{---} \boxed{X} \text{---} \\ | \\ \oplus \text{---} \boxed{X} \text{---} \end{array}, \quad (2.71)$$

$$\begin{array}{c} \bullet \text{---} \\ | \\ \oplus \text{---} \boxed{X} \text{---} \end{array} = \begin{array}{c} \bullet \text{---} \\ | \\ \oplus \text{---} \boxed{X} \text{---} \end{array}, \quad (2.72)$$

$$\begin{array}{c} \boxed{Z} \text{---} \bullet \text{---} \\ | \\ \oplus \text{---} \end{array} = \begin{array}{c} \bullet \text{---} \boxed{Z} \text{---} \\ | \\ \oplus \text{---} \end{array}, \quad (2.73)$$

$$\begin{array}{c} \bullet \text{---} \\ | \\ \oplus \text{---} \boxed{Z} \text{---} \end{array} = \begin{array}{c} \bullet \text{---} \boxed{Z} \text{---} \\ | \\ \oplus \text{---} \boxed{Z} \text{---} \end{array}. \quad (2.74)$$

In the context of quantum errors, this means that a single Pauli- X error on the control qubit will be propagated into a *double* Pauli- X error, one on each qubit, by a CNOT-gates. A single Pauli- X error on the target is left alone. And similarly, a single Pauli- Z error on the target propagates into two errors, whereas a Pauli- Z error on the control stays as a single error. This has implications for the construction of *fault-tolerant* quantum circuits.

The Pauli- Y gate identities can be found from the above, by using $Y = iXZ$. The factor i becomes a global phase factor that can be ignored.

A CNOT gate with opposite control and target qubits can be constructed by applying single-qubit Hadamard gates, according to

$$\begin{array}{c}
 \text{---} [H] \text{---} \bullet \text{---} [H] \text{---} \oplus \text{---} \\
 \text{---} [H] \text{---} \oplus \text{---} [H] \text{---} \bullet \text{---}
 \end{array} = \begin{array}{c}
 \oplus \\
 \bullet
 \end{array} . \quad (2.75)$$

Furthermore, the other controlled-Pauli gates, cZ and cY , can be constructed from the $cX = \text{CNOT}$ gate and single-qubit gates, as

$$\begin{array}{c}
 \text{---} \bullet \text{---} \\
 \text{---} [H] \text{---} \oplus \text{---} [H] \text{---}
 \end{array} = \begin{array}{c}
 \bullet \\
 \text{---} [Z] \text{---}
 \end{array} , \quad (2.76)$$

$$\begin{array}{c}
 \text{---} \bullet \text{---} \\
 \text{---} [S^\dagger] \text{---} \oplus \text{---} [S] \text{---}
 \end{array} = \begin{array}{c}
 \bullet \\
 \text{---} [Y] \text{---}
 \end{array} . \quad (2.77)$$

A useful composite gate is the TOFFOLI-gate, which takes part in several quantum algorithms. The gate acts on two control qubits and one target qubit, and can be defined by its action on computational basis states $|x\rangle, |y\rangle, |z\rangle \in \{|0\rangle, |1\rangle\}$ as

$$\begin{array}{c}
 |x\rangle \text{---} \bullet \text{---} |x\rangle \\
 |y\rangle \text{---} \bullet \text{---} |y\rangle \\
 |z\rangle \text{---} \oplus \text{---} |z \oplus (x \cdot y)\rangle
 \end{array} \quad (2.78)$$

where \oplus means additional modulo 2. The gate can be constructed from H , T and T^\dagger single-qubit gates, and a total of 6 CNOT-gates, as

$$\begin{array}{c}
 \text{---} \bullet \text{---} \\
 \text{---} \bullet \text{---} \\
 \text{---} \oplus \text{---}
 \end{array} = \begin{array}{c}
 \text{---} \bullet \text{---} [T] \text{---} \bullet \text{---} \\
 \text{---} \bullet \text{---} [T] \text{---} \oplus \text{---} [T^\dagger] \text{---} \oplus \text{---} \\
 \text{---} [H] \text{---} \oplus \text{---} [T^\dagger] \text{---} \oplus \text{---} [T] \text{---} \oplus \text{---} [T^\dagger] \text{---} \oplus \text{---} [T] \text{---} [H] \text{---}
 \end{array} . \quad (2.79)$$

2.4.7 Basis Sets of Gates

In practice, we will not be able to perform general unitary operations directly on a quantum computer. Instead, a quantum computer will have a set of basic quantum gates that are physically realized on the quantum hardware and can thus be directly applied to qubits. These gates form the *native gate set* of the quantum computer.

A gate set is said to be *universal* for quantum computation if any arbitrary n -qubit unitary \mathcal{U} can be approximated to arbitrary precision by exclusively using the gates from said gate set.

The native gate set should, however, be chosen carefully. Of note is the Gottesman–Knill theorem that tells us that quantum circuits built from the gate set

$$\{\text{CNOT}, H, S\} \quad (2.80)$$

can be efficiently simulated on a classical computer [28]. In some sense we do not capture anything *super-classical*, so to say, by using this gate set.

There exists alternative gate sets that are both universal and that can not be simulated efficiently in any known way by a classical computer. One such gate set is given by

$$\{\text{CNOT}, H, T\}. \quad (2.81)$$

It is possible to implement arbitrary single-qubit gates, given by the U_3 gate in Equation (2.63), on physical quantum computers. Therefore, it is convenient to consider the native gate set

$$\{\text{CNOT}, U_3\}. \quad (2.82)$$

As the H and T gates can be constructed from the U_3 gate this gate set is indeed universal. From this gate set, however, we gain the advantage that all single-qubit gates can be directly implemented as U_3 -gates, instead of decomposing them into successive H and T gates.

2.4.8 Qubit Connectivity and Virtual Qubits

A concept of practical importance for quantum computation is *qubit connectivity*; the layout of qubits for which multi-qubit gates are physically realized. We focus on the CNOT-gate. As explained in Section 2.4.7, the two-qubit CNOT-gate together with the single-qubit U -gate form an universal set of gates for quantum computing.

An example of a *connectivity map* is presented in Figure 2.2. This map shows which physical qubits are connected by physically realized CNOT-gates. Note that while CNOT-gates in principle are directional, a CNOT-gate may be flipped using only single-qubit gate by Equation (2.75).

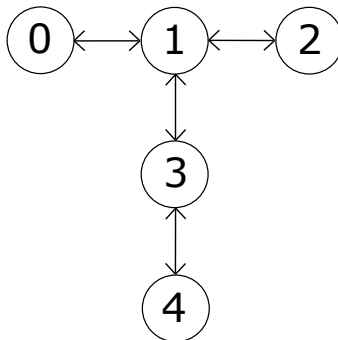
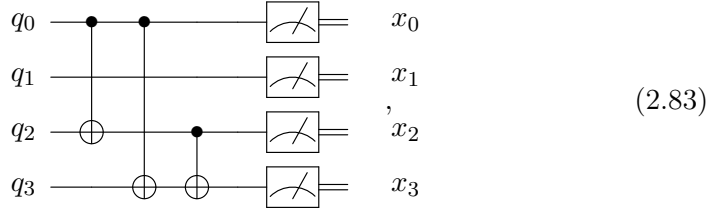


Figure 2.2: Connectivity map for the IBMQ Vigo quantum device. Nodes denote physical qubits and arrows denote which qubits are connected by physically implemented CNOT-gates [27].

The specific connectivity of a quantum computer may leave certain quantum circuits incompatible. To illustrate these implications of limited qubit con-

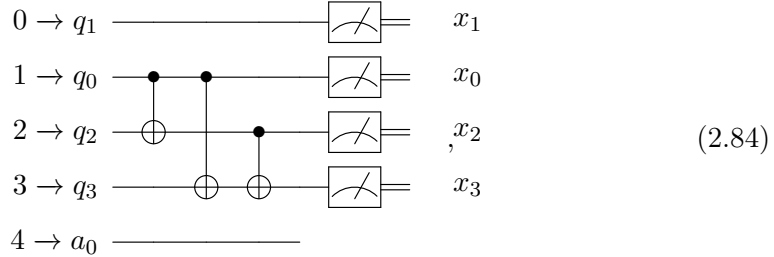
nectivity, and how we may deal with them, we take an example quantum circuit given as



with qubits $q_i, i = 1, 2, 3, 4$, initialized to some initial state. The meters indicate measurements in the computational basis, where the measurement result from qubit q_i is stored in the classical bit x_i .

Assume that we want to implement the circuit given by Equation (2.83) on a quantum computer with a connectivity map as given in Figure 2.2. Assuming that the qubit q_i as denoted in the quantum circuit diagram corresponds to the physical qubit i as denoted in the connectivity map, this circuit seems not to be compatible with the given quantum backend. For example, the first CNOT-gate between qubits 0 and 2 is not possible because there is no physical CNOT-gate realized between these two qubits.

We now want to transform this quantum circuit into an equivalent circuit that is compatible with the specified connectivity map. First, we introduce the concept of *virtual qubits*. We are only interested in the qubit states, and it is irrelevant on which physical qubits are located on. In our quantum circuit we have 4 virtual qubits q_0, q_1, q_2, q_3 that we need to assign, or *map* to the 5 physical qubits denoted by $i = 0, 1, 2, 3, 4$. By reassigning virtual qubits q_0 and q_1 to physical qubits 1 and 0, respectively, we obtain the equivalent quantum circuit



where the left-hand side denotes the assignment of physical qubits to virtual qubits. Now, the two first CNOT-gates from Equation (2.83), between q_0 and q_2 , and between q_0 and q_3 , are possible as there is qubits connectivity between their assigned physical qubits. Note that we assign the leftover physical qubit, qubit 4, to the virtual *ancillary qubit* a_0 . In some cases the inclusion of ancillary qubits may become useful, yet in this example it will not.

We are not done, however, as the CNOT-gate between virtual qubits q_2 and q_3 is still not possible. In fact, this may not be solved by a simple rearrangement at circuit start because we have three virtual qubits, q_0, q_2 and q_3 , that all require pairwise connectivity. No arrangement on the connectivity map shown in Figure 2.2 allows this.

To solve this problem, we introduce the SWAP-gate. The SWAP-gate on two single-qubit state $|\psi\rangle$ and $|\phi\rangle$ is defined as

$$\text{SWAP } |\psi\rangle |\phi\rangle = |\phi\rangle |\psi\rangle, \tag{2.85}$$

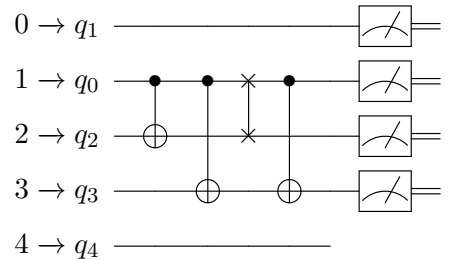
i.e., the SWAP-gate swaps which physical qubits the two states are located on. The SWAP-gate can be implemented by three CNOT-gates as



$$\tag{2.86}$$

In terms of virtual qubits, a SWAP-gate between two physical qubits 1 and 2, assigned to virtual qubits q_1 and q_2 , respectively, the gates swap the locations of these two virtual qubits.

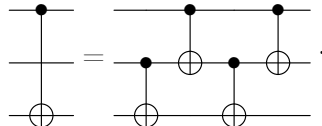
For our example circuit, the third CNOT-gate may now be realized by moving one of the virtual qubits. From Equation (2.84) we observe that the CNOT-gate between q_2 and q_3 , located physical qubits 2 and 3, is not possible. By relocating virtual qubit q_2 to physical qubit 1 by use of a swap gate, this is solved because there is connectivity between physical qubits 1 and 3 as seen from Figure 2.2. This leaves us with the end circuit given by



$$\tag{2.87}$$

that is now indeed fully compatible with the given qubit connectivity. Note how the measurement into classical bit x_0 is now performed on qubit 2, even though the virtual qubit q_0 was initially located on qubit 1. This is because the SWAP-gate has relocated the quantum state associated with virtual qubit q_0 from physical qubit 1 to qubit 2, i.e., has relocated the virtual qubit q_0 to qubit 2.

Another gate that is useful for overcoming connectivity incompatibilities is the *BRIDGE-gate* [29]. A BRIDGE-gate allows a *virtual CNOT-gate* between two virtual qubits located on non-connected physical qubits, as long as there exists a third physical qubit connected to each one of them.



$$\tag{2.88}$$

Considering the example circuit in the state as shown in Equation (2.84), a BRIDGE-gate could have been used to implement the CNOT-gate between virtual qubits q_2 and q_3 . As they were situated on physical qubits 2 and 3, qubit 1 could have been used as the intermediary qubit for the BRIDGE-gate.

Note that the BRIDGE-gate would not interfere with the virtual qubit, q_0 , situated on qubit 1, at least not in the noise-less case.

It is important to note that both SWAP-gates and BRIDGE-gates introduce additional multi-qubit gates to a quantum circuit. In practice, multi-qubit gates such as the CNOT-gate tends to be encumbered with significant noise. Furthermore, they typically have an error rate at least one order of magnitude larger than single-qubit gates [30]. This may impact the accuracy of our computations. The topic of noise in quantum computers is discussed in more detailed in Chapter 3, but for now we note that to obtain the highest possible accuracy we need to minimize the number of noisy gates, especially the noisier multi-qubit gates.

2.4.9 Measurements of Quantum Circuits

In a quantum computation, we initialize some state $|\psi_0\rangle$ and evolve it into an end state $|\psi\rangle$. But we can not observe the state $|\psi\rangle$ directly. To extract any information about $|\psi\rangle$ we have to *measure* it, according to the third postulate of quantum mechanics as presented in Section 2.2.3.

A measurement in a quantum circuit is treated as a special type of gate, as denoted in Equation (2.47). This gate without any further specifications denote a measurement in the computational basis, i.e., the $\{|0\rangle, |1\rangle\}$ basis. The measurement result, either 0 or 1, may then be stored in a classical bit.

For any orthonormal single-qubit basis $\{|\xi_0\rangle, |\xi_1\rangle\}$ we can find an unitary basis change operator B such that

$$B|\xi_0\rangle = |0\rangle, B|\xi_1\rangle = |1\rangle, \quad (2.89)$$

thus, measurements in the computational basis are sufficient for quantum computation.

A common type of measurements are the *Pauli measurements*, i.e., measurements in the eigenbases of the Pauli-operators. The eigenbasis of the Pauli- Z operator is just the computational basis, where the eigenstate $|0\rangle$ and $|1\rangle$ have eigenvalues $+1$ and -1 . The basis change operator between the computational basis and the Pauli- X and Pauli- Y operators are H and HS^\dagger , respectively. Thus, an X -type measurement can be implemented as

$$\text{---} \boxed{H} \text{---} \boxed{\text{meter}} = x = \text{---} \boxed{X} \text{---} \boxed{\text{meter}} = x, \quad (2.90)$$

where the meter gate without any additional specifications imply a measurement in the computational basis. Similarly, a Y -type measurement can be performed as

$$\text{---} \boxed{S^\dagger} \text{---} \boxed{H} \text{---} \boxed{\text{meter}} = x = \text{---} \boxed{Y} \text{---} \boxed{\text{meter}} = x. \quad (2.91)$$

Note that the bit value 0 or 1 of x here corresponds to the eigenvalues $+1$ or -1 , respectively.

Non-destructive measurements of qubits can be done by use of ancillary qubits. Take the circuit

$$\begin{array}{ccc}
 |\psi\rangle & \text{---} & \bullet & \text{---} & |\psi'\rangle \\
 & & | & & \\
 |0\rangle & \text{---} & \oplus & \text{---} & \boxed{\text{meter}} & \text{---} & x.
 \end{array} \tag{2.92}$$

For a general qubit state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ on the input, the full state of the two-qubit system after the CNOT gate will be

$$\text{CNOT } |\psi\rangle |0\rangle = \text{CNOT } (\alpha|0\rangle + \beta|1\rangle) |0\rangle = \alpha|00\rangle + \beta|11\rangle. \tag{2.93}$$

A measurement of the ancillary qubit in the computational basis will thus yield an outcome of $x = 0$ with probability $|\alpha|^2$, and $x = 1$ with probability $|\beta|^2$. The measurement is still projective, such that the upper qubit is left in a post-measurement state depending on the outcome. In this case will be $|\psi'\rangle = |x\rangle$, where $x \in \{0, 1\}$ is the bit value of the measurement outcome.

As previously discussed, quantum mechanics is probabilistic. Therefore, we will need to repeat the experiment of state preparation proceeded by measurements several times to gain sufficient information about the prepared state $|\psi\rangle$. Each individual measurement result may be represented by a n -bit bit-string $x = x_0x_1 \dots x_{n-1}$, where each bit $x_i \in \{0, 1\}$ represents the measurement result on qubit i . After repeating the experiment for a given number of *shots*, we can assign a count to each bit-string x , corresponding to the number of times we recorded the measurement result x . These counts are the *output* of the quantum part of quantum algorithm and are then interpreted into useful results by use of classical post-processing.

2.4.10 Expectation Values

In a quantum computation, we are often interested in measuring the *expectation value* of some operator with respect to a quantum state. In that case, the value we are interested in computing is the value of

$$\langle A \rangle_{|\psi\rangle} = \langle \psi | A | \psi \rangle, \tag{2.94}$$

where A is a quantum operator and $|\psi\rangle$ is a quantum state prepared by a quantum circuit.

Assume that we want to measure the expectation value of an operator A with respect to some state $|\psi\rangle = \mathcal{U}|00\dots 0\rangle$. The unitary operator \mathcal{U} is implemented as a quantum circuit. The state $|\psi\rangle$ can be expressed as

$$|\psi\rangle = \sum_i c_i |\phi_i\rangle \tag{2.95}$$

for $\{\phi_i\}_i$ a set of orthonormal eigenstates of A , such that $A|\phi_i\rangle = a_i|\phi_i\rangle$ for a $a_i \in \mathbb{R}$.

Measurements in other bases than the computational basis can be performed if we find a basis transformation by a unitary operator V such that

$$V\phi_i = |z_i\rangle, \quad (2.96)$$

where z_i are the unique bit-string representations of the integers $i = 0, \dots, 2^n - 1$. When measuring in the computational basis, after the basis change, the measurement outcomes will be the bit string z_i with probability $|c_i|^2$.

To estimate the expectation value of A with respect to $|\psi\rangle$, as each outcome is probabilistic, we repeat the experiment of state preparation and measurement several times. Then, from the set of measurement outcomes, we compute the average observed value.

As an example, assume we prepare the two-qubit state

$$|\psi\rangle = a_{00}|0\rangle \otimes |0\rangle + a_{01}|0\rangle \otimes |1\rangle + a_{10}|1\rangle \otimes |0\rangle + a_{11}|1\rangle \otimes |1\rangle, \quad (2.97)$$

by a quantum circuit. Say we want to measure its expectation value with respect to the operator $Z \otimes Z$. The eigenvalues and eigenstates are

$$Z|0\rangle = +1|0\rangle, \quad Z|1\rangle = -1|1\rangle. \quad (2.98)$$

We get the eigenvalue $+1$ for eigenstates $|0\rangle \otimes |0\rangle$ and $|1\rangle \otimes |1\rangle$, and -1 for the eigenstates $|0\rangle \otimes |1\rangle$ and $|1\rangle \otimes |0\rangle$. The expectation value $E^* = \langle Z \rangle_{|\psi\rangle}$ of a measurement of this operator, with respect to the state $|\psi\rangle$, is thus

$$\begin{aligned} E^* &= (+1)(p_{00} + p_{11}) + (-1)(p_{01} + p_{10}) \\ &= (+1)(|a_{00}|^2 + |a_{11}|^2) + (-1)(|a_{01}|^2 + |a_{10}|^2). \end{aligned} \quad (2.99)$$

Here, $p_{00} = |a_{00}|^2$ is the probability of measuring the outcome 00, and analogously for the other eigenstates.

Let $n_{z_0 z_1}$ be the number of times the measurement outcome $z = z_0 z_1 \in \{0, 1\}^2$, corresponding to the state $|z_0\rangle \otimes |z_1\rangle$ for $z_0, z_1 \in \{0, 1\}$, and $N = \sum_{z_0, z_1} n_{z_0 z_1}$ be the total number of experiments. Then, $n_{z_0 z_1}/N$ serves as an estimate for $p_{z_0 z_1} = |a_{z_0 z_1}|^2$. Furthermore, the expectation value E^* is then estimated by

$$E = \frac{1}{N}((n_{00} + n_{11}) - (n_{01} + n_{10})). \quad (2.100)$$

2.4.11 Mean Square Error and Error Estimation

As presented in Section 2.4.10, quantum computations often involve the estimation of an expectation value by taking an average over a large number of circuit executions. The error in this estimation is determined by the number of circuit executions, or number of *shots*, and the variance in the measurement outcomes. Given a fixed error threshold, the variance thus determines the required number of circuit shots and thus have implications for the running time of a quantum algorithm.

Assume that N shots of a circuit, measuring some observable, are performed, each time with a measurement outcome of μ_j , $j = 1, \dots, N$. Let E^* be the expectation value and σ^2 the variance of the measurement outcomes, which are assumed to be independently and equally distributed. An estimator for the expectation value is then

$$E = \frac{1}{N} \sum_j \mu_j, \quad (2.101)$$

and an estimator for the variance σ^2 in each measurement is the *mean square error* s^2 , given as

$$s^2 = \frac{1}{N-1} \sum_j (\mu_j - E)^2 = \frac{1}{N} \sum_j \mu_j^2 - \left(\frac{1}{N} \sum_j \mu_j\right)^2. \quad (2.102)$$

By the central limit theorem, an estimator for the error ϵ in E is

$$\epsilon = \frac{\sigma}{\sqrt{N}}, \quad (2.103)$$

where E is the average over N measurement outcomes, i.e., with the circuit executed for N shots.

The variance in a quantum measurement is in general hard to know without knowing the exact state $|\psi\rangle$ of the system at the time of measurements. Therefore, we do not in general know how many shots is necessary to estimate E^* to within some tolerance error ϵ_{tol} .

A solution to this problem is instead to *sample* the circuit for some initial number of shots N , obtaining measurement outcomes μ_j . Then, we estimate the variance σ^2 by the mean-square error s^2 , as shown in Equation (2.103). The required amount of shots N' to achieve an error below the threshold is then estimated to be

$$N' = \left(\frac{s}{\epsilon_{\text{tol}}}\right)^2. \quad (2.104)$$

If $N' > N$, we repeat the experiment for an additional number $N' - N$ of shots. The average over all measurement outcomes now gives an estimator of the expectation value, with an estimated error $\epsilon \leq \epsilon_{\text{tol}}$.

Chapter 3

Error Models for Quantum Computation

Near-term quantum computers are noisy. To obtain accurate and useful results from quantum computations, combating this noise is imperative, and in order to do so we first need to model the noise present in NISQ-era quantum devices. This chapter presents noise in quantum computers formulated in terms of quantum channels, as detailed in Section 2.3.4. We further give a few examples of particular useful noise channels, including the depolarizing channel, the amplitude damping channel and the phase damping channel.

3.1 Noise as Quantum Channels

An ideal quantum gate applies a unitary operator U . Real quantum gates, however, are noisy. A noisy quantum gate may potentially no longer be unitary, e.g., if we have some interaction between the qubits and the environment. This is where the density operator and quantum channel formalism come in handy. The ideal quantum gate can be described by the unitary quantum channel

$$\mathcal{E}(\rho) = U\rho U^\dagger, \quad (3.1)$$

acting on a density operator ρ . In the noisy case, we may still describe the action of the noisy quantum gate as a quantum channel, but it might no longer preserve purity of density operators.

Assume that we want to apply a quantum gate described by the quantum channel \mathcal{E} , but because of noise, the channel that is applied is instead $\tilde{\mathcal{N}}$. By writing

$$\tilde{\mathcal{N}} = \tilde{\mathcal{N}} \circ \mathcal{E}^{-1} \circ \mathcal{E} = \mathcal{N} \circ \mathcal{E}, \quad (3.2)$$

where we have defined $\mathcal{N} = \tilde{\mathcal{N}} \circ \mathcal{E}^{-1}$, we can regard a faulty quantum channel or gate as the ideal channel \mathcal{E} followed by the effective noise channel \mathcal{N} . The operator-sum representation, as detailed in Section 2.3.5, now gives a compact way to represent noise channels in terms of their Kraus operators.

Similarly, we can describe a noisy initialization of qubits to the faulty initial state ρ'_0 as the ideal initialization to the ideal initial state ρ_0 followed by a noise channel such that $\rho'_0 = \mathcal{N}_{\text{init}}(\rho_0)$. Noise in measurements can be described as an ideal measurement preceded by a noise channel $\mathcal{N}_{\text{meas}}$.

3.2 Coherent Noise

Coherent noise is noise that can be expressed as a unitary error operator \tilde{U} . The quantum operation acting on density operators then becomes a unitary operation $\mathcal{N}(\rho) = \tilde{U}\rho\tilde{U}^\dagger$. By Equation (2.31), such noise preserves the purity of a pure input state $\rho = |\psi\rangle\langle\psi|$.

An example is an error in a quantum single-qubit gate that causes a perturbation in the rotation of the Bloch sphere. Any single qubit gate can be expressed as a rotation of the Bloch vector as in Equation (2.62). Say one wants to perform a rotation about the z -axis by an angle ϕ , i.e., $U = R_z(\phi)$, but due to an error the applied operator is instead $U' = R_z(\phi + \delta)$ for some small angle δ . This can be described by the ideal rotation followed by the perturbation $R_z(\delta)$ as $U' = R_z(\delta)R_z(\phi) = R_z(\delta)U$. The corresponding quantum channel becomes $\mathcal{N}(\rho) = U'\rho U'^\dagger$, preserving purity.

3.3 Incoherent Noise

Incoherent noise is noise that does not necessarily preserve purity, for example noise stemming from an interaction with the environment. As described by the Stinespring dilation and Equation (2.32), a quantum operation unitary on a larger system, for example the qubit in interaction with the environment, can still be described locally on the qubit as a quantum channel.

3.3.1 Stochastic Pauli-Noise

A simple example of an incoherent noise channel is the stochastic Pauli channel. Consider the Pauli operators X , Z , and Y . The Pauli- X operator can be thought of as the equivalent to a classical *bit-flip* error, as it "flips" a $|0\rangle$ -state to a $|1\rangle$ -state, and vice versa. Similarly, a Pauli- Z operator may be thought of as a *phase-flip* error, and a Pauli- Y as a bit-flip error followed by a phase-flip.

We assume that each of the Pauli-operators are independently and probabilistically applied to a single qubit with probabilities p_x , p_z , and p_y . This case is described by the Kraus operators

$$K_0 = \sqrt{1 - p_x - p_y - p_z}\mathbb{I}, \quad K_1 = \sqrt{p_x}X, \quad K_2 = \sqrt{p_y}Y, \quad K_3 = \sqrt{p_z}Z. \quad (3.3)$$

Then, the condition of Equation (2.35) is fulfilled by these Kraus operators, as

$$\sum_i K_i^\dagger K_i = (1 - p_x - p_y - p_z)\mathbb{I} + p_x XX + p_y YY + p_z ZZ = \mathbb{I}, \quad (3.4)$$

where we use that the Pauli gates are their own adjoints, and their own inverses.

3.3.2 Depolarizing Noise

A noise channel closely related to the stochastic Pauli-channel is the *depolarizing noise channel*. The depolarizing noise channel on a d -dimensional density operator ρ is defined as

$$\mathcal{N}_{p'}(\rho) = (1 - p')\rho + \frac{p'}{d}\mathbb{I}. \quad (3.5)$$

This channel in $d = 2$ dimensions may be transformed into the equivalent noise channel

$$\mathcal{N}_p(\rho) = (1 - p)\rho + \frac{p}{3} \sum_{i=1}^3 \sigma^i \rho \sigma^i, \quad (3.6)$$

where σ^i are the Pauli-operators and $p = (1 - \frac{3p'}{4})$. The probability p is said to be the *error rate* of this channel. The Kraus operators can now be identified as the set

$$K_0 = \sqrt{1 - p}\mathbb{I}, K_1 = \frac{p}{\sqrt{3}}X, K_2 = \frac{p}{\sqrt{3}}Z, K_3 = \frac{p}{\sqrt{3}}Y. \quad (3.7)$$

The depolarizing noise channel is thus equivalent to the homogeneous Pauli-noise channel, i.e., the stochastic Pauli-noise channel with $p_x = p_y = p_z = \frac{1}{3}$.

The depolarizing noise channel on higher dimensions can be transformed equivalently. As a model for noise on two-qubit operators, e.g., the CNOT-gate, we get the depolarizing noise channel on $d = 4$ dimensions as

$$\mathcal{N}_p(\rho) = (1 - p)\rho + \sum_{i,j} \frac{p}{15} O_1^{(i)} O_2^{(j)} \rho O_2^{(j)} O_1^{(i)}, \quad (3.8)$$

where $O_k^{(i)}$ is the operator $O^{(i)}$ acting on qubit k . The sum goes over the Pauli operators including the identity operator, e.g., $O^{(i)} \in \{\mathbb{I}, X, Y, Z\}$. However, the 2-qubit identity operator term $O_1^{(i)} O_2^{(j)} = \mathbb{I}_1 \mathbb{I}_2$ is omitted from the sum.

3.3.3 Amplitude Damping

Let us assume that we have a qubit system in which the $|0\rangle$ state is encoded as the ground-state of the system, and $|1\rangle$ as an excited state. A qubit in the $|1\rangle$ state may then fall to the $|0\rangle$ state by the process of spontaneous emission, but not the other way around.

This is described by the *amplitude damping* channel. Assume that the excited $|1\rangle$ state may collapse down to a $|0\rangle$ state with probability p . This case is described by the Kraus operators

$$K_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-p} \end{bmatrix}, K_1 = \begin{bmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{bmatrix}. \quad (3.9)$$

These Kraus operators give exactly the described case, where a $|0\rangle$ -state is kept in the $|0\rangle$ state as

$$\mathcal{N}(|0\rangle\langle 0|) = K_0 |0\rangle\langle 0| K_0^\dagger + K_1 |0\rangle\langle 0| K_1^\dagger = |0\rangle\langle 0|, \quad (3.10)$$

whereas a $|1\rangle$ -state is transformed to a probabilistic mixture of the state collapsing into the $|0\rangle$ -state with probability p , or no collapse with probability $1 - p$, as

$$\mathcal{N}(|1\rangle\langle 1|) = K_0 |1\rangle\langle 1| K_0^\dagger + K_1 |1\rangle\langle 1| K_1^\dagger = p |0\rangle\langle 0| + (1 - p) |1\rangle\langle 1|. \quad (3.11)$$

The action on a general qubit state, $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$, further follows from linearity.

3.3.4 Phase Damping

A uniquely quantum-mechanical noise process is the process of *phase damping*, describing loss of information in the partial phases of the eigenstates. Take the case in which the qubit state on system A does not transition between states but couples to the environment in the following way,

$$\mathcal{N}(|0\rangle_A \otimes |0\rangle_E) = \sqrt{1-p} |0\rangle_A \otimes |0\rangle_E + \sqrt{p} |0\rangle_A \otimes |1\rangle_E, \quad (3.12)$$

and

$$\mathcal{N}(|1\rangle_A \otimes |0\rangle_E) = \sqrt{1-p} |1\rangle_A \otimes |0\rangle_E + \sqrt{p} |1\rangle_A \otimes |2\rangle_E, \quad (3.13)$$

where $\{|i\rangle_E\}_i$ form some basis for the environment. An interpretation could be that the environment in some way would scatter off of the qubit with probability p , ending up in a state dependent on the qubit state.

By tracing out the environment, we obtain a quantum channel with Kraus operators

$$K_0 = \begin{bmatrix} \sqrt{1-p} & 0 \\ 0 & \sqrt{1-p} \end{bmatrix}, \quad K_1 = \begin{bmatrix} \sqrt{p} & 0 \\ 0 & 0 \end{bmatrix}, \quad K_2 = \begin{bmatrix} 0 & 0 \\ 0 & \sqrt{p} \end{bmatrix}. \quad (3.14)$$

The effect of this channel on a general density operator

$$\rho = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (3.15)$$

becomes

$$\mathcal{N}(\rho) = \begin{bmatrix} a & (1-p)b \\ (1-p)c & d \end{bmatrix}. \quad (3.16)$$

Thus, for repeated applications of this channel, the off-diagonal terms b and c , the relative phases, are suppressed.

Chapter 4

Hybrid Algorithms

This chapter presents the concept of hybrid quantum-classical algorithms, which are hoped to be applicable on near-future quantum hardware. To understand the limitations put on near-future quantum algorithms, we have to consider the two main limiting factors for algorithms on near-term quantum devices. Near-term quantum computers are noisy, which limits the depth of quantum circuits that may be executed within reasonable precision. They are also limited in their number of available, connected qubits.

In the near-future, it is considered realistic that we may obtain from a few hundred to thousands of qubits on quantum devices, without prohibitively large error rates. As an example of a long-term quantum computing application, let us take Shor's factorization algorithm [31]. An estimate for the number of noisy qubits needed to perform Shor's algorithm on 2048-bit integers, which would break modern RSA-encryption schemes, has been found to be over 20 million qubits [9]. The Shor's algorithm uses long-depth quantum circuits, for which quantum error correction codes must be used to obtain reasonable results, thus the large qubit requirements.

Our attention for possible near-future applications therefore turns to *hybrid quantum-classical algorithms*, commonly referred to as *hybrid algorithms*. These are algorithms that combine classical and quantum computation, where the quantum component involves short-depth quantum circuits. The umbrella term of hybrid algorithms includes algorithms such as the variational quantum eigensolver (VQE), the quantum approximate optimization algorithm (QAOA), the variational quantum linear solver, amongst others [3, 10, 12]. In this thesis we will focus solely on the VQE algorithm.

Many of the hybrid algorithms that have been proposed in recent years, such as the QAOA and the VQE, are examples of *variational quantum methods*. Section 4.1 explores the subject of variational quantum methods in more general terms. The VQE algorithm is then presented in Section 4.2, including three different parametrized state ansätze.

4.1 Variational Quantum Methods

A class of useful hybrid algorithms are *variational quantum methods* that approximate ground states of quantum operators by use of the variational principle. The idea is to prepare a parameterized trial state, often referred to as an *ansatz*. Then we try to find the parameters for which the expectation value of the ansatz with respect to the operator is minimal.

Assume that we have a quantum operator \mathcal{H} with real eigenvalues $\{E_i\}_i$ and eigenvectors $\{|\psi_i\rangle\}_i$. Order the eigenvalues such that $E_0 \leq E_1 \leq \dots \leq E_n$. The ground state of \mathcal{H} is the eigenstate $|\psi_0\rangle$ corresponding to the minimal eigenvalue E_0 . If \mathcal{H} is the Hamiltonian of some physical system then we refer to E_0 as the *ground state energy*. If more than one eigenvalue is equal to E_0 , we say that the ground state is *degenerate*.

We want to prepare a quantum state $|\psi(\vec{\theta})\rangle$, parametrized by a set of variational parameters $\vec{\theta}$. In quantum computation we prepare this state by a unitary operator $U(\vec{\theta})$ implemented as a quantum circuit, s.t.

$$|\psi(\vec{\theta})\rangle = U(\vec{\theta}) |\psi_0\rangle, \quad (4.1)$$

where $|\psi_0\rangle$ is some initial state that can be easily prepared on the quantum device. Commonly, the quantum device is reset to the 0-state at the start of a quantum computation, i.e., $|\psi_0\rangle = |00\dots 0\rangle$.

As E_0 is the smallest eigenvalue, we know that it gives a lower bound on the expectation value of the Hamiltonian \mathcal{H} with respect to the ansatz state $|\psi(\vec{\theta})\rangle$. Thus

$$E_0 \leq F(\vec{\theta}) \equiv \langle \psi(\vec{\theta}) | \mathcal{H} | \psi(\vec{\theta}) \rangle. \quad (4.2)$$

This is referred to as the *variational principle*. For a non-degenerate ground state, equality holds if and only if $|\psi(\vec{\theta})\rangle = |\psi_0\rangle$.

The goal now is to minimize the expectation value $F(\vec{\theta})$ with respect to the variational parameters $\vec{\theta}$. Here, we use a classical optimization algorithm. The quantum part of each step in the optimization process is the preparation of $|\psi(\vec{\theta})\rangle$ with following measurements. From the measurement results, we estimate the expectation value $F(\vec{\theta})$, and this is passed to the classical optimizer that adjusts the variational parameters $\vec{\theta}$.

To utilize the variational method on a problem, quantum or classical, we need to find a mapping of the problem to a quantum operator \mathcal{H} such that the solution is given by its ground state. Note that, assuming a finite, discrete spectrum of eigenstates, one can transform this into the problem of finding the *largest* eigenvalue instead by substituting $\mathcal{H} \rightarrow -\mathcal{H}$, s.t. $F(\vec{\theta}) \rightarrow -F(\vec{\theta})$.

4.2 Variational Quantum Eigensolver

The Variational Quantum Eigensolver (VQE) is a hybrid quantum algorithm that finds ground state energies of physical Hamiltonians. The VQE algorithm is based on a variational approach, as described in Section 4.1.

The input to the algorithm is a fermionic Hamiltonian. This is mapped onto an equivalent qubit Hamiltonian, acting on qubits, by a transformation such as the Jordan–Wigner transformation. We prepare a parametrized ansatz state and minimize its energy by use of a classical optimization algorithm. This procedure is sketched in Figure 4.1.

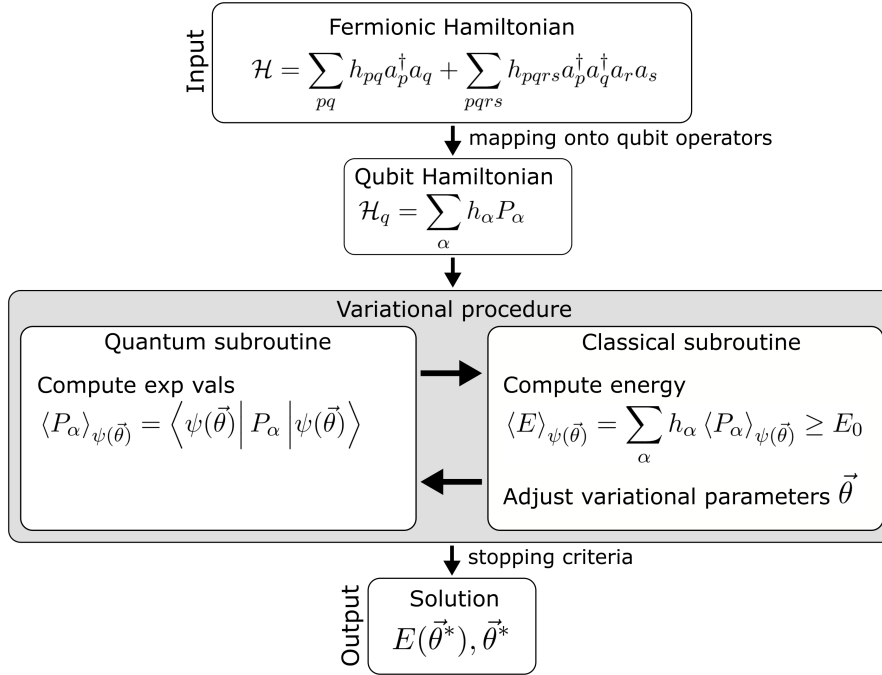


Figure 4.1: Sketch of the procedure for the Variational Quantum Eigensolver (VQE) algorithm.

For the transformation from fermionic to qubit operators, we focus on the aforementioned Jordan–Wigner transformation. This is presented in Section 4.2.2. We further consider three approaches to specific ansatz states. This is the UCCSD ansatz, hardware-efficient ansatzes, and a recent proposal for a symmetry-preserving ansatzes, which are discussed in Section 4.2.3, 4.2.4 and 4.2.5, respectively.

4.2.1 The Molecular Hamiltonian

Our starting point is the Hamiltonian of electrons in matter in the second quantization formalism. This may be written as

$$\mathcal{H} = \sum_{pq} h_{pq} a_p^\dagger a_q + \sum_{pqrs} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s, \quad (4.3)$$

where a_p^\dagger and a_p are respectively the creation and annihilation operators for the orbital p . We consider a subspace of the Fock space spanned by the 2^m

basis states $|f_1 f_2 \dots f_m\rangle$, where f_p is the occupation number of orbital p . As per the Pauli exclusion principle only a single fermion may occupy any single orbital at a time, thus $f_p \in \{0, 1\} \forall p$.

The coefficients h_{pq} and h_{pqrs} are the single and double electron overlap integrals given by

$$h_{pq} = \int d\vec{r} \phi_p^*(\vec{r}) \left(-\frac{\nabla^2}{2} - \sum_I \frac{Z_I}{|\vec{r} - \vec{R}_I|} \right) \phi_q(\vec{r}), \quad (4.4)$$

$$h_{pqrs} = \int d\vec{r}_1 d\vec{r}_2 \phi_p^*(\vec{r}_1) \phi_q^*(\vec{r}_2) \frac{1}{|\vec{r}_1 - \vec{r}_2|} \phi_r(\vec{r}_1) \phi_s(\vec{r}_2), \quad (4.5)$$

which can be computed classically. Here, $\phi_q(\vec{x}_j)$ is the single-particle wavefunction of orbital q . The one-body term h_{pq} represents the kinetic energy and potential energy between the electrons and the nuclei, and the two-body term h_{pqrs} represents the Coloumb repulsion between electrons.

The geometry of the specific molecule in question enters through the nuclei positions \vec{R}_I and charges Z_I . As an example, we consider the H_2 molecule, a linear molecule with two nuclei and two electrons. We have $Z_1 = Z_2 = +1$, $R_1 = (0, 0, 0)$ and $R_2 = (a, 0, 0)$, with an interatomic distance a . Changing the interatomic distance a will alter the coefficients h_{pq} , and thus by Equation (4.3) change the input Hamiltonian. If we for example are interested in the ground state energy as a function of the interatomic distance a we would need to run the full VQE algorithm for each one of a set of discrete distances a_i .

4.2.2 Jordan–Wigner transformation

The first step in the VQE algorithm is to find a mapping from the molecular electron Hamiltonian of Equation (4.3) onto an operator that acts on qubit states. The *Jordan–Wigner transformation* gives such a transformation $\mathcal{H} \rightarrow \mathcal{H}_q$, where the fermionic operators are mapped onto tensor products of Pauli operators. The qubit Hamiltonian \mathcal{H}_q may then be expressed as

$$\mathcal{H}_q = \sum_{\alpha} h_{\alpha} P_{\alpha} = \sum_{\alpha} h_{\alpha} \prod_{i=1}^m \sigma_i^{\beta_{\alpha,i}}, \quad (4.6)$$

for $\beta_{\alpha,i} \in \{0, x, y, z\}$, and $\sigma^0 = \mathbb{I}$. The Hamiltonian terms P_{α} are thus the m -qubit Pauli operators, i.e., the m -fold tensor products of the Pauli operators $\{\sigma^0, \sigma^x, \sigma^y, \sigma^z\}$.

The Jordan–Wigner transformation maps the fermionic annihilation and creation operators onto Pauli operators according to

$$a_p^{\dagger} = \sigma_p^{-} \prod_{k < p} \sigma_k^z, \quad (4.7)$$

$$a_p = \sigma_p^{+} \prod_{k < p} \sigma_k^z, \quad (4.8)$$

where $\sigma_p^\pm = \sigma_p^x \pm i\sigma_p^y$. The 2^m electronic basis states are mapped to qubit states as

$$|f_1 f_2 \dots f_m\rangle \rightarrow |q_1\rangle \otimes |q_2\rangle \otimes \dots \otimes |q_m\rangle \equiv |q_1 q_2 \dots q_m\rangle, \quad (4.9)$$

for qubit states $|q_p\rangle$ where $q_p = f_p \in \{0, 1\}$. Thus, each single-qubit state $|q_p\rangle$ stores the *occupation number* of the corresponding orbital. A $|0\rangle$ -state represents an unoccupied orbital, and a $|1\rangle$ -state represents an occupied orbital.

The energy expectation value may now be written as

$$E_0 \leq \langle E \rangle_{\psi(\vec{\theta})} = \langle \psi(\vec{\theta}) | \mathcal{H}_q | \psi(\vec{\theta}) \rangle = \sum_{\alpha} h_{\alpha} \langle \psi(\vec{\theta}) | P_{\alpha} | \psi(\vec{\theta}) \rangle = \sum_{\alpha} h_{\alpha} \langle P_{\alpha} \rangle_{\psi(\vec{\theta})}, \quad (4.10)$$

i.e., we can compute the energy expectation value by computing the expectation values for each of the individual Pauli-terms P_{α} . Circuit measurement in the eigenbasis of a m -fold Pauli operator tensor can be performed by applying at most m single-qubit gates, one on each qubit, followed by measurements in the computational basis. This is described in more detail in Section 2.4.9.

As an example of a transformed fermionic Hamiltonian, the Pauli-terms P_{α} with corresponding coefficients h_{α} for the H_2 molecule is shown in Table A.1. Note that other transformations exist, such as the spin-parity and the Bravyi-Kitaev transformations [32]. In this text we will focus solely on the Jordan–Wigner transformation, as it is perhaps the simplest and most intuitive of the commonly used mappings.

4.2.3 The UCCSD ansatz

To perform the variational procedure we need a parameterized ansatz state

$$|\psi(\vec{\theta})\rangle = U(\vec{\theta}) |\psi_0\rangle, \quad (4.11)$$

where $U(\vec{\theta})$ is some parameterized unitary operator implemented by means of a quantum circuit. An approach inspired by computational chemistry is the Unitary Coupled Clusters (UCC) ansatz, given by

$$U(\vec{\theta}) = e^{T(\vec{\theta}) - T^\dagger(\vec{\theta})} \quad (4.12)$$

where the operator $T(\vec{\theta})$ is referred to as a *cluster operator*. It is given by

$$T(\vec{\theta}) = \sum_i T_i(\vec{\theta}), \quad (4.13)$$

$$T_1(\vec{\theta}) = \sum_{i,j} \theta_{ij}^j a_j^\dagger a_i, \quad (4.14)$$

$$T_2(\vec{\theta}) = \sum_{i,j,k,l} \theta_{ij}^{kl} a_l^\dagger a_k^\dagger a_j a_i. \quad (4.15)$$

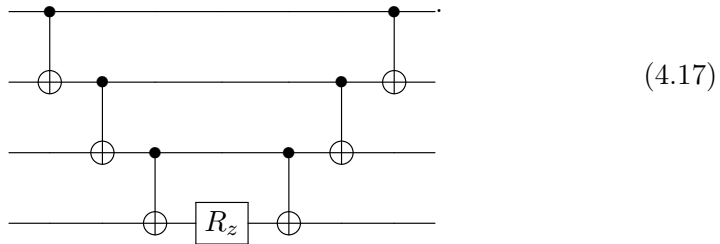
Truncating the cluster operator at the second order, i.e., including only the single and double excitation terms $T_1(\vec{\theta})$ and $T_2(\vec{\theta})$, gives us the *Unitary Coupled Clusters Single Double* (UCCSD) ansatz.

The resulting operator

$$U(\vec{\theta}) = e^{(T_1(\vec{\theta}) - T_1^\dagger(\vec{\theta})) + (T_2(\vec{\theta}) - T_2^\dagger(\vec{\theta}))} \quad (4.16)$$

contains exponentiated terms that do not commute with each other. We therefore need to apply a Suzuki–Trotter approximation in order to write $U(\vec{\theta})$ as a product of separate operators that we can directly implement on a quantum computer. Note that this approximation introduces a so-called Trotterization error [33], however, we claim it to be negligible in the specific cases that we will be considering.

Using the Suzuki–Trotter approximation and again the Jordan–Wigner transformation, we obtain $U(\vec{\theta})$ as a product of single-qubit operators and exponentiated multi-qubit Pauli-operators. As an example, the operator $e^{Z_1 Z_2 Z_3 Z_4}$ is implemented by the following circuit



In general, an exponentiated tensor product of n Pauli-operators can be implemented using $2(n - 1)$ CNOT-gates and 1 single-qubit gate [32].

For the H_2 molecule the full UCCSD ansatz prepares the state

$$|\psi^{H_2}\rangle = U(\vec{\theta}) |\psi^{HF}\rangle = \alpha |0101\rangle + \beta |1010\rangle + \gamma |0110\rangle + \delta |1001\rangle, \quad (4.18)$$

where the initial state $|\psi^{HF}\rangle = |0101\rangle$ is the Hartree-Fock state for H_2 [15]. The circuit that implements $U(\vec{\theta})$ in this case is shown in Equation (A.1).

4.2.4 Hardware-efficient ansatz

The UCCSD ansatz is applicable to arbitrary molecular problems, but suffers from large circuit depths. An alternative approach is to construct an ansatz with the specific quantum hardware in mind, a so-called *hardware-efficient ansatz* [5].

The common approach is to construct hardware-efficient ansatzes in layers of single-qubit gates and so-called *entangler*s. The single-qubit layers consist of parametrized rotational gates. The entanglers are constructed specific to the hardware at hand, by a combination of multi-qubit gates that are compatible

with the configuration of the quantum device. The d -layer

$$U(\vec{\theta}) = \underbrace{\left[\prod_{i=1}^m U_i^{\text{sq}}(\vec{\theta}) \right] U_{\text{ent}}(\vec{\theta}) \left[\prod_{i=1}^m U_i^{\text{sq}}(\vec{\theta}) \right] U_{\text{ent}}(\vec{\theta}) \dots U_{\text{ent}}(\vec{\theta}) \left[\prod_{i=1}^m U_i^{\text{sq}}(\vec{\theta}) \right]}_{\text{Repeated } d \text{ times}}, \quad (4.19)$$

where $U_i^{\text{sq}}(\vec{\theta})$ is a single-qubit operator acting on qubit i , parametrized by a set of variational parameters, and $U_{\text{ent}}(\vec{\theta})$ is an entangler.

The entangler operator $U_{\text{ent}}(\vec{\theta})$ consists of two-qubit operators, e.g., CNOT-operators, in a configuration that is allowed by the *connectivity* between qubits on the quantum device.

An example of such a ansatz circuit

$$\begin{array}{l} q_1 : |0\rangle \\ q_2 : |0\rangle \\ q_3 : |0\rangle \\ q_4 : |0\rangle \end{array} \begin{array}{c} \boxed{R_z(\theta_1)} \\ \boxed{R_z(\theta_2)} \\ \boxed{R_z(\theta_3)} \\ \boxed{R_z(\theta_4)} \end{array} \begin{array}{c} \bullet \\ \oplus \\ \bullet \\ \oplus \end{array} \begin{array}{c} \bullet \\ \oplus \\ \bullet \\ \oplus \end{array} \begin{array}{c} \boxed{R_z(\theta_5)} \\ \boxed{R_z(\theta_6)} \\ \boxed{R_z(\theta_7)} \\ \boxed{R_z(\theta_8)} \end{array} \begin{array}{c} \bullet \\ \oplus \\ \bullet \\ \oplus \end{array} \begin{array}{c} \bullet \\ \oplus \\ \bullet \\ \oplus \end{array} \begin{array}{c} \boxed{R_z(\theta_9)} \\ \boxed{R_z(\theta_{10})} \\ \boxed{R_z(\theta_{11})} \\ \boxed{R_z(\theta_{12})} \end{array} \quad (4.20)$$

The hope is that the resulting state $|\psi(\vec{\theta})\rangle = U(\vec{\theta})|00\dots 0\rangle$ will be able to span the part of the Hilbert space in which the solution exists. This is not in general guaranteed by this type of ad hoc state ansatz. However, results have shown that hardware-efficient ansatz is capable of significantly outperforming UCCSD ansatz in terms of circuit depths and CNOT counts [5].

4.2.5 Symmetry-preserving ansatz

A recent article introduced a general scheme to construct efficient ansatz circuits for molecules that conserve certain symmetries. The idea is to recognise symmetries that valid solution states must fulfill, and then construct state ansatz that span the appropriate symmetry subspace. This is achieved by use of parameterized composite gates that conserve these symmetries.

The idea is to span by the appropriate symmetry subspaces by constructing parameterized gates that conserve these symmetries.

Take the H_2 molecule as an example, for which we may construct basis states

$$|f_{\sigma_{u,\downarrow}} f_{\sigma_{g,\downarrow}} f_{\sigma_{u,\uparrow}} f_{\sigma_{g,\uparrow}}\rangle, \quad (4.21)$$

where $\sigma_{k,s}$ are molecular orbitals with spin s , and $f_{\sigma_{k,s}} \in \{0,1\}$ are the respective occupation numbers [15]. These states span a $2^4 = 16$ dimensional complex vector space.

A first symmetry that we may consider is particle number conservation, given by $\sum_i f_i = n$. For the H_2 molecule we want to place 2 electrons amongst the 4 available orbitals. The basis states that fulfill this condition are the states

$$\{|1100\rangle, |1010\rangle, |1001\rangle, |0110\rangle, |0101\rangle, |0011\rangle\}, \quad (4.22)$$

and thus any valid solution must be a linear combination of these states.

In order to construct ansatz circuits that achieve this, we construct a composite two-qubit gate that entangles two qubits while conserving the particle number. This is achieved by the operator

$$A(\theta, \phi) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & e^{i\phi} \sin \theta & 0 \\ 0 & e^{-i\phi} \sin \theta & -\cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4.23)$$

with parameters θ and ϕ . We observe that this operator will mix the basis states $|01\rangle$ and $|10\rangle$, while it leaves $|00\rangle$ and $|11\rangle$ unchanged. This operator can be implemented on a quantum computer as

$$= \begin{array}{c} \oplus \\ \text{---} \\ \bullet \\ \text{---} \\ \oplus \end{array} \begin{array}{c} \text{---} \\ \bullet \\ \text{---} \\ \oplus \end{array} \begin{array}{c} \text{---} \\ \oplus \\ \text{---} \\ \oplus \end{array}, \quad (4.24)$$

where $R(\theta, \phi) = R_z(\phi + \pi)R_y(\theta + \pi/2)$. The rotation gates R_x and R_z are defined in Equation (2.59).

An ansatz circuit that spans the subspace for $m = 4$ orbitals and $n = 2$ particles with minimal number of parameters is the following

$$\begin{array}{c} |0\rangle \\ |0\rangle \\ |0\rangle \\ |0\rangle \end{array} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array}, \quad (4.25)$$

which uses 10 parameters to span the subspace defined by the 6 basis states in Equation (4.22). This is the minimal number of parameters for which this may be accomplished. This can be further expanded to give a general procedure for arbitrary m and n . Similar circuits that take into account spin symmetries can also be constructed [6].

Chapter 5

Quantum Error Mitigation

As discussed so far, there are two main challenges faced by near-future NISQ-era quantum computation; noise, and limited scaling in terms of available, connected qubits. In the long term, *quantum error detection* promises to provide means to continuously detect and correct errors in quantum computations [7, 8]. Given that enough qubits are available, this can be done to arbitrary precision. While this gives promise for the long-term viability of quantum algorithms, the ancillary qubit requirements are far beyond the capabilities of near-term quantum hardware. The basis of error correction by stabilizer codes is described in Appendix C.

For near-term applications, such as the hybrid algorithms presented in Chapter 4, we instead turn to *quantum error mitigation*. By quantum error mitigation, we mean techniques that aim to reduce the impact of noise, i.e., to mitigate the effect of noise, in quantum circuits. This is in contrast to the quantum error correction techniques, that aim to exactly identify and correct certain errors that may occur. In practice, quantum error mitigation can often be performed with small-to-none additional qubit requirements, thus making them viable for near-future quantum hardware.

This chapter presents two particular, state-of-the-art quantum error mitigation techniques. The zero-noise extrapolation, where noise is amplified by known amplification factors in order to extrapolate the expectation value to the zero-noise case. As well as the error detection scheme, where we recognise symmetries in the states that we want to prepare and detect errors by measuring if these symmetries are upheld.

For the zero-noise extrapolation technique we propose a circuit level noise-amplification scheme for noise in CNOT-gates. The scheme we propose has, to the best of our knowledge, not been previously explored in literature.

5.1 Zero-Noise Extrapolation

The idea behind the zero-noise extrapolation scheme is to extrapolate the results of a noisy circuit to the zero-noise case by computing a set of noise-

amplified circuits, each with the noise amplified by known factors [13, 14].

The base assumption of this method is that the expectation value $E(\lambda)$ of some circuit executed on a noisy quantum computer, that evaluates some observable A , can be expanded as a power series as

$$E(\lambda) = E^* + \sum_{k=1}^n a_k \lambda^k + O(\lambda^{n+1}), \quad (5.1)$$

where $\lambda \ll 1$ is a small noise parameter and E^* is the expectation in the noiseless case.

Mitigating all noise on a quantum device is often infeasible. Instead, we may turn our attention to specific components of the noise that we expect to be the dominant contributors to errors. As an example consider a quantum device with independent noise on CNOT-gates, on single-qubit gates and on measurements. We isolate the CNOT-noise by now expanding the noisy expectation value as

$$E(\lambda_{\text{cnot}}) = (E^* + \delta_{\text{sq}} + \delta_{\text{meas}}) + \sum_{k=1}^n a_k \lambda_{\text{cnot}}^k + O(\lambda_{\text{cnot}}^{n+1}), \quad (5.2)$$

where $\lambda_{\text{cnot}} \ll 1$ now parametrizes specifically the noise on CNOT-gates, and δ_{sq} and δ_{SPAM} encapsulates the shift in the expectation value due to noise on single-qubit gates, and state preparation and measurement (SPAM) noise, respectively. Mitigating the CNOT-noise is sufficient if the single-qubit and measurement noise is negligible, i.e., if $\delta_{\text{sq}}, \delta_{\text{meas}} \ll 1$.

To make use of the zero-noise extrapolation we need some procedure to compute the noise amplified expectation values $\tilde{E}(r_i \lambda)$, with the noise "amplified" such that $\lambda \rightarrow r_i \lambda$, for a set of n known noise amplification factors $\{r_i\}_{i=1}^n$. In Section 5.1.2 and 5.1.3 we introduce two separate schemes for noise amplification. Respectively, noise amplification by sampling of random Pauli-gates and by repeating CNOT-gates. The latter scheme has to the best of our knowledge not been explored previously.

5.1.1 Richardson Extrapolation

Assuming that the noisy expectation value may be expressed as in Equation (5.1) and given a set of noise amplified expectation values $\tilde{E}(r_i \lambda)$ for known amplification factors $\{r_i\}_{i=1}^n$, the *Richardson extrapolation* can be used to obtain an extrapolation to the $\lambda \rightarrow 0$ limit, [34]. The Richardson extrapolation reads

$$\tilde{E}^{(n)} = \sum_{i=1}^n \gamma_i \tilde{E}(r_i \lambda), \quad (5.3)$$

where the coefficients γ_i are determined such as to fulfill the restrictions

$$\sum_i^n \gamma_i = 1, \quad \sum_i^n \gamma_i r_i^k = 0, \quad (5.4)$$

for all $k = 1, \dots, n$. The problem of finding the coefficients γ_i thus reduces to solving a set of linear equations. Now, $\tilde{E}^{(n)}$ serves as an estimator for the noise-free expectation value E^* .

5.1.2 Noise Amplification by Random Pauli-Gate Sampling

One proposed scheme for artificially amplifying noise in multi-qubit gates is randomly add Pauli-gates after each multi-qubit gate in order to imitate the noise present. This method hinges on approximating the noise as a stochastic Pauli-noise channel, and on the noise in the single-qubit Pauli-gates to be negligible compared to the multi-qubit gate noise that we want to mitigate, [23].

To compute the noise amplified expectation value $\tilde{E}(r_i, \lambda)$, we sample from a large number of such circuits with random Pauli-gates added with a probability chosen such as to obtain the desired amplification factor r_i . We refer to this scheme as the *random Pauli-sampling* scheme.

We focus on the CNOT-gate. Assume that the noise present on CNOT-gates in some quantum device is described by the two-qubit depolarizing noise channel with error rate p as given by Equation (3.8). To employ this scheme, the error rate p needs to be approximated experimentally.

To obtain a circuit where the effective noise has been amplified as $p \rightarrow rp$ we add two random Pauli-gates σ^e and σ^f as



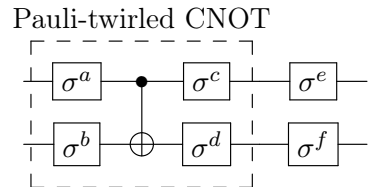
$$(5.5)$$

with probability $(r-1)p$. Both σ^e and σ^f are chosen randomly from the Pauli operators $\{\mathbb{I}, \sigma^x, \sigma^z, \sigma^y\}$. However, we omit the identity term given by $\sigma^e = \sigma^f = \mathbb{I}$, as this term would cause no noise amplification.

The Random Pauli-sampling method relies on approximating the noise channel on CNOT gates as a stochastic Pauli-noise channel. This may not be accurate for general quantum noise on real devices. However, we may use the technique of *Pauli-twirling* to transform general incoherent noise channel into stochastic Pauli-noise channels [35]. The concept of Pauli-twirling is explained in Appendix B.

Applying Pauli-twirling to each CNOT-gate involves 4 additional single-qubit gates. The full noise amplification procedure for each CNOT-gate then becomes

Pauli-twirled CNOT



$$(5.6)$$

where σ^e and σ^f are applied with a probability of $r \cdot p$, σ^a and σ^b are randomly chosen from the Pauli-operators. The operators σ^c and σ^d are chosen such that

$(\sigma^a \otimes \sigma^b) \text{CNOT}(\sigma^c \otimes \sigma^d) = e^{i\phi} \text{CNOT}$ in the noiseless case, where $e^{i\phi}$ is a global phase factor. For the extra Pauli-twirling step to be effective the advantages of transforming the noise channels into stochastic Pauli-noise channels must outweigh the added noise from the 4 added single-qubit gates. This is explored in Figure 6.2.

5.1.3 Noise Amplification by Repeating CNOT-Gates

We proposed here a scheme for circuit-level noise amplification with minimal assumptions about the specifics of the quantum noise. This scheme has to the best of our knowledge not been proposed previously. The idea is to amplify general noise in CNOT-gates by odd factors, $r = 1, 3, 5, \dots$, by replacing each CNOT-gate in the original circuit by a sequence of r CNOT-gates. The idea is to utilize that the CNOT-gate is its own inverse, i.e.,

$$(\text{CNOT})^2 = \text{CNOT} \circ \text{CNOT} = \mathbb{I}, \quad (5.7)$$

where both CNOT-operators act on the same control and target qubits.

A sequence of an odd number of ideal CNOT-gates thus is equivalent to a single CNOT-gate, as

$$[\text{CNOT} \circ (\text{CNOT})^{2n}] = \text{CNOT}, \quad (5.8)$$

where n is an arbitrary integer. Consider now a noise-afflicted CNOT-gate. Assume the noise to be weak, and write the noisy CNOT-procedure as $\mathcal{N} \circ \text{CNOT}$, i.e., as an ideal CNOT-gate followed by an effective noise channel \mathcal{N} .

Now, when replacing each CNOT-gate by a sequence of $r = 2n + 1$ CNOT-gates the noise channel \mathcal{N} will be applied r times throughout the sequence. Thus, the noise can be said to be amplified by a factor r . For $r = 3$ this substitution becomes



$$\quad (5.9)$$

Repeating this procedure for n noise amplification factors $r_i = 1, 3, 5, \dots, 2n+1$ gives an algorithm to compute n noise amplified expectation values $\tilde{E}(r_i \lambda)$ from which to compute the mitigated expectation value $\tilde{E}^{(n)}$. We will refer to this scheme as the *CNOT repetition* scheme.

While the CNOT-gate is perhaps the most relevant gate for such a noise amplification schemes with regards to modern quantum processors, this approach should work for other common gate sets. The requirement is that the gates with significant noise are their own inverse up to a global phase factor. This is the case for several commonly used multi-qubit gates, including the TOFFOLI-gate, as well as the cZ and cY controlled Pauli-gates.

5.1.4 Variance Scaling

As described in Section 2.4.11, larger variance gives a larger statistical error in the estimation of an expectation value. This must then be combated by

increasing the number of circuit shots leading to a higher computational cost. A drawback of the zero-noise extrapolation is indeed a scaling of the variance dependent on the specific γ_i -coefficients.

For n stochastic variables X_i with variances $\text{Var}[X_i]$ the variance of the sum becomes

$$\text{Var}\left[\sum_{i=1}^n X_i\right] = \sum_{i=1}^n \text{Var}[X_i] + 2 \sum_{i=1}^n \sum_{j=1}^{i-1} \text{Cov}[X_i, X_j] \quad (5.10)$$

where $\text{Cov}[X_i, X_j]$ is the covariance between X_i and X_j . Note that $\text{Cov}[X_i, X_j] = 0$ if X_i and X_j are independent variables. Furthermore, for a stochastic variable X we have

$$\text{Var}[aX] = a^2 \text{Var}[X] \quad (5.11)$$

where a is some scalar.

We can now evaluate the variance mitigated expectation value estimator. Let $\text{Var}[\tilde{E}(r_i\lambda)]$ be the variance in the expectation value estimator for the circuit with amplification factor r_i . We assume that all estimators $\tilde{E}(r_i\lambda)$, $i = 1, \dots, n$, are pairwise independently distributed. Then, from Equation (5.3), we get the variance in the mitigated expectation value estimator $\tilde{E}^{(n)}$ as

$$\sigma_{K,mit}^2 \equiv \text{Var}[\tilde{E}^{(n)}] = \sum_{i=1}^n \gamma_i^2 \text{Var}[\tilde{E}(r_i\lambda)], \quad (5.12)$$

where the γ_i -coefficients are the coefficients found by Equation (5.4).

Let σ_i^2 be the variance associated with the measurement outcome of the circuit with noise amplification factor r_i , and N_i the number of shots used in the estimation of the corresponding expectation value. An estimator for the variance in $\tilde{E}(r_i\lambda)$ is given by Equation (2.103). We furthermore get an estimator for the error in the mitigated expectation value $\tilde{E}^{(n)}$ as

$$\epsilon_{mit}^{(n)} = \sqrt{\sum_{i=1}^n (\gamma_i^{(n)})^2 \frac{\sigma_i^2}{N_i}}, \quad (5.13)$$

where $\gamma_i^{(n)}$ is the i -th coefficient in the Richardson extrapolation when using the n noise amplification factors $r_i = 1, 3, 5, \dots, 2n-1$. The specific $\gamma_i^{(n)}$ -coefficients for all cases from $n = 2$ to $n = 10$ is shown in Table 5.1.

More often than not, however, we will not be interested in computing the error after the fact. We rather will have a specific error tolerance, ϵ_{tol} , in mind and want to find the number of circuit shots needed to get the statistical error below our tolerance. A first estimation

$$N = \frac{1}{\epsilon_{tol}^2} \sum_{i=1}^n (\gamma_i^{(n)})^2 \sigma_i^2 \quad (5.14)$$

Table 5.1: Values of the coefficients $\gamma_i^{(n)}$, $i = 1, 2, \dots, n$, for the Richardson extrapolation with noise amplification factors $c_i = 2i - 1$, $i = 1, 2, \dots, n$, for different values of n . The coefficients in each case of n are calculated from Equation (5.4).

	$n = 2$	$n = 3$	$n = 4$	$n = 5$	$n = 6$	$n = 7$	$n = 8$	$n = 9$	$n = 10$
$\gamma_1^{(n)}$	1.50	1.88	2.19	2.46	2.71	2.93	3.14	3.11	2.97
$\gamma_2^{(n)}$	-0.500	-1.25	-2.19	-3.28	-4.51	-5.87	-7.33	-7.05	-6.00
$\gamma_3^{(n)}$	-	0.375	1.31	2.95	5.41	8.80	13.2	12.2	8.80
$\gamma_4^{(n)}$	-	-	-0.313	-1.41	-3.87	-8.38	-15.7	-13.8	-7.52
$\gamma_5^{(n)}$	-	-	-	0.273	1.50	4.89	12.2	9.78	3.07
$\gamma_6^{(n)}$	-	-	-	-	-0.24	-1.60	-6.00	-4.05	0.0247
$\gamma_7^{(n)}$	-	-	-	-	-	0.226	1.69	0.717	-0.240
$\gamma_8^{(n)}$	-	-	-	-	-	-	-0.21	0.0691	-0.274
$\gamma_9^{(n)}$	-	-	-	-	-	-	-	-0.0348	0.239
$\gamma_{10}^{(n)}$	-	-	-	-	-	-	-	-	0.0514

The total number of circuit shots executed in this case will be $N_{\text{tot}} = n \cdot N$.

The variance scaling introduces a few pit falls for the practical application of the zero-noise extrapolation technique. Firstly, we see from Equation (5.13) and Table 5.1 that the specific γ -coefficients will have considerable impact on the statistical error.

We expect the mitigated expectation value to converge towards the ideal value when adding more noise-amplified terms, i.e., when increasing n . However, we cannot simply add further noise-amplified results without further ado. Due to the variance scaling we might have to increase the number of shots over *all* the noise-amplified circuits to compensate for an increased statistical error.

5.2 Error Detection

Another recently proposed error mitigation scheme suitable for near-term devices is error mitigation based on *error detection* [15, 16]. Quantum error detection is described in more detail in Appendix C. The idea is to recognise symmetries in the quantum states that we want to prepare and use ancillary qubit measurements to detect when these symmetries are broken. If they are, there must have occurred an error somewhere in the circuit. When an error is detected in a circuit execution, we discard the corresponding measurement result.

The ideas of this error mitigation scheme are quite similar to that of quantum error correction by stabilizer codes [7, 8]. The main difference, however, is that in quantum error correction we aim to not only detect that an error occurred, but also to identify the specific error such that it may be *corrected*. Error correction also aims to correct general errors, or at least large sets of errors, while the proposed error detection scheme only targets very specific

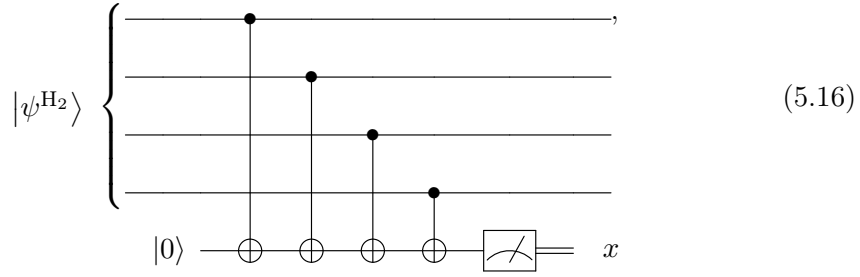
errors that are easily measured. Restricting ourselves to detection may still offer improvements in terms of noise mitigation but without the large ancillary qubit requirements of error correcting codes, and thus may be more feasible for near-term quantum hardware.

As an example, take the following quantum state

$$|\psi^{\text{H}_2}\rangle = \alpha |0101\rangle + \beta |1010\rangle + \gamma |0110\rangle + \delta |1001\rangle, \quad (5.15)$$

with arbitrary complex coefficients $\alpha, \beta, \gamma, \delta$. This covers the exact states spanned by the UCCSD ansatz for the H_2 molecule [15], as presented in Section 4.2.3.

We recognise that $|\psi^{\text{H}_2}\rangle$ is a $+1$ eigenstate of the $ZZZZ$ operator. This is an example of a symmetry of this state. A non-destructive measurement of the state $|\psi^{\text{H}_2}\rangle$ with respect to the $ZZZZ$ operator can be performed by the following quantum sub-circuit



where the measurement outcome is stored in the classical bit x . Because the state $|\psi^{\text{H}_2}\rangle$ lies in the $+1$ eigenspace of the $ZZZZ$ -operator, this measurement will always return a measurement outcome of $+1$ in the noiseless case.

This may, however, not necessarily be the case when we introduce noise. Say a bit-flip error occur on bit 1, e.g., a Pauli- X operator. Because this operator *anti-commutes* with the $ZZZZ$ -operator we observe that

$$ZZZZ(XIII |\psi^{\text{H}_2}\rangle) = -XIII (ZZZZ |\psi^{\text{H}_2}\rangle) = -XIII |\psi^{\text{H}_2}\rangle, \quad (5.17)$$

i.e., the post-error state is now a -1 eigenstate of $ZZZZ$. The state has been *moved out* of the $+1$ eigenspace, so to say. Such an error is *detected* by the measurement, as we now measure an outcome of -1 .

If an error has occurred but the error operator commutes with the $ZZZZ$ -operator, we will still measure $+1$ and no further information is gained. Thus, this procedure leaves us with a set of *detectable* and a set of *undetectable* errors. These are exactly the operators that respectively commutes or anticommutes with our error detection operator, in this case the $ZZZZ$ -operator

This, of course, does not capture every possible quantum error operator. But any general quantum operator O may be decomposed as $O = aO^+ + bO^-$, for two operators O^+ and O^- that respectively commute and anti-commute with our $ZZZZ$ -operator. The interaction with the $ZZZZ$ -operator now goes

as follows;

$$\begin{aligned}
ZZZZ(O|\psi^{H_2}\rangle) &= ZZZZ(aO^+ + bO^-)|\psi^{H_2}\rangle \\
&= aO^+(ZZZZ|\psi^{H_2}\rangle) - bO^-(ZZZZ|\psi^{H_2}\rangle) \quad (5.18) \\
&= aO^+|\psi^{H_2}\rangle - bO^-|\psi^{H_2}\rangle,
\end{aligned}$$

i.e., the state $|\psi\rangle$ been transformed by the error operator O into a superposition of a +1 eigenstate and a -1 eigenstate of the $ZZZZ$ -operator. A $ZZZZ$ -measurement will give a measurement outcome of +1 with probability $|a|^2$ and -1 with probability $|b|^2$. The post-measurement state will correspondingly either be afflicted with a detectable error, or not, with certainty up to possible measurement errors. We say that the error has been *digitized*, from an "analog" arbitrary superposition to the "binary" case where the state is either afflicted by a detectable error or not.

A drawback of the error detection scheme is that the scheme adds additional noisy multi-qubit gates. As an example, the circuit in Equation (5.16) adds 4 CNOT-gates. This is permissible if the noise added from the error detection subcircuit is negligible compared to the noise already present. For example, if the CNOT count contributed by the error detection sub-circuit is small compared to the CNOT count in the original circuit.

Some computational cost is added by the error detection scheme, as we risk discarded large portions of the executed shots. Assume that we need to perform N shots of some circuit in order to estimate the expectation value to within the desired error. Say that the scheme discards 50% of the shots. We would then need to perform twice as many shots in total in order to obtain N non-discarded shots. However, we expect the added shot count to stay within one order of magnitude. This is supported by the findings in Section 6.2.3, shown in Table 6.5.

Chapter 6

Experiments and Results

This chapter presents the experiments that we have performed to investigate the viability of quantum error mitigation on near-future applicable quantum computations. We further present and discuss the results from said experiments.

We have done experiments with the zero-noise extrapolation technique on a trial circuit, the so-called SWAP-test circuit, which are presented in Section 6.1. This was done with our proposed noise amplification scheme by CNOT-repetition, as described in Section 5.1.3, to test its viability and efficiency in mitigating quantum noise. We also performed experiments with the noise amplification scheme by random-Pauli sampling, in order to compare the results from our scheme with a state-of-the-art noise amplification scheme. This was done for several different noise models.

We further explored the potential in combining the zero-noise extrapolation technique with the error detection scheme, as described in Section 5.2. Again, we used our proposed noise amplification scheme. In order to trial these error mitigation techniques on a near-future applicable quantum algorithm, we chose the VQE algorithm. As discussed in Section 4.2, this is a quantum-classical hybrid algorithm for molecular simulations, and is a candidate for one of the first applicable algorithm to real-world problems.

We have used the qiskit library for Python in order to construct our quantum computations [36, 37], and the OpenFermion library to perform the molecular integrals needed for the VQE algorithm [38].

In qiskit, to perform a quantum computation, the quantum circuits are executed on a *backend* for a specified number of shots. The backend may be a simulator, that classically simulate a quantum computer, or one of the IBMQ publicly available quantum computers [27]. We then interpret the measurement results by classical post-processing, for example in order to compute an energy expectation value. We have used simulator backends in all our experiments, with various different noise models that models the noise present in current quantum hardware.

6.1 Zero-Noise Extrapolation for CNOT-Gates

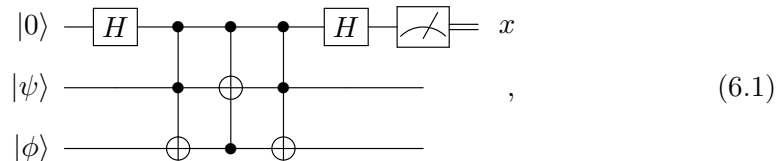
There are several aspects of the zero-noise extrapolation with the CNOT-repetition scheme for noise amplification that we want to examine and discuss. We want to inspect its effectiveness in mitigating errors, on different models for quantum noise and on real quantum processors, and its potential drawbacks in doing so. We further want to compare the results from our amplification scheme with other state-of-the-art schemes, such as the random Pauli-sampling scheme. To do this, we will employ the scheme to a trial quantum circuit.

In all cases, we use the noise amplification schemes to exclusively amplify noise on CNOT-gates, that is the multi-qubit gate native to all current IBMQ devices. Accordingly, this is the noise that is mitigated by the zero-noise extrapolation scheme. For the CNOT-repetition scheme we will use amplification factors $r = 1, 3, 5, \dots, 2n - 1$. For the random-Pauli scheme we will use amplification factors $r = 1, 2, 4, \dots, 2^{n-1}$, as this has seemed to yield best results.

6.1.1 The SWAP-test Circuit

We chose the SWAP-test circuit on 3 qubits as a trial circuit. The SWAP-test circuit has a *CNOT count* of 18 and a depth of 36 when assuming full qubit connectivity. The SWAP-test circuit was chosen because of this intermediate scale. On modern quantum hardware a CNOT count in the order of 10^1 is in the range where information is not completely lost to decoherence. Yet, some sort of noise reduction may be required to obtain useful results. This will be seen from the experiment results in Section 6.1.2, in particular in Figure 6.1.

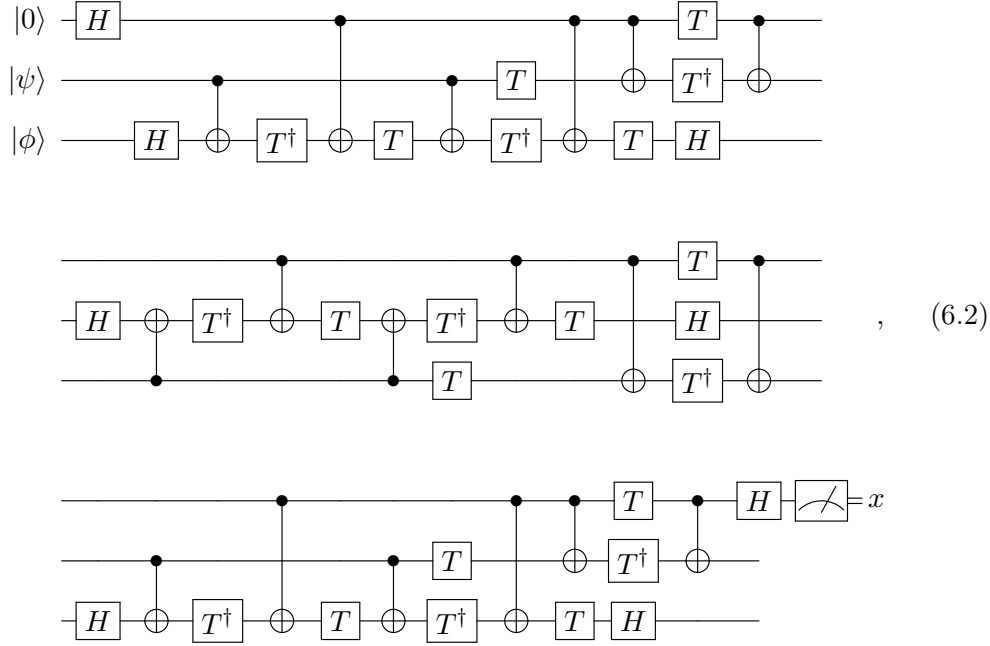
The SWAP-test circuit measures the overlap between two quantum states. The circuit on two single-qubit states $|\psi\rangle$ and $|\phi\rangle$ is expressed in terms of Toffoli gates as



$$\begin{array}{c}
 |0\rangle \text{---} [H] \text{---} \bullet \text{---} \bullet \text{---} \bullet \text{---} [H] \text{---} \text{Measurement} = x \\
 |\psi\rangle \text{---} \bullet \text{---} \oplus \text{---} \bullet \text{---} \\
 |\phi\rangle \text{---} \oplus \text{---} \bullet \text{---} \oplus \text{---}
 \end{array} \quad , \quad (6.1)$$

where the upper qubit, initialized to the $|0\rangle$ -state, is referred to as the probe qubit. The IBMQ devices are all based on gate sets consisting of the CNOT-gate and single-qubit gates. Thus, we need to decompose each Toffoli-gate into such gates if we want to make the circuit compatible with these devices. Such a decomposition is given by Equation (2.79). The resulting quantum circuit

then becomes



where each line contains one Toffoli-gate decomposed into CNOT-gates and single-qubit H , T and T^\dagger -gates.

An advantage of the SWAP-test circuit is the ease of analysis in the measurement results. A Z -type measurement on the probe qubit yields an expectation value of

$$\langle Z \rangle_{\text{probe}} = |\langle \psi | \phi \rangle|^2, \quad (6.3)$$

where the states $|\psi\rangle$ and $|\phi\rangle$ are the two states initialized on the second and third qubits. Thus, the SWAP-test experiment, of repeated runs of the SWAP-test circuit with subsequent measurements, will estimate the *overlap* between the input states $|\psi\rangle$ and $|\phi\rangle$.

For example, we may choose $|\psi\rangle = |0\rangle$ and $|\phi\rangle = |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. The $|0\rangle$ -state is initialized by default and the $|+\rangle$ state requires a single additional Hadamard-gate to prepare. The expectation value then becomes

$$\langle Z \rangle_{\text{probe}} = |\langle 0 | + \rangle|^2 = 0.5. \quad (6.4)$$

The CNOT-complexity and circuit depth of the noise amplified circuits will scale with the amplification factor r . If the bare circuit, i.e., the circuit with amplification factor $r = 1$, has a CNOT-count of k , then the amplified circuit with amplification factor r will have a CNOT count of $r \cdot k$. The CNOT count and circuit depths for the SWAP-test circuit after optimization by the `qiskit` transpiler, and with noise amplification by the CNOT repetition scheme, is shown in Table 6.1. For real devices the CNOT count will be impacted by limited qubit connectivity, as SWAP-gates might need to be inserted to compensate for non-connected qubits. We ran the circuit with `qiskit` transpiler

Noise amplification factor, r	CNOT count	Depth
1	18	36
3	54	72
5	90	108
7	126	144
9	162	180
11	198	216
13	234	252

Table 6.1: CNOT counts and circuit depths for the SWAP-test circuit when noise amplified with the CNOT repetition scheme. The circuit has been optimized assuming full qubit connectivity.

for the specifications of the IBMQ Belem device. This outputted a circuit with a CNOT count of 30, compared to the CNOT-count of 18 if we assume total connectivity, and a depth of 46. When applying the CNOT repetition scheme the CNOT count still scales from the bare circuit in exactly the same way, proportional to the noise amplification factor r .

6.1.2 Experiments

Approximate noise models emulating the noise in specific IBMQ devices can be retrieved by `qiskit`. In order to examine the effect of noise on different gates and measurements, with realistic error rates, we retrieved such a noise model for the IBMQ Vigo device. We decomposed the noise model into the specific CNOT-gates, single-qubit gate and measurement components. The SWAP-test experiment was then run for four different noise models based on these components; with only the CNOT-gate noise, with CNOT-gate and single-qubit gate noise, with CNOT-gate and measurement noise, and the full combined noise model.

This experiment was run with the CNOT-repetition scheme for different numbers of noise amplification factors. The mitigated expectation values for each noise model is shown in Figure 6.1. The mitigated expectation values are plotted as a function of the number of noise amplification factors used, as this gives an indication of the computational cost. For n noise amplification factors, we would need to construct n noise-amplified circuits, and estimate the n corresponding expectation values. We executed all circuits for a total of $128 \cdot 8192 = 1048576$ in order to estimate their expectation value.

The input states were chosen such that $|\langle\psi|\phi\rangle|^2 = \frac{\pi}{4}$. This was chosen, somewhat arbitrarily, to avoid any possible unforeseen symmetry effects. One example of such a symmetry effect would be the case where $|\langle\psi|\phi\rangle|^2 = 0$. We would expect heavier noise amplification to move the end state closer to full decoherence, i.e., a fully mixed density state. Such a state would itself have an expectation value of $\langle Z \rangle = 0$, thus it would be impossible to gain information about the general effectiveness of the zero-noise extrapolation scheme.

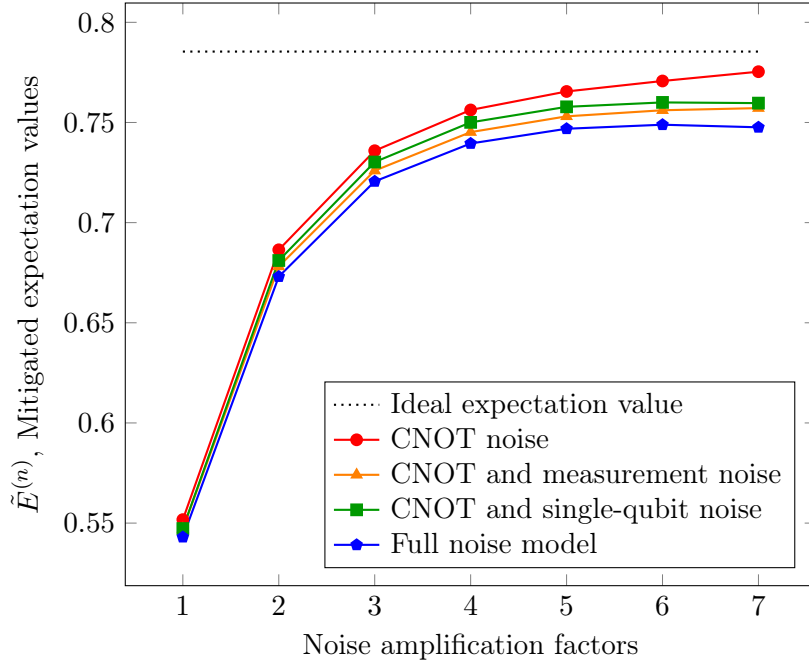


Figure 6.1: Mitigated expectation values for the SWAP-test circuit as a function of the number of amplification factors. Run on a simulator with noise models taken from the IBMQ Vigo device, restructured into separate models with the CNOT noise only, CNOT and measurement noise, CNOT and single-qubit gate noise, and the full IBMQ Vigo noise model.

From Figure 6.1 we observe that the zero-noise extrapolation converges towards the ideal expectation value when only CNOT-noise is present, as we would expect. The bare circuit expectation value has a relative deviation of 30% from the ideal expectation value in all 4 cases. At $n = 7$ the mitigated expectation value has converged to within a relative deviation of 3.3%, when considering only CNOT-noise. This suggests that the CNOT-repetition scheme is effective in amplifying the noise by the desired amplification factors.

When introducing single-qubit noise and measurement noise the mitigated expectation values instead converges to a point that is slightly shifted from the noise-free expectation value E^* , as described by Equation (5.2). As we only mitigate the CNOT-noise, this noise manifests itself as a shift δ in the expectation value. As expected, we may then at best hope that the mitigated expectation value converges towards

$$E_{\text{eff}}^* = E^* + \delta \quad (6.5)$$

where E^* is the ideal expectation value and δ is the shift due to single-qubit noise, as well as state-preparation and measurement (SPAM) noise. We observe that this shift is small compared to the error cause by the CNOT-noise that we do successfully mitigate. At $n = 7$ the relative deviation is 4.8% when

considering the full noise model.

We further want to examine the efficiency of the CNOT-repetition scheme compared to the alternative random-Pauli sampling scheme, as described in Section 5.1.2.

We ran the SWAP-test experiment with states on a simulator with several different noise models. We set the states as $|\psi\rangle = |0\rangle$ and $|\phi\rangle = |+\rangle$, such that $|\langle\psi|\phi\rangle|^2 = 0.5$. A noise model emulating the noise in the IBMQ Athens quantum processor was retrieved from `qiskit`. Depolarizing, amplitude damping and phase damping noise models was constructed with noise both on CNOT-gates and on single-qubit gates. In all cases, the error rate was chosen a factor of 10 larger on CNOT-gate noise than on single-qubit gate noise.

The mitigated expectation values for these experiments as a function of the number of amplification factors is shown in Figure 6.2. Although the two schemes use different sets of noise amplification factors, the main factor for computational complexity is the number of noise amplification factors used. The relative errors $\Delta_K = \frac{|E_K - E^*|}{E^*}$ in all cases, for $n = 7$ noise amplification factors, are shown in Table 6.2.

As seen from Figure 6.2 the random-Pauli scheme mitigates depolarizing noise well but fails to compete with the repeating CNOTs scheme for other noise models. This is as expected, as the random-Pauli scheme relies on the noise model to be approximated as a stochastic Pauli noise channel. As seen from Equation (3.8) the depolarizing noise channel is just a homogeneous stochastic Pauli noise channel.

The repeating CNOT-scheme relies on no such assumption about the specific characteristics of the noise channels on CNOT-gates. From Table 6.2 we observe that the mitigated expectation values converge to relative errors within 1% for the depolarizing, amplitude damping and phase damping noise models, and within 2% for the IBMQ Athens noise model. This is for $n = 7$ amplification terms.

For the amplitude and phase damping noise models, the random-Pauli scheme converges to a value with a relative error that is one order of magnitude larger than that of the CNOT-repetition scheme. The CNOT-repetition scheme also gives an improvement by at least a factor of two for the IBMQ Athens and depolarizing noise models. We assess that the CNOT-repetition scheme gives a more accurate amplification of the noise given by these noise models. The improvements are especially significant for noise models that differ from the stochastic Pauli-noise models.

Interestingly, the process of Pauli-twirling the CNOT-gates does not significantly improve the error mitigation in any of the experiments. For both the IBMQ Athens noise model and the amplitude damping noise model the Pauli-twirling yields a relative error that is larger by about a factor of 2.

We do suspect that the random-Pauli scheme should be more efficient on noise described as stochastic Pauli-noise models. The above results from the depolarizing noise model supports this claim. However, the results after adding

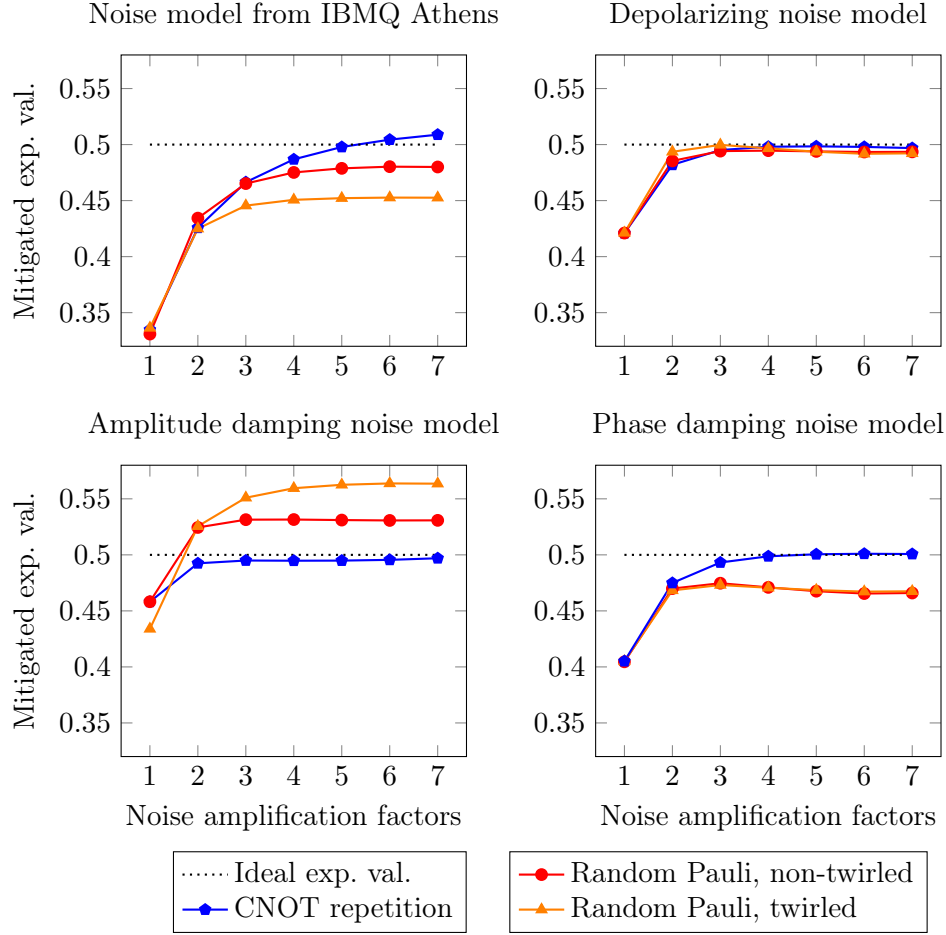


Figure 6.2: Mitigated expectation values for the zero-noise extrapolation with noise amplification by CNOT repetition, and by random Pauli-gate sampling with and without Pauli-twirling. This was run on a simulator with different noise models and with different numbers of amplification factors.

Noise model	$\Delta_K^{\text{CNOT-rep}}$	$\Delta_K^{\text{random-Pauli}}$	$\Delta_K^{\text{random-Pauli, twirled}}$
IBMQ Athens	1.8%	4.0%	9.6%
Depolarizing	0.62%	1.3%	1.6%
Amplitude damping	0.60%	6.2%	13%
Phase damping	0.14%	6.8%	6.5%

Table 6.2: Relative errors Δ_K in the mitigated expectation values $E_K(\delta)$ when employing the zero-noise extrapolation scheme with $n = 7$ noise amplification factors to the SWAP-test circuit. The noise amplification was done by three different schemes; the CNOT-repetition scheme, and the random-Pauli scheme without and with Pauli-twirling.

Pauli-twirling suggests that any possible improvements gained by transforming the noise models into stochastic Pauli-noise models is lost by the additional noise due to the added noisy single-qubit gates. As seen from Equation 5.6 the process of Pauli-twirling adds 4 additional single-qubit gates per CNOT-gate. Even though the noise on single-qubit gates is an order of one magnitude lower than the noise on CNOT-gates in all the simulated cases, this seems to compound to a non-negligible noise influence.

Finally, we want to examine the drawbacks of this error mitigation scheme, which first and foremost is the computational cost. As described in Section 5.1.4, when increasing the number of noise amplified terms n , we might need to increase the number of circuit shots over all the noise amplified circuits in order to compensate for an increased statistical error in the mitigated expectation value.

Assume for the sake of simplicity that we execute each amplified circuit for the same number of shots, i.e., $N_i = N \forall i = 1, 2, \dots, n$. Note that this may not be optimal, but it serves to illustrate the general point of the computational scaling. We can then from Equation 5.14 estimate the number of shots per circuit needed to obtain a statistical error below a fixed error threshold ϵ_{tol} as

$$N_s^{(n)} = \frac{\sum_{i=1}^n (\gamma_i^{(n)})^2 \sigma_i^2}{\epsilon_s^2}, \quad (6.6)$$

where σ_i^2 is the variance in circuit i , which is noise amplified by a factor $r_i = 2i - 1$, and the $\gamma_i^{(n)}$ -coefficients are the coefficients found from Equation 5.4. These coefficients, for the cases from $n = 2$ to $n = 10$, is presented in Table 5.1.

To investigate whether the variance scaling behaves as expected, we repeat the error mitigation procedure for 1000 times with $n = 8$, each time with $N = 8192$ shots per circuit. For each separate run, the mitigated expectation value was computed using the Richardson extrapolation. This was done on a mocked simulator backend that emulates the IBMQ Vigo device [36]. The variance in the mitigated estimator $\tilde{E}_{N=8192}^{(n)}$ was estimated by the mean square error, over the 1000 repeats, as $\sigma_{\text{mit}}^2 = 0.284$. The variance in each of the noise amplified expectation value estimators $\tilde{E}(r_i \lambda)_{N=8192}$ was then estimated by the mean square error over the 1000 repeats. The estimated variances for $r_i = 2i - 1$, $i = 1, 2, \dots, 10$, is shown in Table 6.3. From Equation 5.12, the expected variance for the mitigated expectation value estimator $\tilde{E}_{N=8192}^{(n)}$ was then estimated as $\tilde{\sigma}_{\text{mit}}^2 = 0.285$. The expected variance calculated from the per-circuit variances is thus in close agreement with the measured variance in the mitigated expectation value estimator.

To understand the impact of the variance scaling, we estimated the variance in each noise amplified circuit for amplification factors $r_i = 2i - 1$, $i = 1, 2, \dots, 10$, for the SWAP-test circuit on a backend with noise from the IBMQ Vigo device. We executed each circuit for 8192 shots and estimated the variance by the mean square error. We then used Equation 6.6 to estimate the number

Table 6.3: Estimated variances σ_i^2 for the SWAP-test circuit when noise amplified by factors $r_i = 2i - 1$. The variances were estimated by the mean square error over 8192 circuit shots.

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$
σ_i^2	0.850	0.942	0.977	0.991	0.997
	$i = 6$	$i = 7$	$i = 8$	$i = 9$	$i = 10$
σ_i^2	0.999	0.999	0.998	1.00	1.00

Table 6.4: Estimated number of per-circuit shots $N_s^{(n)}$, and total number of shots $N^{(n)} = n \cdot N_s^{(n)}$, required to achieve an error in the estimator $\tilde{E}_N^{(n)}$ below the threshold of $\epsilon_{\text{tot}} = 0.01$. The noise amplification factors in all cases are $\{1, 3, 5, \dots, 2n - 1\}$.

	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n = 5$
$N_s^{(n)}$	$8.5 \cdot 10^3$	$2.15 \cdot 10^4$	$4.60 \cdot 10^4$	$1.04 \cdot 10^5$	$2.58 \cdot 10^5$
$N_{\text{tot}}^{(n)}$	$8.5 \cdot 10^3$	$4.30 \cdot 10^4$	$1.38 \cdot 10^5$	$4.14 \cdot 10^5$	$1.29 \cdot 10^6$
	$n = 6$	$n = 7$	$n = 8$	$n = 9$	$n = 10$
$N_s^{(n)}$	$7.11 \cdot 10^5$	$2.11 \cdot 10^6$	$6.61 \cdot 10^6$	$5.01 \cdot 10^6$	$1.83 \cdot 10^6$
$N_{\text{tot}}^{(n)}$	$4.27 \cdot 10^6$	$1.48 \cdot 10^7$	$5.29 \cdot 10^7$	$4.51 \cdot 10^7$	$1.83 \cdot 10^7$

of shots per circuit that is needed to achieve an error below a threshold of $\epsilon_{\text{tot}} = 0.01$. The estimated required number of per-circuit shots $N_s^{(n)}$, and the corresponding total required number of shots $N^{(n)} = n \cdot N_s^{(n)}$, is shown in Table 6.4 for different cases of n from 1 to 10.

From Table 6.4 it becomes clear that the zero-noise extrapolation comes with a significant increase in computational cost. From Table 6.3 we observe that the variance is not significantly altered between the various noise amplified circuits. Instead, we observe that the specific $\gamma_i^{(n)}$ -coefficients have a considerable impact on the error scaling, and thus on the required number of circuit shots. For $n = 5$ and above, the required number of shots have increased by more than 2 orders of magnitude.

From Figure 6.1, however, we observe that including more and more noise amplification terms gives diminishing returns. Significant improvements have already been achieved for $n = 2$ and $n = 3$, for which the shot requirements are within or approximately at one order of magnitude larger than that of the bare-circuit case. When considering the full noise model, the relative error in the mitigated expectation value is at 14% for $n = 2$ and 8.3% for $n = 3$, compared to 30% for the bare-circuit. For practical applications, including just a few noise amplified terms is the most realistic option.

6.2 Zero-Noise Extrapolation with Error Detection

The results so far from the zero-noise extrapolation promising. Further improvements may, however, be gained by combining the zero-noise extrapolation with other error mitigation techniques. One error mitigation scheme that may be straight-forwardly combined with the zero-noise extrapolation is the error detection scheme. This method is detailed in Section 5.2. This scheme was proposed in an article by McArdle, Yuan and Benjamin [15].

In the article, McArdle, Yuan and Benjamin explored the use of such a stabilizer-like error detection scheme combined with a zero-noise extrapolation scheme. They trialed this on the VQE algorithm for the H_2 molecule using an UCCSD ansatz. The VQE algorithm is described in Section 4.2. We will later in this chapter recreate the experiments performed by McArdle, Yuan and Benjamin, but using our proposed CNOT-repetition scheme for noise amplification. We will use this experiment as a vessel to explore the efficiency of the error detection and the zero-noise extrapolation techniques on a near-future applicable quantum algorithm. We will further explore the potential of combining the error detection and zero-noise extrapolation techniques, using the CNOT-repetition scheme.

6.2.1 VQE for the Hydrogen Molecule

In the Jordan-Wigner transformation each qubit stores the occupation number of a corresponding orbital. In this problem, we want to arrange $n = 2$ electrons in $m = 4$ orbitals in such a manner that the energy is minimized. The Hamiltonian in this transformation may be written as a linear combination of m -qubit Pauli-operator terms as

$$\mathcal{H} = \sum_i \alpha_i h_i, \quad (6.7)$$

where the α_i are real coefficients and h_i are 4-qubit tensor products of the Pauli-operators $\{\mathbb{I}, X, Y, Z\}$. The Pauli-operator terms with corresponding coefficients for the H_2 molecule, with $n = 2$ electrons and $m = 4$ orbitals, are given in Table A.1.

In the VQE procedure we prepare a parameterized ansatz state $|\psi(\vec{\theta})\rangle$ then measure the energy given by

$$E(\vec{\theta}) = \langle \psi(\vec{\theta}) | \mathcal{H} | \psi(\vec{\theta}) \rangle = \sum_i \alpha_i \langle \psi(\vec{\theta}) | h_i | \psi(\vec{\theta}) \rangle. \quad (6.8)$$

As the Hamiltonian terms h_i are tensor products of Pauli-operators, its contribution $\langle \psi(\vec{\theta}) | h_i | \psi(\vec{\theta}) \rangle$ can be estimated by applying single-qubit gates to the end state $|\psi(\vec{\theta})\rangle$, then performing measurements in the computational basis.

We thus prepare one circuit for each Hamiltonian term h_i , estimate the corresponding expectation value $\langle \psi(\vec{\theta}) | h_i | \psi(\vec{\theta}) \rangle$, then compute the resulting energy expectation value from Equation (6.8).

An advantage of trialing these methods on the problem of molecular simulations is that it gives an obvious milestone to aim for; the chemical accuracy. The chemical accuracy of $1.6 \cdot 10^{-3}$ Hartree is the accuracy needed in energy computations in order to predict reaction rates at room temperature to within an order of magnitude [15].

6.2.2 UCCSD Ansatz Circuit with Error Detection

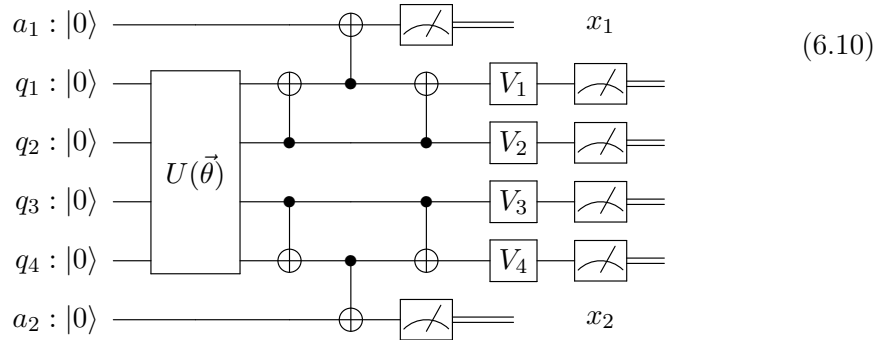
The UCCSD ansatz for the H_2 molecule, as discussed in Section 4.2.3, prepares the parameterized state

$$|\psi^{H_2}(\vec{\theta})\rangle = U(\vec{\theta})|0000\rangle = \alpha|0101\rangle + \beta|1010\rangle + \gamma|0110\rangle + \delta|1001\rangle, \quad (6.9)$$

where $\alpha, \beta, \gamma, \delta$ are complex coefficients dependent on the variational parameters $\vec{\theta}$, and the unitary $U(\vec{\theta})$ is applied by use of a quantum circuit. The quantum circuit for the UCCSD ansatz is shown in Equation (A.1).

From Equation (6.9) we observe that the state $|\psi^{H_2}(\vec{\theta})\rangle$ is a simultaneous eigenstate of the operators Z_1Z_2 and Z_3Z_4 with eigenvalue -1 . Here, Z_i is the Z -operator acting on qubit q_i . This suggests the following error detection scheme; by use of ancillary qubit measurements, measure the operators Z_1Z_2 and Z_3Z_4 . Store the measurement output in the bits $x_1 \in \{-1, +1\}$ and $x_2 \in \{-1, +1\}$. We complete the circuit with measurements on the state $|\psi^{H_2}(\vec{\theta})\rangle$. Afterwards, we inspect the ancillary measurements. If $(x_1, x_2) = (-1, -1)$ then we have detected no error and we keep the measurement results. Otherwise, we have detected at least one error and we discard the measurements.

The full circuit that implements the UCCSD ansatz with error detection is given as



where the subcircuit implementing the parametrized unitary $U(\vec{\theta})$ is given by Equation (A.1). For a quantum processor with linear qubit connectivity this circuit is compatible as is, requiring no additional SWAP-gates. The V_i -gates implement the single-qubit gates that required to measure the corresponding

Hamiltonian Pauli-operator h_i . The procedure for measuring arbitrary Pauli-operators is described in Section 2.4.9.

The ansatz subcircuit that implements $U(\vec{\theta})$ has a CNOT complexity of 52 after optimization by the `qiskit` transpiler with the heaviest optimization preset and assuming linear qubit connectivity. The error detection sub-circuit contributes 6 additional noisy CNOT-gates. This is comparably small to the CNOT-contribution of the ansatz circuit.

More efficient ansätze that have achieved more accurate results do exist for the H_2 molecule [5, 6]. This is made possible by taking advantage of particular symmetries and simplifications possible for the H_2 molecule, but this is not generally applicable to larger molecules. The UCCSD ansatz was therefore used by McArdle et. al. to ensure general applicability to quantum chemistry.

6.2.3 Experiments

The VQE procedure was first run using a noiseless simulator in order to obtain a set of optimized variational parameters $\vec{\theta}_{\text{opt}}$. This was done as we are not looking at the effects of noise on the optimization procedure. We then used the UCCSD ansatz circuit with optimized parameters to compute the energy on a simulator with a noise model. A 2-qubit depolarizing noise channel with error rate p , as shown in Equation (3.8), was applied to all CNOT-gates. In these experiments no single-qubit or measurement noise was added.

Each experiment was repeated with four different combinations of error mitigation techniques; no error mitigation, error detection, zero-noise extrapolation and zero-noise extrapolation combined with error detection.

This experiment was run for the H_2 molecule for interatomic distances a , ranging from $a = 0.3 \text{ \AA}$ to $a = 2.5 \text{ \AA}$. A 2-qubit depolarizing noise channel with error rate $p = 0.1\%$ was applied to CNOT-gates. The results are shown in Figure 6.3.

As seen from Figure 6.3 the error detection scheme gives improvements over the bare noisy results. The zero-noise extrapolation technique, however, give far better results.

The absolute errors from the above experiment is shown in Figure 6.4 compared to the chemical accuracy of $1.6 \cdot 10^{-3}$ Hartree. From this, we observe that for an error rate of $p = 0.1\%$ the zero-noise extrapolation does obtain an error below the chemical accuracy. The variance, however, makes it unclear in this experiment if the error detection scheme may give any improvements when combined with the zero-noise extrapolation, as it was approximately equal to the chemical accuracy for both of these schemes.

We repeated the experiment with H_2 at a fixed distance of $a = 0.74 \text{ \AA}$. Now, we applied a similar 2-qubit depolarizing noise channel on CNOT gates, but with different error rates ranging from $p = 0.1\%$ to $p = 2\%$. The error in the energy as compared to the exact ground state energy of $E = -1.137$ Hartree is shown in Figure 6.5.

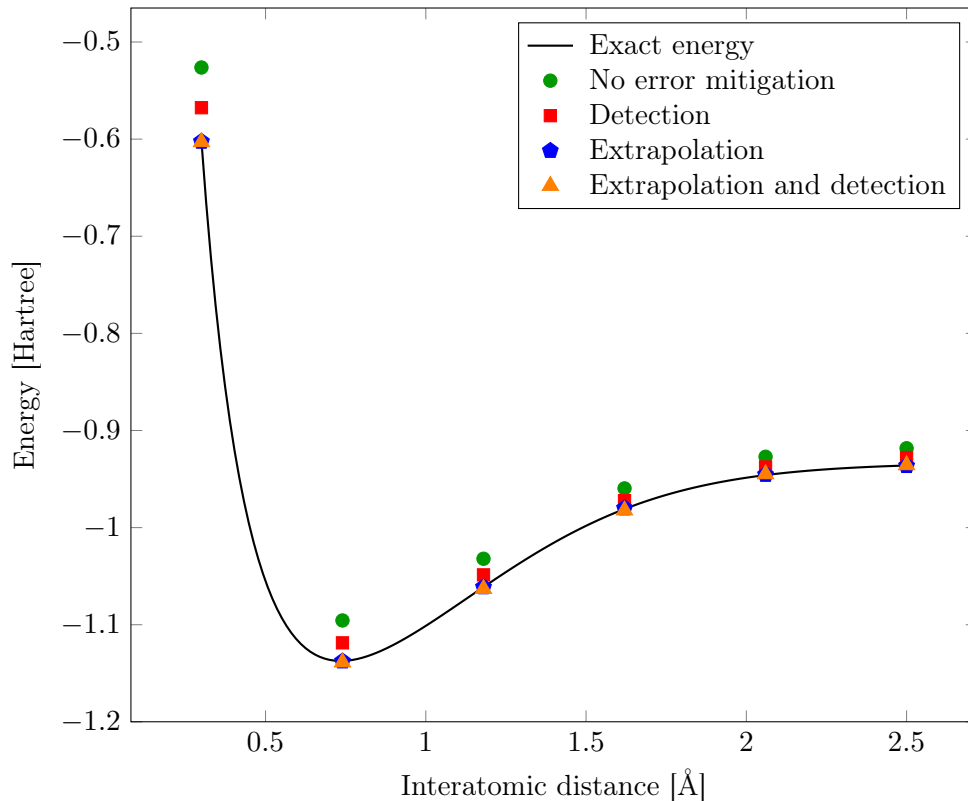


Figure 6.3: Ground state energies from VQE simulations on H_2 when applying different methods of error mitigation, with the H_2 molecule at different interatomic distances. The UCCSD ansatz was used for the VQE procedure.

From Figure 6.5 we again observe that we get improvements over the bare noisy results when employing the error detection technique. Even further improvements are obtained by the zero-noise extrapolation. Now, however, we observe that combining the two techniques yields even further improvements.

For $p = 0.1\%$, the error in both extrapolation results are within the chemical accuracy of the ideal result. A CNOT-error rate in this order may be feasible for near-future quantum computers [39, 40]. On the IBMQ devices that we have previously considered, the CNOT error rate is however stated to be approximately $p = 1\%$ on average [27]. This applies to the IBMQ Athens and IBMQ Belem devices.

As discussed in Section 5.2 a drawback of the error detection technique is that we may potentially end up discarding a large portions of circuit shots that we perform. In that case we would need to perform additional shots of the quantum circuit in order to obtain an estimated expectation value that is still converged to within a reasonable error. To examine this, the rate of total measurement experiments that was discarded when employing the error detection scheme to the noise amplified UCCSD circuit is shown in Table 6.5.

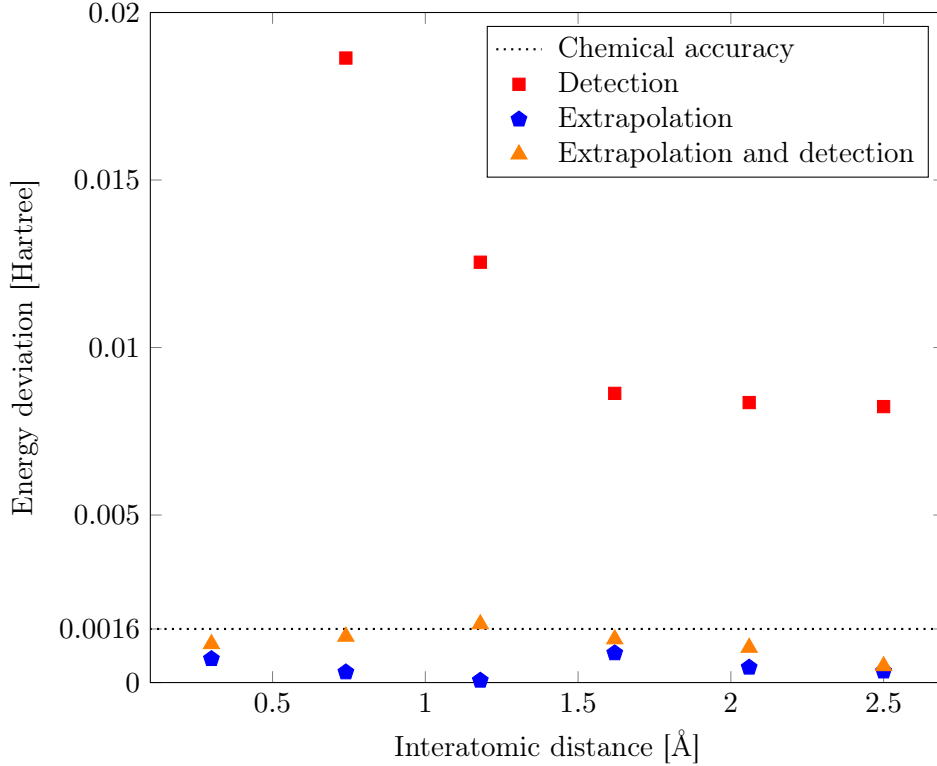


Figure 6.4: Error in the ground state energy for H_2 as computed by the VQE, when applying different methods of error mitigation. The H_2 molecule was set to different interatomic distances. The UCCSD ansatz was used, and a 2-qubit depolarizing noise channel was applied to CNOT-gates in the quantum computation. The simulations were stopped at a variance approximately equal to the chemical accuracy.

Table 6.5: Rate of discarded measurements for the error detection scheme on the UCCSD ansatz circuit for H_2 . The discarded rates are shown for different CNOT-noise error rates p , and for different noise amplification factors r when amplifying the noise with the CNOT-repetition scheme.

	$p = 0.1\%$	$p = 0.5\%$	$p = 1\%$	$p = 1.5\%$	$p = 2\%$
$r = 1$	5.9%	25%	36%	52%	54%
$r = 3$	16%	47%	62%	72%	72%
$r = 5$	25%	59%	70%	73%	74%

This is shown for the different error rates from $p = 0.1\%$ to $p = 2\%$, and for noise amplification factors $r = 1$, $r = 3$ and $r = 5$.

We observe from Table 6.5 that the scheme discards only a small percentage of the total circuit shots for the lower error rates. At $p = 2\%$ we discard 54% of shots, which would require us to perform less than 3 times the original shot count to obtain the same error in the expectation value estimation. These

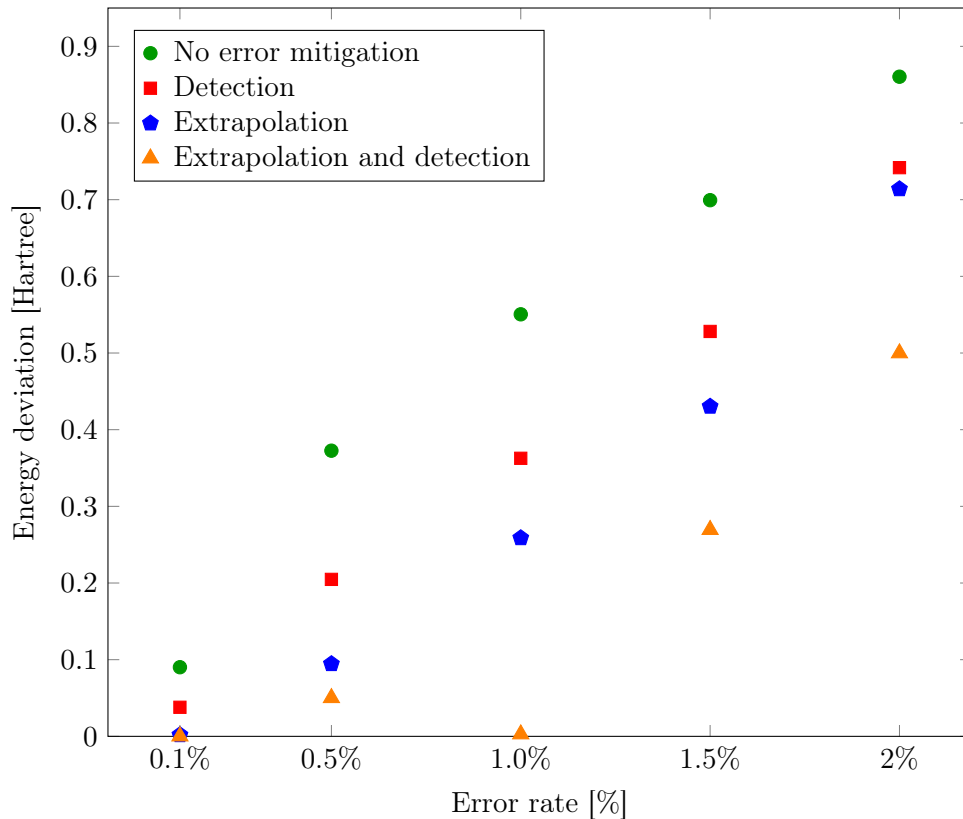


Figure 6.5: Deviation from the exact energy for noisy VQE simulations of H_2 at different error rates and with different error mitigation schemes. The interatomic distance is $a = 0.74\text{\AA}$ and the UCCSD ansatz have been used.

results suggest that while the error detection technique give limited error mitigation, failing to achieve the chemical accuracy at $p = 0.1\%$, the technique comes at a significantly lesser computational cost than the zero-noise extrapolation technique. The trade-off is that this scheme requires some additional ancillary qubits for the symmetry measurements.

For the noise amplified circuits, as expected, the rate of discarded measurements increased as the effective error rates on CNOTs is increased. On the circuit with a noise amplification factor of $r = 5$, with $p = 0.1\%$, we observe that 25% of the shots are discarded. This circuit should have an effective error rate of $\tilde{p} = 5 \cdot 0.1\% = 0.5\%$, and indeed, we find the bare circuit run with an error rate of 0.5% to also discard exactly 25% of the circuit shots. For the circuit with $r = 5$, run with an error rate of $p = 2\%$, we observe that 74% of the measurements are discarded. To compensate for this, we would need to perform 4 times the original shot count, still below one order of magnitude. This suggests that the lesser cost of the error detection scheme also holds for higher effective error rates, and thus still holds when combining the scheme

with the zero-noise extrapolation.

Chapter 7

Conclusion

In this thesis, we have studied some techniques relevant for near-future quantum computation. In particular, we have examined techniques of quantum error mitigation, and their applications to NISQ-era viable hybrid quantum-classical algorithms. We studied the zero-noise extrapolation technique and proposed a variant with a novel noise amplification scheme, as well as the error detection scheme. These techniques were applied to a trial quantum circuit in the SWAP-test circuit, and to the VQE algorithm for molecular simulations, in order to explore how error mitigation may improve results in near-future quantum algorithms.

7.1 Conclusions

In order to test its efficiency, the zero-noise extrapolation technique with our proposed noise amplification scheme by CNOT-repetition was tested on the SWAP-test circuit. A noise model emulating noise in the IBMQ Vigo quantum device was applied. The error mitigated expectation value, in the best case, obtained a mitigated expectation value within a relative error of 3.3% of the ideal-case expectation value when considering only noise on CNOT-gates, and 4.8% when considering noise also on single-qubit gates and measurements. When compared to the bare-circuit error of 30%, this constitutes an improvement of one order of magnitude.

We further compared the zero-noise extrapolation when using our proposed CNOT-repetition scheme for noise amplification, with the alternative scheme that uses random Pauli-gate sampling. This was done for several different noise models. As expected, the two schemes gave comparable results for the depolarizing noise channel, as the random-Pauli sampling scheme relies on approximating the noise as stochastic Pauli-noise. The CNOT-repetition scheme relies on no such assumptions, and as expected, produced better results for the other noise models. The CNOT-repetition scheme gave an improvement in the relative error by one order of magnitude for the amplitude damping and phase damping schemes, and by more than a factor of 2 for the IBMQ Athens noise

model. For the random Pauli-sampling scheme, the process of Pauli-twirling did not significantly improve results for any of the noise models, suggesting that the noise from single-qubit gates was more significant than the advantages gained from the transformation of the noise models into stochastic Pauli-noise channels.

The zero-noise extrapolation technique, however, was found to come with a significant computational cost that is highly dependent on the specific $\gamma_i^{(n)}$ -coefficients. The mitigated expectation value estimator is afflicted by a large variance, dependent on the specific $\gamma_i^{(n)}$ -coefficients and the per-circuit variances, causing an increased statistical error that must be compensated by increasing the number of per-circuit shots. While the mitigated expectation value was found to converge towards a lower error when increases the number of noise amplified terms n , the required total number of shots increased by several orders of magnitude for larger n . The most realistic use cases are the cases of $n = 2$ and $n = 3$, where the shot requirements increased by approximately one order of magnitude, while still gaining significant improvements in terms of noise mitigation. The relative errors found for the SWAP-test circuit with $n = 2$ and $n = 3$, respectively, was 14% and 8.3%, respectively, compared to 30% for the bare-circuit case.

We further tested the error detection scheme combined with the zero-noise extrapolation, with our proposed noise amplification scheme, on the VQE algorithm for the H_2 molecule. This was done with several goals in mind; to explore the error detection technique, to examine the potential in combining both these error mitigation techniques, and to test state-of-the-art error mitigation schemes on an algorithm that may soon be applicable to real-world problems. We run these experiments on a simulator with a depolarizing noise model, with several different error rates, and tested both the error detection and zero-noise extrapolation techniques by themselves, as well as a combined error mitigation scheme involving both techniques.

We ran the VQE experiment on the H_2 molecule with several different interatomic distances, with an error rate on CNOT-gates of $p = 0.1\%$. While the error detection scheme gave a significant improvement over the bare-circuit results, it failed to achieve errors within the chemical accuracy of $1.6 \cdot 10^{-3}$ Hartree by itself. Both the zero-noise extrapolation and the combined scheme, however, achieved errors around the benchmark of the chemical accuracy.

The experiment was repeated at a fixed interatomic distance for error rates ranging from $p = 0.1\%$ to $p = 2\%$. For all error rates, both error mitigation schemes gave significant improvements over the bare-circuit expectation value. For all cases, the zero-noise extrapolation outperformed the error detection scheme, and the combined zero-noise extrapolation with error detection outperformed both. Errors in the range of the chemical accuracy, however, were only achieved in the case of $p = 0.1\%$.

The error detection scheme on its own fails to achieve errors below the chemical accuracy, even for $p = 0.1\%$. However, it was found to come at a

considerable lesser computational cost than the zero-noise extrapolation. At $p = 0.1\%$, only 5.9% of the circuit shots were discarded.

The noise amplification procedure inherently increases the error rates and therefore the amount of discarded measurements also increased. For the noise amplified circuit with amplification factor $r = 5$, at $p = 2\%$, 74% of the circuit shots, which require 4 times the original shot count. This is still below one order of magnitude and suggests that it is still computationally cheap to run the error detection scheme even when combined with noise amplification. The trade-off is the additional required ancillary qubits, needed for the symmetry measurements, which in this case is 2 additional qubits.

7.2 Future Work

We have tested two different quantum error mitigation techniques with promising results. Future work on this topic includes verifying these results on real quantum hardware. The noise models provided by qiskit that aims to emulate the noise in specific quantum devices are quite simple and might not necessarily be realistically recreating complicated noise present on physical quantum hardware. This remains to be tested.

Furthermore, work on the zero-noise extrapolation algorithm includes a scheme for per-circuit adjustments of the circuit shots, taking the specific γ_i -coefficient and variance for each circuit into account. We looked at the case where each circuit was executed for the same number of shots, but this might not be optimal, as the γ_i -coefficients vary by several orders of magnitude for different amplification factors. Improvements in this aspect might mitigate some of the computational cost brought by the zero-noise extrapolation.

Bibliography

- [1] F. Arute, K. Arya, R. Babbush and et al., ‘Quantum supremacy using a programmable superconducting processor,’ *Nature*, vol. 574, pp. 505–510, 2019. DOI: [10.1038/s41586-019-1666-5](https://doi.org/10.1038/s41586-019-1666-5).
- [2] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information, 10th Anniversary Edition*. Cambridge University Press, 2016, ISBN: 978-1-107-00217-3.
- [3] P. K. Barkoutsos, J. F. Gonthier, I. Sokolov, N. Moll, G. Salis, A. Fuhrer, M. Ganzhorn, D. J. Egger, M. Troyer, A. Mezzacapo, S. Filipp and I. Tavernelli, ‘Quantum algorithms for electronic structure calculations: Particle-hole hamiltonian and optimized wave-function expansions,’ *Phys. Rev. A*, vol. 98.2, p. 022322, 2018. DOI: [10.1103/PhysRevA.98.022322](https://doi.org/10.1103/PhysRevA.98.022322).
- [4] H. R. Grimsley, S. E. Economou, E. Barnes and N. J. Mayhall, ‘An adaptive variational algorithm for exact molecular simulations on a quantum computer,’ *Nat. Commun.*, vol. 10, p. 3007, 2019.
- [5] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow and J. M. Gambetta, ‘Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets,’ *Nature*, vol. 549, pp. 242–246, 2017. DOI: [10.1038/nature23879](https://doi.org/10.1038/nature23879).
- [6] B. T. Gard, L. Zhu, G. S. Barron, N. J. Mayhall, S. E. Economou and E. Barnes, ‘Efficient symmetry-preserving state preparation circuits for the variational quantum eigensolver algorithm,’ *npj Quantum Inf*, vol. 6, p. 10, 2020. DOI: [10.1038/s41534-019-0240-1](https://doi.org/10.1038/s41534-019-0240-1).
- [7] P. W. Shor, ‘Scheme for reducing decoherence in quantum computer memory,’ *Phys. Rev. A*, vol. 52.4, R2493–R2496, 1995. DOI: [10.1103/PhysRevA.52.R2493](https://doi.org/10.1103/PhysRevA.52.R2493).
- [8] D. Gottesman, ‘Stabilizer codes and quantum error correction,’ *arXiv preprint arXiv:quant-ph/9705052*, 1997. [Online]. Available: <https://arxiv.org/pdf/quant-ph/9705052.pdf>.
- [9] C. Gidney and M. Ekerå, ‘How to factor 2048 bit rsa integers in 8 hours using 20 million noisy qubits,’ *arXiv preprint arXiv:1905.09749*, 2019.
- [10] E. Farhi and J. Goldstone, ‘A quantum approximate optimization algorithm,’ *arXiv preprint arXiv:1411.4028*, 2014.

- [11] F. Hadlock, ‘Finding a maximum cut of a planar graph in polynomial time,’ *SIAM J. Comput.*, vol. 4, pp. 221–225, 1973. DOI: [10.1137/0204019](https://doi.org/10.1137/0204019).
- [12] C.-C. Chen, S.-Y. Shiao, M.-F. Wu and Y.-R. Wu, ‘Hybrid classical-quantum linear solver using noisy intermediate-scale quantum machines,’ *Sci Rep*, vol. 9, p. 16251, 2019. DOI: [10.1038/s41598-019-52275-6](https://doi.org/10.1038/s41598-019-52275-6).
- [13] K. Temme, S. Bravyi and J. M. Gambetta, ‘Error mitigation for short-depth quantum circuits,’ *Phys. Rev. Lett.*, vol. 119, p. 180509, 2017. DOI: [10.1103/PhysRevLett.119.180509](https://doi.org/10.1103/PhysRevLett.119.180509).
- [14] S. Endo, S. C. Benjamin and Y. Li, ‘Practical quantum error mitigation for near-future applications,’ *Phys. Rev. X*, vol. 8, p. 031027, 2018. DOI: [10.1103/PhysRevX.8.031027](https://doi.org/10.1103/PhysRevX.8.031027).
- [15] S. McArdle, X. Yuan and S. Benjamin, ‘Error-mitigated digital quantum simulation,’ *Phys. Rev. Lett.*, vol. 122, p. 180501, 2019. DOI: [10.1103/PhysRevLett.122.180501](https://doi.org/10.1103/PhysRevLett.122.180501).
- [16] X. Bonet-Monroig, R. Sagastizabal, M. Singh and T. O’Brien, ‘Low-cost error mitigation by symmetry verification,’ *Phys. Rev. A*, vol. 98, p. 062339, 2018. DOI: [10.1103/PhysRevA.98.062339](https://doi.org/10.1103/PhysRevA.98.062339).
- [17] A. Kandala, K. Temme, A. D. Córcoles, A. Mezzacapo, J. M. Chow and J. M. Gambetta, ‘Extending the computational reach of a noisy superconducting quantum processor,’ *arXiv preprint arXiv:1801.00862*, 2018.
- [18] J. Sun, X. Yuan, T. Tsunoda, V. Vedral, S. C. Benjamin and S. Endo, ‘Mitigating realistic noise in practical noisy intermediate-scale quantum devices,’ *arXiv preprint arXiv:2001.04891*, 2020.
- [19] V. N. Premakumar and R. Joynt, ‘Error mitigation in quantum computers subject to spatially correlated noise,’ *arXiv preprint arXiv:1812.07076*, 2018.
- [20] Y. Li and S. C. Benjamin, ‘Efficient variational quantum simulator incorporating active error minimisation,’ *Phys. Rev. X*, vol. 7, p. 021050, 2017. DOI: [10.1103/PhysRevX.7.021050](https://doi.org/10.1103/PhysRevX.7.021050).
- [21] P. Suchsland, F. Tacchino, M. Fischer, T. Neupert, P. Baskoutsos and I. Tavernelli, ‘Algorithmic error mitigation scheme for current quantum processors,’ *arXiv preprint arXiv:1612.02058*, 2020.
- [22] F. Maciejewski, Z. Zimboras and M. Oszmaniec, ‘Mitigating measurement errors in quantum computers by exploiting state-dependent bias,’ *Quantum*, vol. 4, p. 257, 2020. DOI: [10.22331/q-2020-04-24-257](https://doi.org/10.22331/q-2020-04-24-257).
- [23] F. G. Fuchs, V. Falch and C. Johnsen, ‘Quantum poker—a game for quantum computers suitable for benchmarking error mitigation techniques on nisq devices,’ *The European Physical Journal Plus*, vol. 135, p. 353, 2020. DOI: [10.1140/epjp/s13360-020-00360-5](https://doi.org/10.1140/epjp/s13360-020-00360-5).

- [24] J. B. Fraleigh, *A First Course in Abstract Algebra, 7th Edition*. Pearson Education Limited, 2013, ISBN: 9781292024967.
- [25] P. C. Hemmer, *Kvantemekanikk*. Tapir akademisk forlag, 2005, ISBN: 9788251920285.
- [26] T. E. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to Algorithms, Third edition*. The MIT Press, 2009, ISBN: 978-0-262-53305-8.
- [27] IBM. (2020). ‘Ibm quantum experience.’ Accessed: 11.11.2020, [Online]. Available: <https://quantum-computing.ibm.com/>.
- [28] D. Gottesman and S. Aaronson, ‘Improved simulation of stabilizer circuits,’ *Phys. Rev. A*, vol. 70, p. 052328, 2004. DOI: [10.1103/PhysRevA.70.052328](https://doi.org/10.1103/PhysRevA.70.052328).
- [29] T. Itoko, R. Raymond, T. Imamichi and A. Matsuo, ‘Optimization of quantum circuit mapping using gate transformation and commutation,’ *arXiv preprint arXiv:1907.02686*, 2019.
- [30] C. J. Ballance, T. P. Harty, N. M. Linke, M. A. Sepiol and D. M. Lucas, ‘High-fidelity quantum logic gates using trapped-ion hyperfine qubits,’ *Phys. Rev. Lett.*, vol. 117.6, p. 060504, 2016. DOI: [10.1103/PhysRevLett.117.060504](https://doi.org/10.1103/PhysRevLett.117.060504).
- [31] P. W. Shor, ‘Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer,’ *SIAM J. Comput.*, vol. 26.5, pp. 1484–1509, 2006. DOI: [10.1137/S0097539795293172](https://doi.org/10.1137/S0097539795293172).
- [32] J. T. Seeley, M. J. Richard and P. J. Love, ‘The bravyi-kitaev transformation for quantum computation of electronic structure,’ *J. Chem. Phys.*, vol. 137, p. 224109, 2012. DOI: [10.1063/1.4768229](https://doi.org/10.1063/1.4768229).
- [33] A. Tranter, P. J. Love, F. Mintert, N. Wiebe and P. V. Coveney, ‘Ordering of trotterization: Impact on errors in quantum simulation of electronic structure,’ *Entropy*, vol. 21, 2019. DOI: [10.3390/e21121218](https://doi.org/10.3390/e21121218).
- [34] L. F. Richardson, ‘The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam,’ *Phil. Trans. R. Soc. Lond. A*, vol. 210, pp. 307–357, 1910. DOI: [10.1098/rsta.1911.0009](https://doi.org/10.1098/rsta.1911.0009).
- [35] Z. Cai and S. Benjamin, ‘Constructing smaller pauli twirling sets for arbitrary error channels,’ *arXiv preprint arXiv:1807.04973*, 2018.
- [36] IBM. (2020). ‘Qiskit documentation.’ Accessed: 11.11.2020, [Online]. Available: <https://qiskit.org/documentation/>.
- [37] H. Abraham, A. Offei, R. Agarwal *et al.*, *Qiskit: An open-source framework for quantum computing*, 2019. DOI: [10.5281/zenodo.2562110](https://doi.org/10.5281/zenodo.2562110).

- [38] J. R. McClean, K. J. Sung, I. D. Kivlichan, Y. Cao, C. Dai, E. S. Fried, C. Gidney, B. Gimby, P. Gokhale, T. Häner, T. Hardikar, V. Havlíček, O. Higgott, C. Huang, J. Izaac, Z. Jiang, X. Liu, S. McArdle, M. Neeley, T. O'Brien, B. O'Gorman, I. Ozfidan, M. D. Radin, J. Romero, N. Rubin, N. P. D. Sawaya, K. Setia, S. Sim, D. S. Steiger, M. Steudtner, Q. Sun, W. Sun, D. Wang, F. Zhang and R. Babbush, 'Openfermion: The electronic structure package for quantum computers,' 2019.
- [39] C. J. Ballance, T. P. Harty, N. M. Linke, M. A. Sepiol and D. M. Lucas, 'High-fidelity quantum logic gates using trapped-ion hyperfine qubits,' *Phys. Rev. Lett.*, vol. 117, p. 060 504, 2016. DOI: [10.1103/PhysRevLett.117.060504](https://doi.org/10.1103/PhysRevLett.117.060504).
- [40] J. P. Gaebler, T. R. Tan, Y. Lin, Y. Wan, R. Bowler, A. C. Keith, S. Glancy, K. Coakley, E. Knill, D. Leibfried and D. J. Wineland, 'High-fidelity universal gate set for $^9\text{Be}^+$ ion qubits,' *Phys. Rev. Lett.*, vol. 117, p. 060 505, 2016. DOI: [10.1103/PhysRevLett.117.060505](https://doi.org/10.1103/PhysRevLett.117.060505).

Appendix A

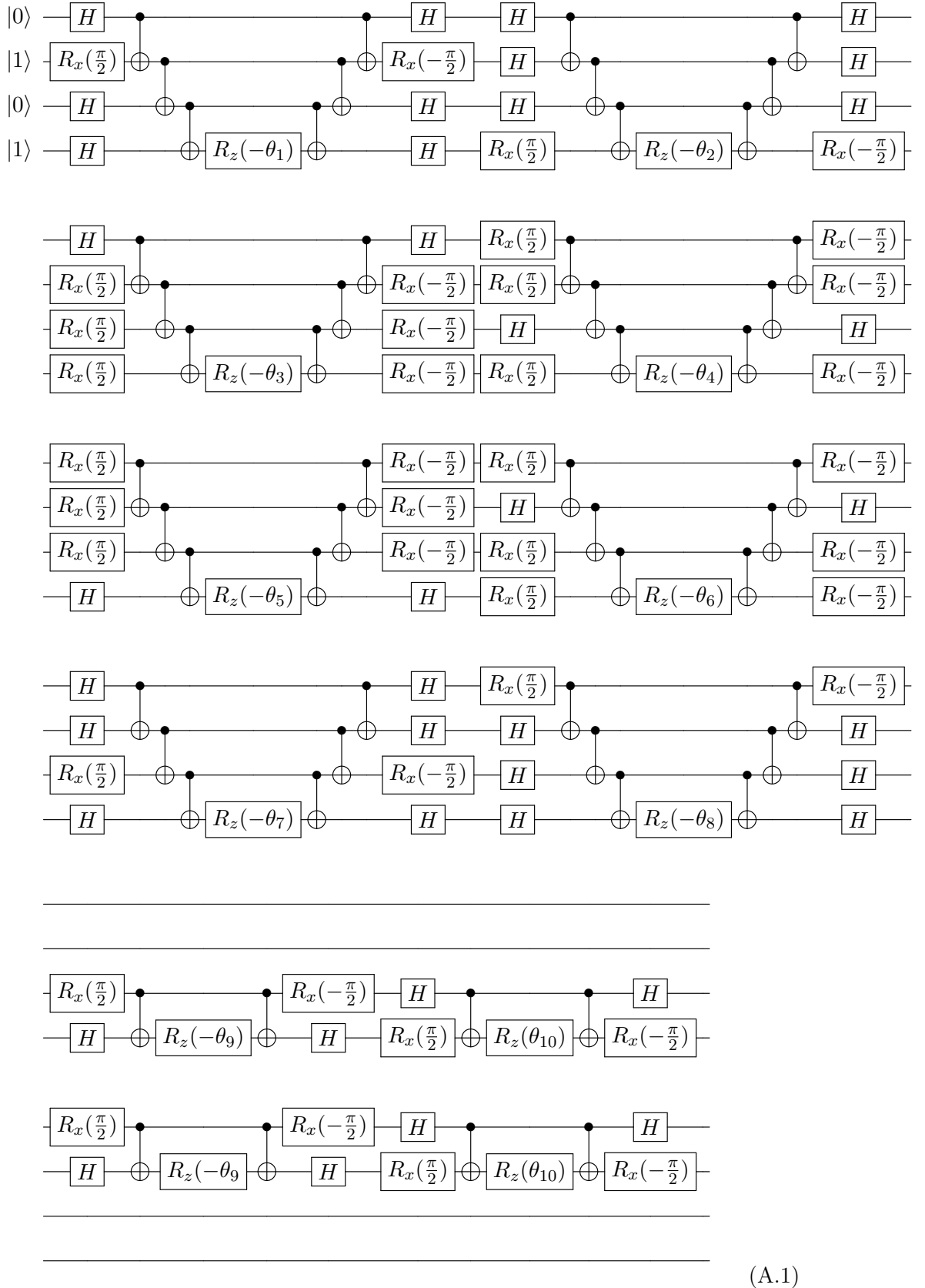
The H₂ Molecule

This chapter presents some specifics for the H₂ molecule relevant for the VQE algorithm, which is detailed in Section 4.2. The Pauli-terms P_α and coefficients h_α , as given by Equation (4.6), for the qubit Hamiltonian of the H₂-molecule is presented in TableA.1. Here, the Jordan-Wigner transformation was used to transform the fermionic Hamiltonian into a qubit Hamiltonian.

Table A.1: Pauli-operator terms P_α and corresponding coefficients h_α for a qubit-Hamiltonian for the H₂ molecule at an interatomic distance of $a = 0.74$ Å. The Jordan-Wigner transformation was used, and $m = 4$ electron orbitals was included.

h_i	α_i
<i>IIIZ</i>	0.1714
<i>IIZI</i>	-0.2234
<i>IZII</i>	0.1714
<i>ZIII</i>	-0.2234
<i>IIZZ</i>	0.1206
<i>IZIZ</i>	0.1687
<i>ZIIZ</i>	0.1659
<i>IZZI</i>	0.1659
<i>ZIZI</i>	0.1744
<i>ZZII</i>	0.1206
<i>XXYY</i>	0.04530
<i>YYXX</i>	0.04530
<i>YYYY</i>	0.04530
<i>XXXX</i>	0.04530

The following circuit gives the UCCSD ansatz, as described in Section 4.2.3, for the H₂ molecule when including $m = 4$ orbitals.



(A.1)

The two states initialized in the $|1\rangle$ state may be prepared by initialization in the $|0\rangle$ state followed by a single Pauli- X gate.

Appendix B

Pauli-Twirling

The process of *Pauli-twirling* transforms general error channels into stochastic Pauli-type error channels, which in certain scenarios may be easier to handle, by use of redundant Pauli gates.

By Equation (3.2), we can write the action of a faulty gate as the quantum channel

$$\tilde{\mathcal{N}} = \mathcal{N} \circ \mathcal{E}. \quad (\text{B.1})$$

Here, \mathcal{E} is the quantum channel of the ideal, noiseless gate and \mathcal{N} is an effective noise channel. By Equation (2.34) we can write the quantum channel \mathcal{N} in terms of its Kraus operators $\{K_l\}_l$, such that $\mathcal{N}(\rho) = \sum_l K_l \rho K_l^\dagger$.

We focus now on 2-qubit gates, e.g., the CNOT-gate. Assume that we perform a faulty 2-qubit gate with quantum channel $\tilde{\mathcal{U}}$. From Equation (3.2) we may write $\tilde{\mathcal{U}} = \mathcal{N} \circ \mathcal{U}$, where \mathcal{U} is the noiseless unitary gate $\mathcal{U}(\rho) = U \rho U^\dagger$ and $\mathcal{N}(\rho) = \sum_l K_l \rho K_l^\dagger$ is the noise operation.

The Kraus operators for a quantum channel on a 2-level system can be expanded in terms of the Pauli operators as

$$E_k = \sum_{i,j=0}^3 \alpha_{k;i,j} \sigma^i \otimes \sigma^j \quad (\text{B.2})$$

where $\sigma^0 = \mathbb{I}$, the identity, and σ^i for $i = 1, 2, 3$ are the Pauli matrices such that $\sigma^1 = \sigma^x$, $\sigma^2 = \sigma^z$ and $\sigma^3 = \sigma^y$, and $\alpha_{k;i,j} \in \mathbb{C}$ are complex coefficients [20].

To Pauli-twirl the faulty gate we choose two random Pauli-operators $a, b \in \{\mathbb{I}, X, Z, Y\}$. From a and b we determine Pauli operators c and d such that

$$(a \otimes b)U(c \otimes d) = e^{i\alpha}U, \quad (\text{B.3})$$

for the noiseless unitary gate U , where $e^{i\alpha}$ is some global phase that may be ignored. The operators c and d can be determined as

$$c \otimes d = U^\dagger(a \otimes b)U. \quad (\text{B.4})$$

The Pauli-twirled noisy gate can now be written as

$$\begin{array}{c} \text{---} \boxed{a} \text{---} \\ \text{---} \boxed{b} \text{---} \end{array} \boxed{\tilde{U}} \begin{array}{c} \text{---} \boxed{c} \text{---} \\ \text{---} \boxed{d} \text{---} \end{array} = \begin{array}{c} \text{---} \boxed{a} \text{---} \\ \text{---} \boxed{b} \text{---} \end{array} \boxed{U} \begin{array}{c} \text{---} \boxed{c} \text{---} \\ \text{---} \boxed{d} \text{---} \end{array} \boxed{\mathcal{N}} \begin{array}{c} \text{---} \boxed{c} \text{---} \\ \text{---} \boxed{d} \text{---} \end{array} = \begin{array}{c} \text{---} \boxed{a} \text{---} \\ \text{---} \boxed{b} \text{---} \end{array} \boxed{U} \begin{array}{c} \text{---} \boxed{c} \text{---} \\ \text{---} \boxed{d} \text{---} \end{array} \boxed{\mathcal{N}} \begin{array}{c} \text{---} \boxed{c} \text{---} \\ \text{---} \boxed{d} \text{---} \end{array} \quad (\text{B.5})$$

Note that the gate \mathcal{N} here represents the noise operation $\mathcal{N}(\rho) = \sum_l K_l \rho K_l^\dagger$, ρ being the density operator of the ingoing multi-qubit state, which is possibly not unitary.

The Pauli-twirled error channel is now described by the Kraus operators

$$\tilde{K}_l = (a \otimes b) \left(\sum_{i,j=0}^3 \alpha_{l;i,j} \sigma^i \otimes \sigma^j \right) (a \otimes b) = \sum_{i,j=0}^3 \alpha_{l;i,j} a \sigma^i a \otimes b \sigma^j b. \quad (\text{B.6})$$

The effective Pauli-twirled error channel can then be written as a stochastic Pauli-error channel

$$\tilde{\mathcal{N}}(\rho) = \sum_l \tilde{K}_l \rho \tilde{K}_l^\dagger = F \rho + \sum_{\substack{i,j=0 \\ (i,j) \neq (0,0)}}^3 \epsilon_{i,j} \sigma^i \sigma^j \rho \sigma^j \sigma^i, \quad (\text{B.7})$$

where $F = \sum_k |\alpha_{k;0,0}|^2$ is the fidelity of the Pauli-twirled error channel and the $\epsilon_{i,j} = \sum_k |\alpha_{k;i,j}|^2$, for $(i,j) \neq (0,0)$, are the error probabilities corresponding to the error operator $\sigma^i \sigma^j$ [20].

The process of Pauli-twirling requires applications of ideal single-qubit operators. Therefore, this technique is limited to cases where the noise in the single-qubit gates are negligible compared to the noise in the Pauli-twirled gate. In modern quantum hardware this is often the case for multi-qubit gates, such as the CNOT-gate, that typically have error rates an order of magnitude larger than that of single-qubit gates [30].

Appendix C

Quantum Error Correction

Quantum error correction using quantum error-correcting codes provides means to detect and exactly correct errors during a quantum computation by making use of *redundancy*. By encoding virtual, or *logical*, qubits into states on multiple physical qubits, errors that move the states out of the assigned code subspace can be detected by doing measurements. Based on the measurement outcomes, recovery operations are then done to recover the original logical qubit state.

C.1 The Classical Repetition code

To illustrate the concept, first consider the classical 3-bit repetition code with syndrome decoding. In this code, a bit $x \in \{0, 1\}$ is encoded into an encoded *logical* bit according to

$$0 \rightarrow 000, 1 \rightarrow 111. \quad (\text{C.1})$$

Say now that a bit flip error occurs with probability p , flipping a 0 into a 1 or vice versa. The encoded bit $\tilde{x} \in \{000, 111\}$ can be recovered and decoded even if a single bit-flip error has happened on any of the three bits.

This can be done as follows. For an encoded bit $\tilde{x} = x_0x_1x_2$, assign the two *syndromes* s_0 and s_1 as

$$s_0 = x_0 \oplus x_1, \quad s_1 = x_1 \oplus x_2, \quad (\text{C.2})$$

where \oplus means addition modulo 2. The decoding process proceeds as follows; check the two syndromes s_0 and s_1 . Assume that at most a single bit-flip error has occurred. If $s_0 = 1$, then $x_0 \neq x_1$ and a bit-flip error has occurred on either x_0 or x_1 . If $s_1 = 1$, then $x_1 \neq x_2$ and a bit-flip error has occurred on either x_1 or x_2 .

If $s_0 = 1$ and $s_1 = 0$, flip x_0 . If $s_0 = 1$ and $s_1 = 1$, flip x_1 . If $s_0 = 0$ and $s_1 = 1$, flip x_2 . The encoded bit is now either a string of three zeros or three ones, coinciding with either the encoded 0-bit, or the encoded 1-bit.

This procedure ensures that a single bit-flip error on one of the *physical* bits does not result in an error in the *logical* bit. If bit-flip errors may occur

independently with probability p on any bit, the 3-bit repetition code lowers the probability of a *logical* error to $O(p^2)$, as two or more bit-flip errors must happen for the logical bit to retain an error.

C.2 The Quantum Repetition code

Now, we want to construct an analogous error-correction code for use on quantum bits. A first, naive idea would be to simply repeat the qubit state in question, $|\psi\rangle$ three times, giving

$$|\psi\rangle \rightarrow |\psi\rangle |\psi\rangle |\psi\rangle. \quad (\text{C.3})$$

This is, however, not allowed by the *no-cloning* theorem. The no-cloning theorem states that no quantum operator C , such that

$$C(|\psi\rangle |\phi\rangle) = |\psi\rangle |\phi\rangle, \quad (\text{C.4})$$

can exist. The theorem can be derived from the unitary nature of quantum operators, or as a consequence of *no-signalling*, which states that information cannot be transferred faster than the speed of light.

Thus, we need a different scheme. We first consider the analogue to a classical bit-flip error, which is a Pauli- X error. Consider the noise channel \mathcal{E} on a single qubit, defined by Kraus operators

$$E_0 = \sqrt{1-p}\mathbb{I}, \quad E_1 = \sqrt{p}X. \quad (\text{C.5})$$

Extend the noise model to several qubits as $\mathcal{E}_3 = \mathcal{E} \otimes \mathcal{E} \otimes \mathcal{E}$.

A better proposal for a quantum repetition code is the following; encode the computational basis states $\{|0\rangle, |1\rangle\}$ as

$$|0\rangle \rightarrow |000\rangle, \quad |1\rangle \rightarrow |111\rangle. \quad (\text{C.6})$$

The states $|\bar{0}\rangle \equiv |000\rangle$ and $|\bar{1}\rangle \equiv |111\rangle$ are the *code words* of this code. In the same way that $\{|0\rangle, |1\rangle\}$ form a basis for the state space of a single qubit, $\{|000\rangle, |111\rangle\}$ form a basis of the state space of the encoded *logical* qubit. This space is called the *code space*. The encoding of a general qubit state then becomes

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \rightarrow |\bar{\psi}\rangle = \alpha |\bar{0}\rangle + \beta |\bar{1}\rangle = \alpha |000\rangle + \beta |111\rangle. \quad (\text{C.7})$$

This encoding of a general qubit state $|\psi\rangle$ can be performed by the quantum circuit sub-procedure

$$\begin{array}{c}
 |\psi\rangle \text{---} \bullet \text{---} \bullet \text{---} \\
 |0\rangle \text{---} \oplus \text{---} \text{---} \\
 |0\rangle \text{---} \oplus \text{---} \text{---}
 \end{array} \cdot \quad (\text{C.8})$$

Analogous to the syndromes s_0 and s_1 of the classical repetition code, the syndrome operators, or *stabiliser operators*, of the quantum repetition code are defined as

$$Z_0Z_1, \quad Z_1Z_2, \quad (\text{C.9})$$

where Z_i is the Pauli- Z operator acting on qubit i , for $i = 0, 1, 2$. The eigenvalues of these operators are ± 1 . Notice how the general encoded state $|\bar{\psi}\rangle$ is a $+1$ eigenstate of both stabiliser operators. This is by design.

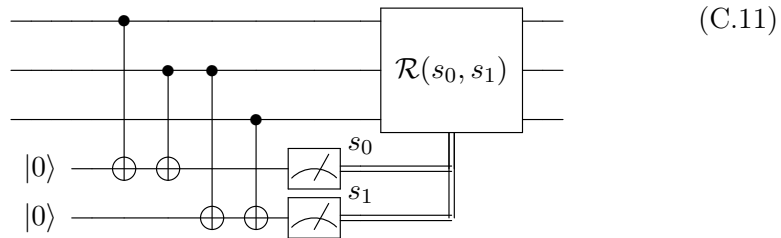
The syndromes s_0 and s_1 are obtained by measuring the stabiliser operators Z_0Z_1 and Z_2Z_3 , respectively. If an X -error have occurred, on say qubit 1, the error-afflicted encoded state becomes a -1 eigenstate of the Z_1 operator, as

$$Z_0Z_1X_1|\bar{\psi}\rangle = -X_1Z_0Z_1|\bar{\psi}\rangle = -X_1|\bar{\psi}\rangle. \quad (\text{C.10})$$

The faulty state $X_1|\bar{\psi}\rangle$ is thus a -1 eigenstate of both stabiliser operators. Here, we use that X_i anticommutes with Z_i , i.e., $X_iZ_i = -Z_iX_i$. X_i , however, commutes with any Z_j for $i \neq j$.

After obtaining the measurement outcomes, stored in s_0 and s_1 , the recovery proceeds in the same way as for the classical repetition code. Now, $s_0 = -1$ indicates an X error on physical qubit 0 or 1, $s_1 = -1$ indicates an X error on physical qubit 1 or 2, and so on. From this example, the measurements return -1 for both syndromes, indicating an X error on qubit 1

The recovery operation $\mathcal{R}(s_0, s_1)$ is done by applying an X -gate to the physical qubit on which the error has been detected. The non-destructive measurements of the stabilizers, followed by recovery operation, is done by the circuit sub-procedure as follows



The single-qubit X -errors, $\{X_i\}_{i=1,2,3}$, is the set of *correctable errors* for the 3-qubit quantum repetition code. These are the errors this code can exactly detect and correct.

In general, a *stabilizer code* is a quantum error-correcting code where the logical qubit states are encoded in the simultaneous $+1$ eigenstate of some set of stabilizer operators. Errors are detected by doing measurements of the stabilizer operators. In general, the set of correctable errors are the error operators which anticommutes with one or more of the stabilizer operators.

C.3 Digitization of errors

Consider an error-channel Kraus operator

$$E = \alpha\mathbb{I} + \beta X + \gamma Z + \delta XZ, \quad (\text{C.12})$$

where $\alpha, \beta, \gamma, \delta$ are complex coefficients.

Assume that we have a quantum error-correcting code in which Z is a stabilizer operator, and with an encoded state $|\bar{\psi}\rangle$. Then, $Z|\bar{\psi}\rangle = |\bar{\psi}\rangle$. By using the anticommutation relation $ZX = -XZ$, we get

$$ZE|\bar{\psi}\rangle = -1(\beta X + \delta XZ)|\bar{\psi}\rangle + (+1)(\alpha\mathbb{I} + \gamma Z)|\bar{\psi}\rangle. \quad (\text{C.13})$$

A measurement outcome of $+1$ thus leaves the logical qubit in the post-measurement state $(\alpha\mathbb{I} + \gamma Z)|\bar{\psi}\rangle$, which is a $+1$ eigenstate of Z , where no X error have occurred. An outcome of -1 leaves the post-measurement state $(\beta X + \delta XZ)|\bar{\psi}\rangle$, where an X error definitely has occurred.

In this way, the measurement of the stabiliser operator has collapsed the state, from some arbitrary superposition of error-free and error-afflicted states, into a state that can be said to definitely be either error-free or error-afflicted. This is with respect to the errors that are correctable by the given stabiliser. We say that the error has been *digitized*.

By concatenating the 3-qubit repetition code with the 3-qubit phase code, with stabilizers X_0X_1 and X_1X_2 , and code words $|\tilde{0}\rangle \equiv |+++ \rangle$ and $|\tilde{1}\rangle \equiv |-- \rangle$, we get the 9-qubit *Shor code*. The correctable errors for the Shor code are thus any X , Z , and joint XZ errors occurring on only a single qubit at a time. As all single-qubit Kraus operators can be written in the form of Equation (C.12), the Shor code can correct any arbitrary single one-qubit error [7].

C.4 Error Thresholds

Consider the quantum repetition code and an error model with Pauli- X error operators occurring on the qubits independently, on each with a probability p . The quantum repetition code allows any case in which at most one X -error occurs to be corrected. However, two or more X -errors happening at the same time will result in an error on the logical qubit, i.e., a *logical error*. The probability of a logical error p_{logical} , given a *physical* error probability p on the physical qubits, thus becomes

$$p_{\text{logical}} = 3p^2(1 - p) + p^3 = 3p^2 + \mathcal{O}(p^3). \quad (\text{C.14})$$

Assume the physical error rate p is such that $p = p^*$ for some p^* fulfilling

$$p_{\text{logical}} = p^* \quad \Rightarrow \quad 3(p^*)^2 + \mathcal{O}((p^*)^3) = p^* \quad \Rightarrow \quad p^* = \frac{1}{3}. \quad (\text{C.15})$$

Thus $p^* = \frac{1}{3}$ is the error threshold for the 3-qubit quantum repetition code, up to order $\mathcal{O}(p^3)$. If $p \leq p^*$, the encoding leaves the logical qubit with a logical error rate that is improved compared to the physical error rate.

If the physical error rate is below the threshold value, the logical error rate can be lowered to arbitrary precision by *concatenation*. One concatenation of the 3-qubit repetition code gives the 9-qubit code

$$|\tilde{\psi}\rangle = \alpha |\bar{0}\bar{0}\bar{0}\rangle + \beta |\bar{1}\bar{1}\bar{1}\rangle = \alpha |000000000\rangle + \beta |111111111\rangle, \quad (\text{C.16})$$

with logical error rate $p_{l=2} = (p_{l=1})^2 = 9p^4 + \mathcal{O}(p^5)$, where p_l is the logical error rate of a level l concatenated code. This can be repeated.

Note that the number of physical qubits needed grows as 3^l , i.e., exponentially. The saving grace is that the logical error rate $p_l = (p_{l-1})^2 = p^{2^l}$ drops super-exponentially with l . The total qubit requirement can thus be shown to increase polylogarithmically as a function of the inverse, target precision ϵ^{-1} , i.e., as $\text{polylog}(\epsilon^{-1})$.

