Hybrid Quantum-Classical Algorithms and Error Mitigation

Suguru Endo
Oriel College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy
Michaelmas 2019

Hybrid Quantum-Classical Algorithms and Error Mitigation

Suguru Endo

Oriel College
University of Oxford

A thesis submitted for the degree of Doctor of Philosophy

Michaelmas term 2019

These days, research groups such as Google, Microsoft, and Rigetti are working towards fabricating quantum devices which have hundreds to thousands of qubits. In recently proposed quantum algorithms, i.e. hybrid quantum-classical algorithms, such quantum devices are used as subroutines for calculation of classically intractable tasks. However, the applicability of hybrid algorithms was quite limited, for example, the search of the ground state, and simulation of the dynamics of quantum systems. In addition, near-term quantum devices are expected to be quite noisy, impairing the precision of computation seriously and removing any potential advantages with quantum computers. Therefore, it is necessary to extend the applicability of hybrid algorithms and mitigate errors on quantum computers.

In this thesis, we describe the generalisation of hybrid algorithms so that they can be applied to variety of problems, such as search of the spectrum of quantum systems, simulation of open quantum systems and even general mathematical tasks. The algorithm for finding spectra is useful for applications such as new drug discovery, design of batteries. Generally speaking, quantum systems of interest to physicists, chemists, and materials scientists inevitably interact with their environments, and quantum states are decohered. Therefore, to investigate quantum phenomena, new algorithms simulating open quantum systems are described in this thesis. Algorithms for general mathematical tasks such as matrix multiplication to a vector and solving linear equations, useful for a large number of problems, such as machine learning, are also presented.

Furthermore, in order to make hybrid algorithms useful, we also invented a practical error mitigation method for suppressing physical errors, which may make hybrid algorithms useful. We showed that errors can be suppressed for a quantum system with over 50 qubits for the current achievable error rate, which is believed to be hard to simulate with classical computers. Also, we generalised the error mitigation technique to be applied to mitigate algorithmic errors, which may enhance the accuracy of Hamiltonian simulation.

This thesis is dedicated to my beloved family.

Acknowledgements

Firstly, I thank my supervisor Professor Simon C. Benjamin. I am sure that he is the best supervisor in the world, as well as he is the best quantum researcher in the world. Whenever I had some ideas or was stuck in the research, he always kindly had discussions with me, and they were always very fruitful and stimulating. I will never forget you spent hours teaching me, and discussing research, however hard my English was to understand when I just came to UK three years ago. Even whenever I had problems, not related to research, for example, when I barely had unfair contract about housing, he always helped me. I can never thank enough the greatest supervisor in the world, Prof. Simon C. Benjamin. I really learned a lot of important things from you, as a researcher and as a person. Simon-sensei, Hontouni Arigatou Gozaimashita! Thank you very much!

I cannot thank Dr. Xiao Yuan enough. As a posdoc in our group, he always supervised me in really a nice way. He always polished my ideas, many of which became really nice works I am really proud of. As well as a great posdoc, he is the best friend in Oxford: we went drinking roughly twice a week, discussing research, very often talking about stupid things, sometimes very serious things about private life. I sincerely trusted Xiao, and I could talk about everything to him. I cannot imagine the life in Oxford without you, Xiao. Xiao, Hontouni Arigatou, Xie Xie!

I cannot thank Professor Ying Li enough. My first work was done with him, and accepted in really a nice journal. I leaned really a lot from that work and discussions with him. He also spent a long time teaching me, and it was really nice and inspiring. I got a lot of confidence from the work with him. Hontouni Arigaou- gozaimashita, Ying, Xie Xie!

I want to acknowledge Professor Yuichiro Matsuzaki. He gave me a chance to come to Oxford. Even after I came to Oxford, we had discussions and they were really useful. Matsuzaki-san, Itsumo- Arigatou Gozaimasu!

I thank Dr. Xiaosi-Xu. We went having lunch many times and talked about a lot of things. Sometimes the conversation with you was very encouraging, and saved me. I really thank you. Although your English is still better than mine, I am sure I can pronounce 'measurement' better than you! Anyway, I am really glad that we started in the same year and did our Ph.D course together. Xie Xie, Xiaosi-Xu!

I thank Dr. Zhenyu Cai. He always helped me in a lot of situation. I really enjoyed going around with you in China! I still remember talking about Chinese poems. Arigatou Xie Xie, Cai!

I thank Dr. Carlos Outeiral. The discussion with you was always inspiring! Gracias, Carlos!

I thank Dr. Tyson Jones. He is really good at programming, and always helped me. He is almost like a magician for programming! One of works I really take pride in is done with him. It is a really nice memory that we went around Tokyo and Kyoto! Arigato-thanks, Tyson-sama!

I thank Dr. Sam Mcardle. He is really knowlegeable in quantum computational chemistry, and asked him really a lot of questions. He always kindly taught me about it. I will never forget going around Bilbao with you! Sam-'tan' Arigatou!

I thank Dr. Takahiro Tsunoda. As he is an experimentalist, he gave me a lot of information about experiment and it was very useful. We also organised many events together, and all of them were always successful. This is thanks to him! Arigatou Takhiro!

I thank Dr. Bálint Koczor. He really helped me a lot with quantum metrology project. He is really a nice physicist and I want to learn a lot from him! It was a lot of fun to have sightseeing in Insbruck together with you, Bálint! Köszönöm!

I thank Dr. Jinzhao Sun. I had a really nice collaboration with him. He was a really nice friend as well, and we frequently went drinking! I am really proud that I could have a nice and excellent collaborator like you, Jinzhao! Jinzhao-chan Arigato Xie Xie!

I thank Dr. Leonard Wossnig. He is really a nice theorist. I still remember we went lunch together many time just when I arrived in Oxford three years ago. Even though my English is not good enough to have a fluent conversation at that time, you were always kind and patiently listened to me. I hope we will have a nice collaboration during my internship in your company! Danke, Leo!

I thank Dr. Joe O'Gorman, Dr. Andrea Rocchetto. They helped me a lot when I began my DPhi course! I also thank Dr. Qi Zhao, who I had fruitful discussions and went drinking many times with. I thank Dr. Martin Nicolle and Dr. Takuma Morimoto, Dr. Shinichi Sunami, Dr. Satoshi Kishigami, Dr. Daisuke Kawai and Mr. Tyuki Imamura. Thank you for always helping me! I also thank everyone in Japanese community! I thank quantum people in Riken, Keio, Insbruck University, people in Quanasys, and people in AIST, and NTT. Arigatou-gozaimasu! Thank you very much, Hayase-sensi, Koike-san, Neill-san, Nori-sensei.

I also acknowledge Professor Andrew Briggs, Mrs Marion Beckett. I also acknowledge Professor Peter Leek and Professor Earl Campbell.

Also, I would like to thank all of my beloved friends!

I also thank Miss Aiko Iwatusbo. Thank you very much for always being with me, supporting me. I love you. Itusumo-Arigatou, Ai-chan!

Finally, I thank my beloved and dearest family. Kono kazoku no ichiin to shite umarete korete yokatta. I am very glad that I was born as your son, Mom. Mom, please watch over me and enjoy seeing my growth from the heaven. Kazokuno minna, kokorokara arigatou.

Statement of original research contributions and related publications

The research chapters in this thesis are chapters 6 through 9. In each chapter the work reported is that of the author. Specifically,

Chapter 6 describes research that was reported in publication Phys. Rev. A 99, 062304 (2019). The present author was the first author of that publication, and was solely responsible for the key idea as well as the derivations of the analytic results (with input from other authors). There were numerical simulations that were not the work of the author, but rather the work of co-author Tyson Jones - some of these results are presented in chapter 6 in order to show the success of the technique, but each such graph is clearly identified.

Chapter 7 describes work that was reported in preprint arXiv:1812.08778. The present author was first author of that work and, with input from the other authors, conceived the ideas and produced the analytic treatments. Numerical results were obtained by the author, helped by Xiao Yuan.

Chapter 8 describes work that was reported in Phys. Rev. X 8, 031027 (2018). The concepts that are explored in that paper were jointly conceived by all authors; the present author was the first author of that publication. Analytic results and the derivations of the publications were also performed jointly by the author and Ying Li. Numerical results were mainly obtained by the author, except where noted.

Chapter 9 describes work that was reported in Phys. Rev. A 99, 012334 (2019). The present author was the first author of that publication, and was solely responsible for the key idea as well as the derivations of the analytic results (with input from other authors). The numerical results presented here were the work of the author.

In addition to the description above, the present author has appeared as co-author of several papers; this thesis does not feature those studies in detail, although ideas and insights from those papers are mentioned and referenced as appropriate.

Contents

Chapter 1

Introduction

Quantum computation (QC) has been studied for over two decades and many researchers believe that it is approaching realisation. QC is a completely new computation method using entities such as quantum bits (qubits), which are quantum generalisation of bits, permitting superposition and entanglement. It has been theoretically proven that, by using superposition and entanglement of a large number of qubits, we can use new approaches to solving problems classical computers cannot tackle practically. For example, it takes exponentially increasing time for even the best known classical algorithm to factorise numbers. However, Shor's algorithm can solve it efficiently in a polynomial time [1]. Also, it has been shown quantum computing enables the simulation of quantum systems such as chemical interactions and material systems efficiently [2, 3, 4, 5].

The most serious problem for the realisation of quantum computers is that when implementing qubits experimentally, there inevitably exist imperfections, leading to deterioration of quantumness. This impairs the precision of computation seriously, potentially removing any quantum advantage [6]. Therefore, methods to suppress the error by using encoding in quantum computation have been researched for decades [7, 8, 9, 10, 11], and it has been shown that fault-tolerant quantum computation can be achieved in principle, with error correction codes [12, 13]. Typically,

these codes have a threshold, i.e. a level of noise below which error correction can be successful. Sub-threshold qubits have been fabricated in superconducting systems [14] and ion trap systems [15, 16]. However, error correction techniques require hundreds to thousands of qubits to encode logical qubits [17], and the number of qubits necessary for postclassical factoring with Shor's algorithm is estimated to be several million [18], which is well beyond current quantum technology.

Meanwhile, research groups such as IBM and Google are expected to realise quantum processors composed of over 50 qubits in the near future, which is estimated to be large enough that their dynamics cannot be simulated by classical computers. Therefore, many researchers are trying to grasp what types of computing can be much more efficient with such quantum devices. Recently, hybrid quantum-classical algorithms, in which quantum processors with small number of qubits are utilised as a subroutine and classical computers are used for optimisation, are anticipated to be a promising candidate for the application of such quantum devices [19, 20, 21, 22, 23, 24]. It is hoped that, as we can assign a large number of tasks to classical computers, fully coherent operations are not required, unlike early quantum algorithms, which need quantum error correction. Such algorithms are shown to be useful for analysing molecular static and dynamic problems, for example [22, 24]. In a hybrid quantum-classical algorithm, the number of qubits per quantum processor is modest and the depth of the quantum circuit is shallow, so that quantum computation without full quantum error correction is feasible. However, when the hybrid quantum-classical algorithms were first proposed, the application was limited to the estimation of the ground state of a molecule Hamiltonian [19]. Afterwards, quantum algorithms for simulating the real and imaginary time were proposed. My coauthors and I further generalised the hybrid quantum-classical algorithms so that it can be used for obtaining the spectrum of the Hamiltonian [25] (a similar work was also suggested in Ref. [26] almost at the same time) and simulate the general process such as multiplication of general sparse matrix

to a state vector, which may be useful for machine learning, and simulating open quantum systems [27].

It is also vital to establish methods to suppress errors not relying on encoding, as well as developing hybrid quantum-classical algorithms. Y. Li and S. C. Benjamin [24] and an IBM group [28] independently invented an error mitigation scheme without encoding, which is expected to enhance the performance of the hybrid quantum/classical algorithm dramatically. However, these methods were incomplete in that the tolerable error rate was low and the applicable noise models were limited. My coauthors and I further developed the error mitigation schemes to be applicable to variety of noises and higher error rates [29]. In addition, we further generalised the quantum error mitigation to be applicable to mitigation of algorithmic errors [30].

In this DPhil thesis, in the second chapter, we will explain the basic knowledge required to understand this thesis. In the third chapter, we discuss early quantum algorithms which require fully coherent evolution and quantum error correction, to evaluate how costly these methods are well-beyond the current quantum technology. In the fourth chapter, we mention hybrid quantum/classical algorithms, which may be much more feasible for current experimental setups, where quantum devices are used as subroutines with small quantum devices, combined with classical computers for optimisation. In the fifth chapter, the error mitigation techniques to suppress errors on hybrid algorithms will be discussed. From the sixth chapter on, we will discuss the author's contributions to this field. In the sixth chapter, we will describe a new quantum algorithm for discovering the spectrum of the Hamiltonian. In the seventh chapter, we will mention our quantum algorithms for simulating general processes. In the eighth chapter, we will introduce the practical quantum error mitigation techniques, which are applicable to high error rate and general error models. In the ninth chapter, we will discuss the quantum error mitigation for algorithmic errors. The thesis will conclude with a summary chapter.

Chapter 2

Basics of quantum mechanics and quantum computing

The aim of this section is to give the basics of quantum mechanics required later in this thesis.

2.1 Quantum state

We will begin our description by describing pure quantum state, and introduce mixed state subsequently. The minimum component of the conventional classical computing is a "bit", which is expressed by 0 or 1. Meanwhile, the minimum component of quantum information is a "qubit", whose decomposition into basis states is expressed as

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \tag{2.1}$$

where $\alpha, \beta \in \mathbb{C}$, satisfying $|\alpha|^2 + |\beta|^2 = 1$, and $|0\rangle$ and $|1\rangle$ are orthogonal normalised vectors. For example, when we use an atom with different two energy levels as a qubit, we can use the lower energy level as $|0\rangle$ and the upper level as $|1\rangle$, and vice versa. $|\psi\rangle$ is called a state vector (or just state). $p_0 = |\alpha|^2$ and $p_1 = |\beta|^2$ are

the probabilities that the qubit is observed in state $|0\rangle$ or $|1\rangle$, respectively. The difference of the qubit from classical bit is that Eq. (2.1) is interpreted that the states $|0\rangle$ and $|1\rangle$ coexist, and such a phenomenon is called "superposition". The state vector in a quantum system can be expressed by using a Hilbert space, which is a complete vector space, where the coefficients are complex numbers and inner product is properly defined. The inner product of the two states $|\psi\rangle$ and $|\phi\rangle$ is denoted as $\langle\psi|\phi\rangle \in \mathbb{C}$. A qubit is expressed by using the two-dimensional Hilbert space $\mathbb{C}^2$. The norm of the quantum state $|\psi\rangle$ is unity, corresponding to the fact that the sum of the probabilities is always unity $(|\alpha|^2 + |\beta|^2 = 1)$.

One of other examples of quantum states is a qutrit, which is a superposition of $|0\rangle$, $|1\rangle$, and $|2\rangle$, expressed as

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle + \gamma\,|2\rangle\,. \tag{2.2}$$

Similarly to the case of qubit, $|0\rangle$, $|1\rangle$, $|2\rangle$ are orthnormal vectors, $\alpha, \beta, \gamma \in \mathbb{C}$, and $|\alpha|^2 + |\beta|^2 + |\gamma|^2 = 1$. Also, the probabilities that the state is observed in $|0\rangle$, $|1\rangle$, $|2\rangle$ are $p_0 = |\alpha|^2$, $p_1 = |\beta|^2$, and $p_2 = |\gamma|^2$. A qutrit is expressed in a three-dimensional Hilbert space $\mathbb{C}^3$. Such arguments can also be applicable to higher dimensional quantum states as well.

2.2 Composite quantum state

Here, we explain how to mathematically express the composite quantum system when we have two quantum systems, A and B, and corresponding Hilbert spaces $\mathrm{H_A}$ and $\mathrm{H_B}$. We denote the orthonormal basis in $\mathrm{H_A}$ and $\mathrm{H_B}$ as $\{|\psi_i\rangle_{\mathrm{A}}\}$ and $\{|\phi_j\rangle_{\mathrm{B}}\}$. When the states in A and B are $|\psi_i\rangle_{\mathrm{A}}$ and $|\phi_j\rangle_{\mathrm{B}}$, we denote the state in the composite system as

$$|\psi_i\rangle_{\mathrm{A}} \otimes |\phi_j\rangle_{\mathrm{B}} \tag{2.3}$$

or simply

$$|\psi_i\rangle_{\mathrm{A}} |\phi_j\rangle_{\mathrm{B}} , \tag{2.4}$$

where $\otimes$ denotes the tensor product. $\{|\psi_i\rangle_{\mathrm{A}} |\phi_j\rangle_{\mathrm{B}}\}$ is the orthonormal basis in the composite Hilbert space $\mathrm{H_A} \otimes \mathrm{H_B}$. Given $\mathrm{H_A}$ and $\mathrm{H_B}$ are N and M dimensional Hilbert space, the composite Hilbert space $\mathrm{H_A} \otimes \mathrm{H_B}$ becomes $N \times M$ dimensional Hilbert space.

Generally, the pure states in $\mathrm{H_A} \otimes \mathrm{H_B}$ can be written as

$$\sum_{i,j} \alpha_{i,j} |\psi_i\rangle_{\mathrm{A}} |\phi_j\rangle_{\mathrm{B}} , \tag{2.5}$$

where $\alpha_{i,j} \in \mathbb{C}$, and $\sum_{i,j} |\alpha_{i,j}|^2 = 1$ is satisfied.

If the composite system can be written as

$$|\psi\rangle_{\mathrm{A}} \otimes |\phi\rangle_{\mathrm{B}} , \tag{2.6}$$

where $|\psi\rangle_{\mathrm{A}} = \sum_i a_i |\psi_i\rangle_{\mathrm{A}}$ and $|\phi\rangle_{\mathrm{B}} = \sum_i b_i |\phi_i\rangle_{\mathrm{B}}$, the state in a composite system is called a product state or a separable state. On the other hand, when the state cannot be expressed as a separable state, the state is called an entangled state. Entangle states are of great interest at the philosophical level, since when two states are entangled it is no longer possible to understand them as independent entities [31, 32] .

2.3 Observable and projector

Here, we consider a measurement of an observable O, such as energy, the population of a certain level, photon number, etc. Observables are expressed by Hermitian

operators whose eigenvalues are real, implying measurement results of observables are also real. As O is a Hermitian operator, by using spectral decomposition, it necessarily can be decomposed as

$$O = \sum_k \lambda_k P_k, \tag{2.7}$$

where $\lambda_k \in \mathbb{R}$. For simplicity we assume O is a full rank matrix. Here, P_k is called projector and has properties such as

$$P_k P_l = \delta_{k,l} P_k \tag{2.8}$$

$$\sum_k P_k = I. \tag{2.9}$$

Now, we assume that an error free measurement can be implemented. In this case, one of the eigenvalues of O, λ_k, can be obtained as a measurement result, and the probability corresponding to λ_k is

$$p_k = \langle\psi| P_k |\psi\rangle, \tag{2.10}$$

according to the Born rule. Here, $|\psi\rangle$ is the measured quantum state. When the result λ_k (or simply we can say the result k) is obtained, the state is projected to

$$\frac{P_k |\psi\rangle}{\sqrt{\langle\psi| P_k |\psi\rangle}}. \tag{2.11}$$

This is the projection postulate, which has been verified empirically.

The expectation value of O is

$$\langle O \rangle = \sum_k p_k \lambda_k \tag{2.12}$$

$$= \sum_k \langle \psi | P_k | \psi \rangle \lambda_k \tag{2.13}$$

$$= \langle \psi | O | \psi \rangle . \tag{2.14}$$

2.4 Evolution of a quantum system

Here, we consider the time evolution of a closed quantum system, which is isolated from a noisy environment. The dynamics of the quantum system is governed by a Hermitian operator H, describing the energy in the quantum system, such as

$$i\hbar \frac{d}{dt} |\psi\rangle = H(t) |\psi\rangle , \tag{2.15}$$

which is so called the Schrödinger equation. H is called the Hamiltonian. For simplicity, we set reduced Planck constant $\hbar = 1$ in this thesis. In the case $H(t)$ is time independent, Eq. (2.15) can be formally solved as

$$|\psi(t)\rangle = \exp(-iHt) |\psi(0)\rangle . \tag{2.16}$$

Defining the time evolution operator $U(t) := \exp(-iHt)$, $U(t)$ is a unitary operator, satisfying $U^\dagger(t)U(t) = I$, where $\dagger$ denotes Hermitian conjugate of operators and I is an identity operator. When $H(t)$ is time dependent, the expression of $U(t)$ becomes as

$$U(t) := \vec{T} \exp\Big(-i \int_0^t H(t')dt' \Big) \tag{2.17}$$

$$:= \sum_{m=0}^{\infty} (-i)^m \int_0^t dt_1 ... \int_0^{t_{m-1}} dt_m H(t_1)...H(t_m), \tag{2.18}$$

where $\vec{T}$ denotes a time ordered product operator. Also in this case, $U(t)$ is a unitary operator.

Assuming the quantum system is a qubit, and $H = \frac{\omega}{2}X$, $\omega \in \mathbb{R}$, $X := |0\rangle\langle 1| + |1\rangle\langle 0|$, $U(t)$ becomes rotation X operator defined as

$$R_X(\theta(t)) := \exp\left(-i\frac{\omega}{2}tX\right) \tag{2.19}$$

$$= \cos\left(\frac{\theta(t)}{2}\right)I - i\sin\left(\frac{\theta(t)}{2}\right)X, \tag{2.20}$$

where we set $\theta(t) = \omega t$ and used $X^2 = I$, and $\exp(\mathrm{A}) = \sum_{k=0} A^k/k!$. By setting $\theta(t) = \pi$, we have $R_X(\pi) = -iX$, which flips $|0\rangle$ to $|1\rangle$ of the state, and vice versa. Therefore, we could build a unitary operator which implements a bit flip operation. Similarly, by designing a Hamiltonian and evolving the state for the proper time duration, we can realise the unitary operator which implements the task we hope to obtain.

Two important examples are for operators $Y := i(|0\rangle\langle 1| - |1\rangle\langle 0|)$, and $Z = |0\rangle\langle 0| - |1\rangle\langle 1|$, we define rotation operators as

$$R_Y(\theta(t)) := \exp\left(-i\frac{\omega}{2}tY\right) \tag{2.21}$$

$$= \cos\left(\frac{\theta(t)}{2}\right)I - i\sin\left(\frac{\theta(t)}{2}\right)Y, \tag{2.22}$$

$$R_Z(\theta(t)) := \exp\left(-i\frac{\omega}{2}tZ\right) \tag{2.23}$$

$$= \cos\left(\frac{\theta(t)}{2}\right)I - i\sin\left(\frac{\theta(t)}{2}\right)Z. \tag{2.24}$$

Here, operators X, Y, Z are called Pauli operators, and satisfy $X^2 = Y^2 = Z^2 = I$,

and

$$[X, Y] = 2iZ,\ [Y, Z] = 2iX,\ [Z, X] = 2iY \tag{2.25}$$

where $[A, B] = AB - BA$. Also, it is important to note that these Pauli operators have eigenvalues ± 1.

2.5 Density matrix

Prior to this section, we discussed the case where the quantum system is closed, isolated from a noisy environment. However, quantum systems generally interact with the environment. Generally we will not know the exact quantum state of the environment - we will need to use a probability distribution to describe the environment. After interacting with the environment, our quantum system can be no longer described only by one state vector. Suppose that, in the quantum system of interest, we have a quantum state $|\psi_i\rangle$ with the probability q_i. Then, the quantum state can be decomposed as

$$\rho = \sum_i q_i \left|\psi_i\right\rangle \left\langle\psi_i\right| . \tag{2.26}$$

Here, ρ is called a density matrix, a representation of a quantum state suitable for an open system interacting with a noisy environment. If the quantum state can be written as

$$\rho = \left|\psi\right\rangle \left\langle\psi\right| , \tag{2.27}$$

the quantum state is called a pure state. If the state is not a pure state, the state is in a mixed state.

When we measure an observable $O = \sum_k \lambda_k P_k$, the probability to have the

result λ_k is

$$p_k = \sum_i q_i \langle\psi_i| P_k |\psi_i\rangle \tag{2.28}$$

$$= \mathrm{Tr}(\rho P_k). \tag{2.29}$$

Here, Tr is a trace of an operator, and it is defined for an operator A as

$$\mathrm{Tr}(A) := \sum_k \langle\phi_k| A |\phi_k\rangle , \tag{2.30}$$

where $\{|\phi_k\rangle\}$ is an orthonormal basis, and trace is independent of the choice of the basis. Similarly, we can have the expectation value of an observable O as,

$$\langle O\rangle = \sum_k p_k \lambda_k \tag{2.31}$$

$$= \sum_k \lambda_k \sum_i q_i \langle\psi_i| P_k |\psi_i\rangle \tag{2.32}$$

$$= \sum_i q_i \langle\psi_i| \left(\sum_k \lambda_k P_k \right) |\psi_i\rangle \tag{2.33}$$

$$= \mathrm{Tr}\big(\rho O\big) \tag{2.34}$$

2.6 Bloch sphere representation

Considering the fact that the density matrix has a trace equal to unity and it is a Hermitian operator, the density matrix for a two level system i.e. a qubit can be expressed as

$$\rho = \frac{I + r_x X + r_y Y + r_z Z}{2}, \tag{2.35}$$

where $r_i \in \mathbb{R}$, $i = \{x, y, z\}$ and $r_x^2 + r_y^2 + r_z^2 \leq 1$. Therefore, the qubit state can be mapped to the surface or inside of a ball whose radius is equal to unity, which

is called the Bloch sphere. When the state is a pure state, the state is mapped to a surface of the Bloch sphere. If $r_x = r_y = r_z = 0$, $\rho = I/2$, the quantum state is called a completely mixed state.

2.7 Density matrix in a composite quantum system

Suppose we have two quantum systems A and B, and the corresponding Hilbert spaces are $\mathrm{H_A}$ and $\mathrm{H_B}$. When the states in A and B are ρ_A and ρ_B, we can express the quantum state in the composite Hilbert spaces $\mathrm{H_A} \otimes \mathrm{H_B}$ as

$$\rho_\mathrm{AB} = \rho_\mathrm{A} \otimes \rho_\mathrm{B}, \tag{2.36}$$

which is called a product state. If the state is expressed as

$$\rho_\mathrm{AB} = \sum_i q_i \rho_\mathrm{A}^{(i)} \otimes \rho_\mathrm{B}^{(i)}, \tag{2.37}$$

where $\rho_\mathrm{A}^{(i)}$ and $\rho_\mathrm{B}^{(i)}$ are the quantum states in A and B, the state is a separable state. A separable state only has classical correlations between systems A and B. If the state cannot be expressed in the form of Eq. (2.37), the state is an entangled state.

2.8 Partial trace

Conversely, partial trace is a mathematical operation to derive the states of individual systems. Given the composite state ρ_AB, the individual state in A is described,

due to partial trace,

$$\rho_{\mathrm{A}} := \mathrm{Tr}_{\mathrm{B}}(\rho_{\mathrm{AB}}) \tag{2.38}$$

$$:= \sum_k \langle\phi_k|_{\mathrm{B}}\, \rho_{\mathrm{AB}}\, |\phi_k\rangle_{\mathrm{B}}\,, \tag{2.39}$$

where $\{|\phi_k\rangle_{\mathrm{B}}\}$ is an orthogonal basis of the Hilbert space $\mathrm{H_B}$. This mathematical operation can also be applied to general operators other than density matrices. The procedure is the same for having the individual state ρ_{B}. Also,

$$\mathrm{Tr}(\rho_{\mathrm{AB}}) = \mathrm{Tr}_{\mathrm{A}}(\mathrm{Tr}_{\mathrm{B}}(\rho_{\mathrm{AB}})) \tag{2.40}$$

$$= \mathrm{Tr}_{\mathrm{B}}(\mathrm{Tr}_{\mathrm{A}}(\rho_{\mathrm{AB}})) \tag{2.41}$$

is satisfied, which also holds for general operators. For an operator $M_{\mathrm{A}} \otimes I_{\mathrm{B}}$, which only operates on the system A, we have

$$\mathrm{Tr}_{\mathrm{A}}(\rho_{\mathrm{A}} M_{\mathrm{A}}) = \mathrm{Tr}_{\mathrm{AB}}(\rho_{\mathrm{AB}}\ M_{\mathrm{A}} \otimes I_{\mathrm{B}}), \tag{2.42}$$

by using Eq. (2.41). Therefore, ρ_{A} gives the correct expectation value of the observable in the system A, which is the reason we can use the partial trace for knowing the individual state in a composite system.

2.9 Evolution of a density matrix in a closed quantum system

Now, we discuss the evolution of a density matrix in a closed system. We assume each quantum state $|\psi_i\rangle$ of $\rho = \sum_i q_i\, |\psi_i\rangle\, \langle\psi_i|$ obeys the Shrödinger equation (2.15),

obtaining

$$\frac{d}{dt}|\psi_i(t)\rangle\langle\psi_i(t)| = \frac{d|\psi_i(t)\rangle}{dt}\langle\psi_i(t)| + |\psi_i(t)\rangle\frac{d\langle\psi_i(t)|}{dt} \tag{2.43}$$

$$= -iH|\psi_i(t)\rangle\langle\psi_i(t)| + |\psi_i(t)\rangle\langle\psi_i(t)|(iH) \tag{2.44}$$

$$= -i[H, |\psi_i(t)\rangle\langle\psi_i(t)|]. \tag{2.45}$$

Therefore, we have the equation $\rho(t)$ follows as

$$\frac{d\rho(t)}{dt} = -i[H, \rho(t)], \tag{2.46}$$

which is von-Neumann equation describing the time evolution of a density matrix in a closed system. When H is time independent, the solution of Eq. (2.46) is

$$\rho(t) = U(t)\rho(0)U^\dagger(t), \tag{2.47}$$

where $U(t) = \exp(-iHt)$, which is a unitary operator.

2.10 Open quantum system and Kraus representation

Generally, the quantum system interacts with the environment, and cannot be described only with a unitary operator. Suppose the evolution of the system composed of the quantum system we are interested in and the environment is described by a unitary operator U. If we only consider the dynamics of the system, the dynamics becomes non-unitary.

Now, assume that the initial state of the composite system is written as the

product form as

$$\rho_{\text{com}} = \rho_{\text{s}} \otimes \rho_{\text{env}}. \tag{2.48}$$

Suppose the evolution of the composite system is described by a unitary operator U and the final state becomes

$$\rho'_{\text{com}} = U\rho_{\text{s}} \otimes \rho_{\text{env}}U^{\dagger}. \tag{2.49}$$

Since we are only interested in the evolution of the system, not the environment, we apply the partial trace to ρ_{com} to extract the evolution of the system density matrix ρ_{s}, such as

$$\rho'_{\text{s}} = \text{Tr}_{\text{env}}\big(U\rho_{\text{s}} \otimes \rho_{\text{env}}U^{\dagger}\big). \tag{2.50}$$

Now, we set $\rho_{\text{env}} = \sum_k q_k \left|\phi_k\right\rangle \left\langle\phi_k\right|$, and the orthogonal basis for taking partial trace as $\{\left|\psi_l\right\rangle\}$. Then we have

$$\rho'_{\text{s}} = \mathcal{E}(\rho_{\text{s}}) = \sum_{k,l} q_k \left\langle\psi_l\right| U\rho_{\text{s}} \otimes \left|\phi_k\right\rangle \left\langle\phi_k\right| U^{\dagger} \left|\psi_l\right\rangle \tag{2.51}$$

$$= \sum_{k,l} \sqrt{q_k} \left\langle\psi_l\right| U \left|\phi_k\right\rangle \rho_{\text{s}} \sqrt{q_k} \left\langle\phi_k\right| U^{\dagger} \left|\psi_l\right\rangle \tag{2.52}$$

$$= \sum_{m} K_m \rho_{\text{s}} K_m^{\dagger}, \tag{2.53}$$

where we set $m = (k, l)$, $K_m = \sqrt{q_k} \left\langle\psi_l\right| U \left|\phi_k\right\rangle$. Eq. (2.53) is called a Kraus representation, and it is known that any physical process can be described by Kraus representation [31]. The probability the process K_m occurs is $p_m = \text{Tr}(\rho K_m^{\dagger} K_m)$. It can be easily proved that the Kraus map satisfies the completeness condition $\sum_m K_m^{\dagger} K_m = I$, therefore $\text{Tr}(\rho'_s) = 1$, and the trace of the quantum state is generally preserved, which corresponds to the preservation of the probability.

2.11 Examples of noisy quantum process

The depolarising channel is a well-studied noisy quantum process [31]. Suppose X, Y, Z randomly operates on a qubit with probability $p_d/3$ respectively, with I occurring with probability $1 - p_d$. In this case, the set of Kraus operators can be written as $\{\sqrt{1-p_d}I, \sqrt{p_d/3}X, \sqrt{p_d/3}Y, \sqrt{p_d/3}Z\}$, and the quantum process can be expressed as

$$\mathcal{E}_d(\rho) = (1 - p_d)\rho + \frac{p_d}{3}(X\rho X + Y\rho Y + Z\rho Z). \tag{2.54}$$

Also, by using the identity $I/2 = \frac{1}{4}(\rho + X\rho X + Y\rho Y + Z\rho Z)$, satisfied for an arbitrary ρ, we have,

$$\mathcal{E}_d(\rho) = \left(1 - \frac{4p_d}{3}\right)\rho + \frac{4p_d}{3}\frac{I}{2}. \tag{2.55}$$

Therefore, we can also interpret the depolarising channel as the channel where the quantum state is replaced with completely mixed state with probability $4p_d/3$.

Now, consider the case of a single qubit, and suppose ρ can be mapped to (r_x, r_y, r_z) when we employ the Bloch representation of a qubit. Due to the depolarising channel, the Bloch sphere representation of a qubit becomes as

$$(r_x, r_y, r_z) \tag{2.56}$$

$$\rightarrow \left(\big(1 - \frac{4p_d}{3}\big)r_x, \big(1 - \frac{4p_d}{3}\big)r_y, \big(1 - \frac{4p_d}{3}\big)r_z\right). \tag{2.57}$$

Therefore, x, y, z coordinates shrinks uniformly. When the probabilities associated with the X, Y, Z channels are unisotropic, we call such a quantum process inhomogeneous Pauli error.

Another well-known type of quantum noisy process is amplitude damping, which

describes the energy loss from a qubit [31]. The quantum process can be written as

$$\mathcal{E}_a(\rho) = K_0 \rho K_0^\dagger + K_1 \rho K_1^\dagger, \tag{2.58}$$

where

$$K_0 = |0\rangle\langle 0| + \sqrt{1-\gamma}\,|1\rangle\langle 1| \tag{2.59}$$

$$K_1 = \sqrt{\gamma}\,|0\rangle\langle 1|\,, \tag{2.60}$$

and $K_0^\dagger K_0 + K_1^\dagger K_1 = I$. It can be easily checked K_0 reduces the population of the state $|1\rangle$, and K_1 changes the state $|1\rangle$ into $|0\rangle$. Therefore, the relaxation process from $|1\rangle$ to $|0\rangle$ can be explained by using the amplitude damping.

2.12 Lindblad master equation

Here, we explain the non-unitary time evolution of an open quantum system, interacting with the environment. When the open quantum system obeys the Markov process, the dynamics can be described with Lindblad master equation [33, 34]. A Markov process is one in which the probabilistic dynamics at time $t + \delta t$ depends only on the time just before, t, where δt is an infinitestimal. Therefore, the quantum state of the system $\rho_s(t + \delta t)$ can be described with $\rho_s(t)$ by using Kraus representation, such as

$$\rho(t + \delta t) = \sum_{m \geq 0} K_m \rho(t) K_m^\dagger. \tag{2.61}$$

Physically, this can be interpreted that the system and the environment are in the product state, and after the system interacted with the environment from t to $t+\delta t$, the environment is replaced with the new environment which has never interacted with the system. As this new environment has no information with regard to the

past of the system, the quantum state of the system obeys Markov process. Now, we set

$$K_0 = I + (D - iH)\delta t \tag{2.62}$$

$$K_m = L_m\sqrt{\delta t}, \tag{2.63}$$

where D and H are Hermitian operator and L_m operators describing the interaction with the environment. From completeness condition $\sum_{m\geq 0} K_m^\dagger K_m = I$, we have

$$I + \left(2D + \sum_{m\geq 1} L_m^\dagger L_m\right)\delta t + O(\delta t^2) = I. \tag{2.64}$$

Thus, by ignoring higher order terms of δt as it is sufficiently small, we have

$$D = -\frac{1}{2}\sum_{m\geq 1} L_m^\dagger L_m + O(\delta t^2). \tag{2.65}$$

By substituting K_m in Eq. (2.61) with Eq. (2.63), we have

$$\rho(t+\delta t) = \rho(t) - i[H,\rho(t)]\delta t + \left(\sum_m L_m\rho(t)L_m^\dagger + D\rho(t) + \rho(t)D\right)\delta t + O(\delta t^2), \tag{2.66}$$

Substituting D with Eq. (2.65) and taking $\delta t \to 0$, finally we have

$$\dot{\rho}(t) = -i[H,\rho(t)] + \frac{1}{2}\sum_m \left(2L_m\rho(t)L_m^\dagger - L_m^\dagger L_m\rho(t) - \rho(t)L_m^\dagger L_m\right). \tag{2.67}$$

This time derivative equation is known as Lindblad master equation. Note that, H corresponds to the system Hamiltonian and L_m are called Lindblad operators.

2.13 Quantum circuits

In this thesis, we focus on the qubit-based circuit model of quantum computation [31]. Other types of quantum computation are adiabatic quantum computing [35, 36], one-way or measurement based quantum computing [37, 38, 39], and continous-variable quantum computing [40, 41]. In the case of qubit-based circuit model of quantum computation, there is a useful graphical representation for showing what operations and measurements are applied to the quantum state. This graphical representation is simply called the quantum circuit. Now we show an example of the quantum circuit in Fig. 2.1.

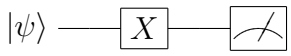


Figure 2.1: An example of a quantum circuit. The input state is $|\psi\rangle$, X operation is applied, and eventually the qubit is measured in the Z basis.

The input state is written in the leftmost of the quantum circuit, and the operations applied are shown in the middle, eventually the state is measured in Z basis, which is drawn in rightmost. When X and Y operations are applied, they are similarly written to Fig. 2.1.

Typical gates also include the Hadamard gate and T gate, which can be expressed as in Eq. (2.68),

$$\begin{aligned} H &= \frac{1}{\sqrt{2}}(X+Z) \\ T &= \exp\left(i\frac{\pi}{8}Z\right), \end{aligned} \tag{2.68}$$

and they are graphically drawn as Fig. 2.2.

We also express the rotation gate around x, y, z by rotation angle θ as Fig. 2.3.

To describe, or even approximate arbitrary input-output relationships in quantum computing (so called universal quantum computing), we need to have entangling operations. It has been proved that only single qubit operations and controlled not (CNOT) gates, which are one of the entangling gates, are required to

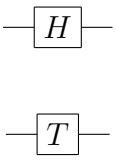


Figure 2.2: The graphical representation of Hadamard gate and T gate.

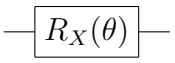


Figure 2.3: The graphical representation of rotation gate around x axis by rotation angle θ.

approximate any multi-qubit gate to any desired accuracy [42]. The CNOT gate is mathematically expressed as

$$U_{CN} = |0\rangle\langle 0|_C \otimes I_T + |1\rangle\langle 1|_C \otimes X_T, \tag{2.69}$$

where C and T denote controlled and target qubits, respectively. The graphical representation of CNOT gate is shown in Fig. 2.4. It is worth mentioning that,

Figure 2.4: The graphical representation of CNOT gate.

although the gates we have introduced so far are specific fixed unitaries, we can also consider circuits whose gates are parametrised with a classical value. This is essential in variational quantum algorithms (VQAs) [19, 20, 21, 22, 23, 24].

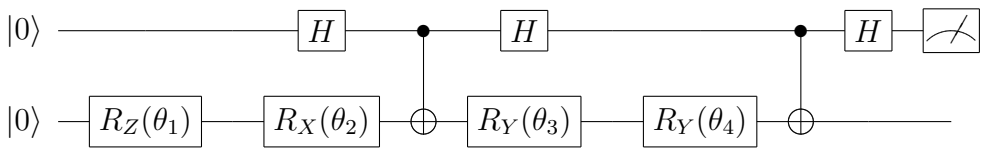


Figure 2.5: An example of a parametrised quantum circuit.

Chapter 3

Quantum algorithms in the early stage of quantum computing and quantum error correction

3.1 Quantum algorithms

In this section, we briefly discuss certain quantum algorithms which were invented in the early stage of quantum computing. Some of these algorithms have exponential speedup over existing classical algorithms, and are expected to have a large number of applications. However, these early quantum algorithms necessitate deep quantum circuit and need to be implemented with fully coherent operations. Therefore, a quantum error correction technique requiring encoding qubits has to be implemented, which is costly in current quantum devices.

3.1.1 Quantum Fourier transform

Firstly, we mention quantum Fourier transform [43], as it has large number of applications. The definition of quantum Fourier transform on orthonormal basis

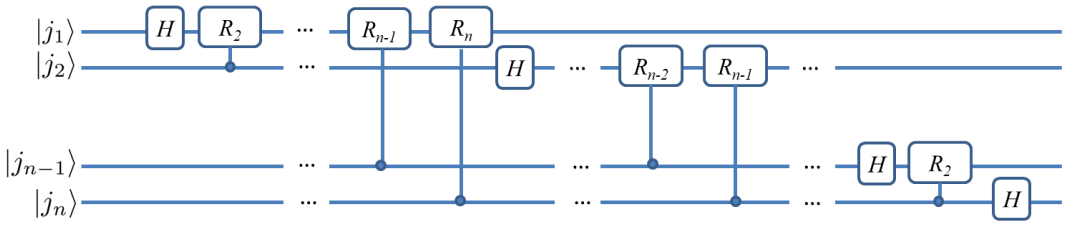


Figure 3.1: The quantum circuit for Quantum Fourier Transform

$|0\rangle$, ... $|N-1\rangle$ is

$$|j\rangle \rightarrow \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} e^{2\pi jm/N} |m\rangle \tag{3.1}$$

To express this process using qubits, it is convenient to introduce the product representation as follows

$$|j_1, j_2, ..., j_n\rangle \rightarrow \frac{\big(|0\rangle + e^{2\pi 0.j_n}|1\rangle\big)\big(|0\rangle + e^{2\pi 0.j_{n-1}j_n}|1\rangle\big)...\big(|0\rangle + e^{2\pi 0.j_1 j_2 ... j_n}|1\rangle\big)}{2^{n/2}} \tag{3.2}$$

where $N = 2^n$, $j = j_1 2^{n-1} + j_2 2^{n-2} + ... + j_n 2^0$ using binary representation, and $0.j_m j_{m+1} ... j_n = j_m/2 + j_{m+1}/2^2 + ... + j_n/2^{n-m+1}$ is the binary fraction. This transformation can be realized using the quantum circuit shown in Fig. 3.1, where

$$R_l = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^l} \end{pmatrix} \tag{3.3}$$

With Fig. 3.1 and a straightforward calculation, the number of total gates for the quantum Fourier transform turns out to scale as $O(n^2)$. On the other hand, the best classical algorithm for computing Fast Fourier Transform requires $O(n2^n)$ gates [44], which means the cost increases exponentially. However, this does not immediately imply that quantum Fourier transform can be an extremely useful tool for computing, as we cannot obtain the information of the amplitudes

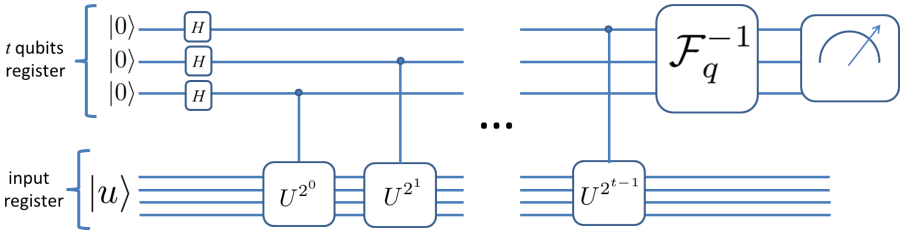


Figure 3.2: The quantum circuit for quantum phase estimation

in a quantum computer directly. In spite of that, quantum Fourier transform has useful applications for quantum algorithms, such as quantum phase estimation and factorisation as we discuss in the following subsections.

3.1.2 Quantum phase estimation

Phase estimation is one of the important applications of quantum Fourier transform [45]. Phase estimation algorithm is the quantum algorithm with which we can evaluate the eigenvalue $e^{2\pi i\phi}$ for an eigenstate $|u\rangle$ of a unitary operator U. Here we assume that the phase can be expressed by using t bits as $\phi = 0.\phi_1...\phi_t = \phi_1/2 + \phi_2/2^2 + ... + \phi_t/2^t$. In this case, The crucial requirement of this algorithm is the capability to perform the controlled U^{2^j} gate. The quantum circuit for quantum phase estimation is shown in Fig. 3.2.

The qubits comprise of two parts: the first register with t qubits, (t is relevant to the precision) and the second register which is used for expressing $|u\rangle$. Also, the algorithm consists of two stages.

Suppose, firstly the t qubit register is set to $|0\rangle$ and the first stage corresponds to applying Hadamard gates to the t qubit register and performing controlled U^{2^j} gates from $j = 0$ to $j = t$ successively, as shown in Fig. 3.2. Then, the state of the

t qubit register becomes

$$\frac{1}{\sqrt{2^t}}\big(\,|0\rangle + e^{2\pi 2^{t-1}\phi}\,|1\rangle\,\big)\big(\,|0\rangle + e^{2\pi 2^{t-2}\phi}\,|1\rangle\,\big)...\big(\,|0\rangle + e^{2\pi 2^{0}\phi}\,|1\rangle\,\big) \tag{3.4}$$

$$= \frac{1}{\sqrt{2^t}}\big(\,|0\rangle + e^{2\pi i 0.\phi_t}\,|1\rangle\,\big)\big(\,|0\rangle + e^{2\pi i 0.\phi_{t-1}\phi_t}\,|1\rangle\,\big)...\big(\,|0\rangle + e^{2\pi i 0.\phi_1\phi_2...\phi_t}\,|1\rangle\,\big) \tag{3.5}$$

$$= \mathcal{F}_{\mathrm{q}}\big(\,|\phi_1,\phi_2,...,\phi_t\rangle\,\big) \tag{3.6}$$

Where $\mathcal{F}_{\mathrm{q}}$ is the map for quantum Fourier transform, and we expressed $\phi = 0.\phi_1...\phi_t = \phi_1/2 + \phi_2/2^2 + ... + \phi_t/2^t$, using the binary expression. The second stage involves performing the inverse quantum Fourier transform $\mathcal{F}_{\mathrm{q}}^{-1}$ to the t qubits register, which can be implemented with $O(t^2)$ steps, so that the final state becomes $|\phi_1,\phi_2,...\phi_t\rangle$. Therefore, by measuring the state in the computational basis, we can obtain ϕ. The precision is restricted by t. For a more general case where the phase estimation is applied to the superposition of eigenstates of U such as $\sum_u c_u\,|u\rangle$, the state is projected to $|u\rangle$ with a probability $|c_u|^2$ [46, 31].

3.1.3 Shor's factoring algorithm

Shor's factoring algorithm is the most well-known quantum algorithm that has had a significant impact on the field of the quantum information. The vital point of this algorithm is that it is practically useful: it makes factoring exponentially faster than existing classical algorithms, and breaks the conventional public-key cryptography such as RSA scheme based on the complexity of factoring [1]. Shor's factoring algorithm only requires $O(L^3)$ gates [31, 1], where $L = \lceil \log(N) \rceil$. The key of Shor's factoring algorithm is to use the phase estimation algorithm to solve the order finding problem

$$a^r \equiv 1 \pmod{N} \tag{3.7}$$

and to find integer r for a randomly generated in the rage $0 \leq a \leq N$, which cannot be solved efficiently by classical computers.

3.1.4 Quantum simulation

As the quantum system to be simulated becomes large, the Hilbert space to express it grows exponentially, thus simulating a large sized quantum system is an intractable problem for a classical computer.

The concept of quantum simulation was proposed by Richard Feynman in 1982, as can be seen in the quote "Let the computer itself be built of quantum mechanical elements which obey quantum mechanical laws." [47]. After more than a decade, Seth Lloyd showed that quantum computers can act as the universal simulator of quantum systems [2]. Quantum simulation is considered to be useful for a variety of problems, including quantum chemistry.

The algorithm introduced by Seth Lloyd, generally called digital quantum simulation, is for simulating the time evolution of a quantum dynamics, and can be implemented by using the Trotter decomposition [31]. Suppose we try to simulate the Hamiltonian H on a quantum computer. As the formal solution of Shrödinger equation $i\frac{d}{dt}|\psi(t)\rangle = H|\psi(t)\rangle$ is $|\psi(t)\rangle = \exp(-iHt)|\psi(0)\rangle$, the goal is to simulate the time evolution unitary operator $U(t) = \exp(-iHt)$. Assume that the Hamiltonian can be expressed as the sum of local interactions, which is a reasonable assumption for many quantum systems, for instance the Hubbard and Ising models, as follows

$$H = \sum_k H_k, \tag{3.8}$$

where H_k is a local interaction Hamiltonian. Then, by using the first order Trotter decomposition, we can decompose the time evolution unitary operator into local

gates so that we have

$$U(t) = \Big(\prod_k \exp\big(- iH_k t/N \big) \Big)^N + O\big(t^2/N\big) \tag{3.9}$$

For sufficiently large N, we obtain

$$U(t) \approx \Big(\prod_k \exp\big(- iH_k t/N \big) \Big)^N \tag{3.10}$$

By iteratively applying $\prod_k \exp\big(-iH_k t/N\big)$ for N times, we can simulate the quantum dynamics to time t. However, to enhance the accuracy we have to have large N, so that a larger number of gates are required, and to prevent errors accumulating, fault tolerant quantum error correction we will mention later is required.

Also, Alán Aspuru-Guzik *et al.*, presented a quantum algorithm which can estimate the ground state energy and prepare the ground state of a molecule, rather than the time evolution of a quantum system by using a quantum phase estimation algorithm combined with the adiabatic state preparation [3]. The idea is that in the quantum phase estimation algorithm, by setting $U = \exp\big(- iH\tau \big)$ where H is the simulated Hamiltonian, the molecule energy E are mapped to the information of phase as $2\pi\phi/\tau$ where ϕ is the phase estimated in the algorithm. The gate complexity of this algorithm grows polynomially to the system size and the desired accuarcy, assuming the number of terms in the Hamiltonian grows polynomially, and this algorithm enables exponential speedup over classical algorithms [3]. However, to implement the phase estimation, every process on the quantum circuit has to be coherent, therefore, the performance of it is severely restricted by coherence time of the system.

3.2 Quantum error correction

As the main focus of the thesis is hybrid quantum-classical algorithms and quantum error mitigation, it does not contribute to the field of the quantum error correction. However, it is important to mention quantum error correction for readers to understand the motivation to introduce hybrid quantum-classical algorithms and quantum error mitigation. Here, we explain the principle of fault tolerant error correction. The basic idea of any error correction code is to encode and thus protect quantum states. Before proceeding to the discussion of quantum error correcting codes, we briefly review the classical 3 bit code. In this encoding, the information is encoded in 000 or 111. Suppose, these bits are sent through a noisy channel, and flip with independent probability p. In this case, if we employ majority voting to correct the code, the failure probability p_{f} is $p_{\mathrm{f}} = 3p^2(1-p) + p^3$, which is a probability where errors occur to two bits or three bits. Thus, we can suppress the error probability p to $O(p^2)$. The basic idea of the quantum error correction is the same. However, there are some constraints in it: no-cloning theorem and collapse of wave function by measurement.

3.2.1 Three-qubit code

Firstly, we review the most basic error correcting code, the three-qubit code with which we can correct a qubit flip error (X error) [31].

Suppose, the qubit state we want to preserve is $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. By using two control NOT gates as shown in Fig. 3.3, we can encode the state as

$$|\psi\rangle \to \alpha|000\rangle + \beta|111\rangle \tag{3.11}$$

However, if we naively measure the state of the qubit, the superposition is destroyed. Therefore, a "syndrome" measurement is implemented, which does not measure the state but just "parity" of the state. More concretely, using the circuit

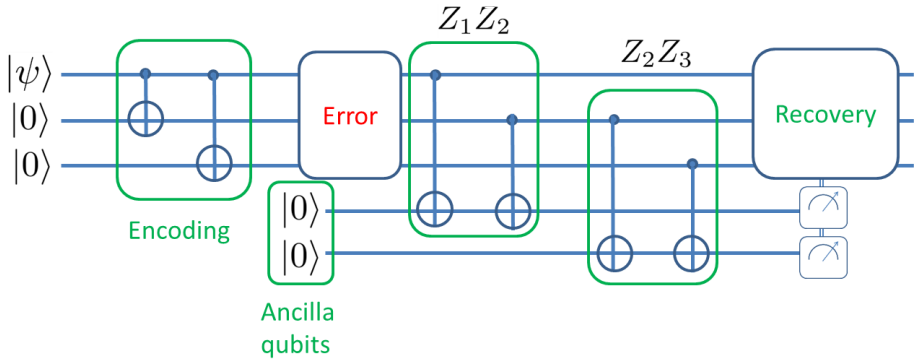


Figure 3.3: The quantum circuit for three qubits code

shown in Fig. 3.3, we measure Z_1Z_2 and Z_2Z_3 by using two ancilla qubits, where Z_i denotes sigma Z operator operating on the i th qubit. Depending on the signs of these operators, we can spot what qubit was affected by errors, summarised in Table 1. To find the value of $Z_\mathrm{A}Z_\mathrm{B}$, in practice we can apply CNOT gates from qubits A and B onto a target ancilla initially in $|0\rangle$, and measure as in Fig. 3.3.

error	no error	qubit 1	qubit 2	qubit 3
state of ancilla	00	10	11	01
Z_1Z_2	$+1$	-1	-1	$+1$
Z_2Z_3	$+1$	$+1$	-1	-1

Table 3.1: State of the ancilla state depending on the error location and measured value of Z_1Z_2 and Z_2Z_3

If we hope to suppress the phase flip error (Z error), we should simply change the basis using the Hadamard gate and have the syndrome measurement of X_1X_2 and X_2X_3.

3.2.2 Shor code and Steane code

The problem of the three qubit code is that we cannot suppress bit flip and phase flip simultaneously. Here, we mention Shor and Steane codes [7, 8], which can

suppress arbitrary errors on a single qubit.

In the Shor code, firstly each qubit is encoded as $|0\rangle \rightarrow |+++\rangle$ and $|1\rangle \rightarrow |---\rangle$, which protects the state from phase flip errors and secondly each $|+(-)\rangle$ state is encoded as $|\pm\rangle \rightarrow \frac{1}{\sqrt{2}}(|000\rangle \pm |111\rangle)$, protecting flip errors, and hence $3 \times 3 = 9$ qubits are required. Then, the state is encoded as

$$|0\rangle \rightarrow \frac{(|000\rangle + |111\rangle)(|000\rangle + |111\rangle)(|000\rangle + |111\rangle)}{2\sqrt{2}} \tag{3.12}$$

$$|1\rangle \rightarrow \frac{(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)}{2\sqrt{2}} \tag{3.13}$$

This encoding corresponds to the syndrome measurements of the observables Z_1Z_2, Z_2Z_3, Z_4Z_5, Z_5Z_6, Z_7Z_8, Z_8Z_9, which can detect flip errors, and $X_1X_2X_3X_4X_5X_6$, $X_4X_5X_6X_7X_8X_9$, detecting phase flip errors. The reason we associate the observables with the code will be clarified in the later section on the stabiliser formalism. Also, as the Y operator can be expressed as a product of X and Z, we can detect Y errors, resulting in the capability to protect against all the arbitrary one-qubit errors.

Furthermore, there is a similar code for protecting arbitrary errors on a single qubit, which only uses 7 qubits (Steane code). This implements a syndrome measurement of observables $IIIXXXX$, $XIXIXIX$, $IXXIIXX$, $IIIZZZZ$, $ZIZIZIZ$, $IZZIIZZ$. Only I and either of Z or X is included, leading to the convenience of this quantum code. Also, it has been shown that the minimum code for protecting arbitrary errors on a single qubit is five qubits code [11, 10] by Bennet and Laflamme indendently, which measures $XZZXI$, $IXZZX$, $XIXZZ$, $ZXIXZ$.

3.2.3 Fault tolerant quantum computation

To implement quantum computation, merely encoding qubits is insufficient. If the noise is amplified during the error correction, it can no longer be corrected.

Therefore, we should construct the quantum circuit on the assumption that errors can occur in every quantum operation, and even in that situation quantum error correction should be successful. This is the concept of fault tolerant quantum computation [12, 13]. While the details of fault tolerance techniques are beyond the scope of this thesis, we note that the community has made considerable effort in the design of circuits for the evaluation of quantum error correction codes that are fault tolerant in the following sense: If a logical qubit without error enters a circuit for error correction (for example, for the Steane 7-qubit code) then no single gate or measurement failure can result in corruption to that logical state to the extent that a further, perfect round of error correction would result in a logical error. This concept is highly relevant to concatenated codes, which we now outline.

3.2.4 Concatenated codes

By recursively repeating the encoding, it is possible to suppress the error rate even further, which is called code concatenation [48]. For example, if we use the 7 qubit Steane code, if the failure probability of a single qubit is p, then the effective error rate in a encoded block becomes cp^2, where c is some constant. If concatenated once, (using 49 qubits), the effective error rate reaches $c(cp^2)^2$, and with L levels of concatenation reaching the error rate $(cp)^{2^L}/c$. Suppose that the number of gates in the quantum circuit is N_{gate}, and the simulation accuracy we hope to achieve is ϵ. In this case, the following ineaquality should be satisfied, as the acceptable inaccuracy per gate should be below ϵ/N_{gate}

$$\frac{(cp)^{2^L}}{c} < \frac{\epsilon}{N_{\text{gate}}} \tag{3.14}$$

Thus, given $p = p_{\text{th}} \equiv 1/c$, we can find L which suffice inequality (3.14).

It is important to note that the number of the gates to realise this accuracy is $O(\text{poly}(\log(N_{\text{gate}}/\epsilon)))$, only increasing poly-logarithmically. The problem of the

concatenation encoding is that as L increases, it is required to have non-local entangling operation, which is extremely difficult to implement in real experiments.

3.2.5 Stabiliser formalism

There exists a extremely useful formalism for expressing error correcting codes, called the stabiliser formalism [49] . In the explanation above, we expressed states using a state vector. Meanwhile, the coding space can be a defined as the eigenspace with eigenvalue 1 of chosen Pauli operators. To give an example, in the three qubit code, syndrome measurement is implemented by measuring observables Z_1Z_2 and Z_2Z_3, which are stabilisers. Also, the operators for syndrome measurements used in Steane code $IIIXXXX$, $XIXIXIX$, $IXXIIXX$, $IIIZZZZ$, $ZIZIZIZ$, $IZZIIZZ$ are also stabilisers to define Steane code. It can be easily shown the eigenspace of eigenvalue 1 corresponding to these operators is a space spanned by vectors $|000\rangle$ and $|111\rangle$, which is obviously the coding space of the three qubit code.

Then, suppose the flip error X_1 occurs. The noise affected state is $X_1|\psi\rangle$ and $Z_1Z_2X_1|\psi\rangle = -X_1Z_1Z_2|\psi\rangle = -X_1|\psi\rangle$, using anti-commuting relationship of X_1 and Z_1Z_2. From this simple argument, by the effect of X_1 noise, the measurement of the stabiliser operator gives the measurement result -1, which signals the existence of errors. Therefore, we can detect the action of operators which takes the state out of the code space as noise, due to the noncommutative relationship between stabiliser operators and noise operators.

3.2.6 Topological error correction codes

Here, we discuss topological error correction codes [50]. Topological error correction codes are codes that have stabilisers which are spatially local and products of Pauli operators, and have translational symmetry. As examples of topological codes, there are the surface code [51], color code [52], 3D cubic code [53], and fractal code [54]. Here we mainly illustrate the surface code.

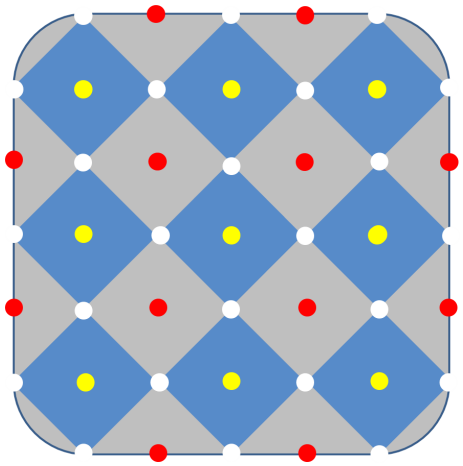


Figure 3.4: The schematic figure for surface code. White circles correspond to data qubits. Blue plaquettes denotes the stabilizer $XXXX$ with the central yellow circles corresponding to the ancilla qubit to measure it. Gray plaquettes denotes the stabilizer $ZZZZ$ with the central red circles corresponding to the ancilla qubit to measure it.

In Fig. 3.4, we show the schematic of the surface code, where white circles are data qubits. The surface code specifies the code space by operators $(XXXX)_i$ and $(ZZZZ)_j$, where i and j corresponds to the plaquettes shown in Fig. 3.4, the central circles of which are qubits for syndrome measurements of these operators. The size of the surface code can be arbitrarily large, resulting in lower failure probability. A key advantage of topological codes to note is that only local operations to plaquette operators are required for error correction, unlike the case using concatenation code, leading to an easier route to the scalability of the quantum processors. We also show the schematic of the smallest two-dimensional colour code in Fig. 3.5. It is worth mentioning that it is equivalent to the Steane 7-qubit code.

It was shown that by using the surface code, quantum computation is possible with the per-step fidelity around 99 %, and eventually in 2014, John Martinis's group reported they reached single qubit gate fidelity 99.92 %, and 2 qubit gate fidelity 99.4 % with 5 qubit device, which is above the threshold [14]. Also, by utilising ion trap system, single qubit fidelity 99.9999 % was realised in 2014 [15], with 2 qubit gate fidelity 99.91 % attained in 2015 [16]. These high fidelities quantum

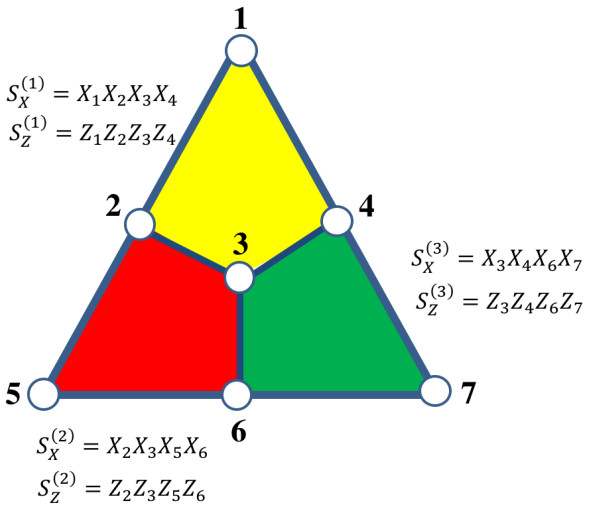


Figure 3.5: The schematic figure for the smallest two-dimensional colour code. White circles correspond to data qubits. $S^{(i)}_{X(Z)}$ $(i = 1, 2, 3)$ in the figure correspond to the stabilisers specifying the code. The smallest two-dimensional colour code is equivalent to the Steane 7-qubit code.

gates are very encouraging, but it remains unclear if the fidelities can be achieved with larger numbers of qubits.

Chapter 4

Hybrid quantum/classical algorithms

4.1 Hybrid quantum/classical algorithms

As seen in the previous section, for fault tolerant quantum devices based on today's prototypes, it is necessary to use hundreds, or even thousands of physical qubits to encode a logical qubit. It has been shown that to execute a conventional quantum algorithm, such as Shor's algorithm, millions of qubits are required [18], which is well beyond the current technologies.

On the other hand, it is believed that it is possible to implement some computation tasks exponentially faster than classical computers, by combining quantum computers and classical computers [19, 20, 21, 22, 23, 24]. The intuition is, as a large portion of the computational burden is assigned to the classical computer, fully coherent deep circuits are not required, unlike quantum algorithms in the early stage of quantum computing, which need quantum error correction. Such algorithms are called hybrid quantum/classical algorithm. Also, to suppress computation errors, we can use quantum error mitigation techniques we will mention later in this thesis, which do not require encoding. This leads to a significant reduction of necessary

qubits required to outperform classical computers. These algorithms are studied intensely as some research groups such as Google and IBM are working towards realising the small quantum devices which may be useful for hybrid quantum/classical algorithm composed of tens or hundreds of qubits. In this section, we illustrate the current development of hybrid quantum/classical algorithms.

4.1.1 Variational quantum eigensolver

The variational Quantum eigensolver (VQE) is an algorithm to evaluate eigenenergies and eigenstates of some Hamiltonian H, theoretically proposed and experimentally demonstrated in the same publication in 2013 for HeH^+ using two qubits with a photonic quantum processor [19]. The VQE can be contrasted with the quantum phase estimation (QPE) algorithm, which also can evaluate eigenenergies of a given Hamlitonian, but requires fully coherent evolution [3]. On the other hand, the essential point of the VQE is that it does not need fully coherent evolution, and so the coherence time does not have to be as long as the one required for QPE algorithm.

A concise explanation of the VQE is as follows [19, 21, 22]. We generate the parametrised trial wave function $|\Phi(\vec{\theta})\rangle$ by using a quantum processor. According to the Rayleigh-Ritz variational principle, the following inequality holds;

$$\langle\Phi(\vec{\theta})|\, H\, |\Phi(\vec{\theta})\rangle \geq E_G, \tag{4.1}$$

where E_G is the lowest energy eigenvalue of the Hamiltonian H, and $\vec{\theta} = (\theta_1, \theta_2, ..., \theta_n)^T$ is a vector of independent parameters. Therefore, by calculating $\langle\Phi(\vec{\theta})|\, H\, |\Phi(\vec{\theta})\rangle$ and optimising parameter $\vec{\theta}$ by using a classical computer, we can find the ground state energy and the parameter $\vec{\theta}$ giving the ground state.

If the trial state is characterised by an exponentially large number of parameters, quantum simulation cannot be implemented with a polynomial number of

operations. Therefore, the quantum state has to be parametrised by a polynomial number of parameters. For general problems this may mean that the true ground state cannot be represented exactlly by the trial state (only approximated). When we parametrise the trial state as

$$|\Phi(\vec{\theta})\rangle = U(\vec{\theta})\,|\Phi_{ref}\rangle\,, \tag{4.2}$$

where $|\Phi_{ref}\rangle$ is a reference state, we refer to $|\Phi(\vec{\theta})\rangle$ as the ansatz state, and $U(\vec{\theta})$ as the ansatz circuit. To have an efficient quantum simulation, it is vital to use a suitable ansatz for the problem. For example, the unitary coupled cluster ansatz is a suitable ansatz for chemistry, because it reflects the detailed structure of the chemistry problem [55, 19, 56]. However, the structure of the quantum circuit tends to be complicated when the unitary coupled cluster ansatz is used, and hard to implement by using the current noisy quantum devices with limited connectivity. This can be circumvented using so called "hardware efficient ansätze" , which are experimentally more feasible to implement [19, 23, 20]. However, hardware efficient ansatz does not consider any details of simulated quantum systems, therefore it seems unsuitable for the simulation of large molecules [57].

To simulate molecules, firstly we consider the fermionic Hamiltonian

$$H_{\mathrm{f}} = \sum_{ij} t_{ij} c_i^\dagger c_j + \sum_{ijkl} u_{ijkl} c_i^\dagger c_k^\dagger c_l c_j, \tag{4.3}$$

where $c_j^\dagger$ denotes a creation operator for a fermion in the j-th orbital, and t_{ij} and u_{ijkl} are the one and two electron interactions, which can be efficiently calculated by integrating the basis set wave functions [58, 59]. We map this Hamiltonian to qubits, by using second quantised encoding methods, such as Jordan-Wigner, parity, and Bravi-Kitaev encodings [3, 60, 61]. For example, by using Jordan-

Wigner transformation

$$c_i^\dagger \to I^{\otimes i-1} \otimes \sigma^- \otimes \sigma_z^{\otimes N-i} \tag{4.4}$$

$$c_i \to I^{\otimes i-1} \otimes \sigma^+ \otimes \sigma_z^{\otimes N-i} \tag{4.5}$$

where I is an identity operator, $\sigma^\pm$ is a raising (lowering) ladder operator, and σ_z is a Pauli Z operator, we can map the fermion Hamiltonian to the form

$$H_\mathrm{q} = \sum_\alpha f_\alpha \sigma_\alpha. \tag{4.6}$$

Here, $\sigma_\alpha = \sigma_1^{\alpha_1} \otimes \sigma_2^{\alpha_2} \otimes ... \sigma_M^{\alpha_M}$ with Pauli operators $\sigma_j^{\alpha_j} \in \{I^j, \sigma_x^j, \sigma_y^j, \sigma_z^j\}$, which can be described using a quantum circuit composed of qubits. Then, the cost function corresponding to the energy to be minimized $E(\vec{\theta}) = \langle\Phi(\vec{\theta})| H_\mathrm{q} |\Phi(\vec{\theta})\rangle = \sum_\alpha f_\alpha \langle\Phi(\vec{\theta})| \sigma_\alpha |\Phi(\vec{\theta})\rangle$ can be computed by calculating the expectation value of Pauli operators σ_α in the quantum circuit and summing up with appropriate coefficients f_α. The measurement of σ_α can be parallelised.

The time required after state preparation until measurement is $O(1)$ and we can reinitialise the quantum circuit. This implies that the VQE does not require as long a coherence time as is necessary for phase estimation [3]. Once we have obtained $E(\vec{\theta})$, we update the parameters by using a classical computer. A fast and accurate optimisation method should be chosen, so that the solution can reach the global minimum (or feasible solution) in a reasonable time.

The downside of this algorithm is the total number of operations is $O(\epsilon^{-2})$ for precision ϵ due to shot noise [19, 22], which is quadratically worse than phase estimation. In addition, the solution is not necessarily correct: the solution might be a local minimum of the ansatz space.

The VQE was experimentally demonstrated by several groups [19, 20, 62, 63, 64, 65, 66]. For example, it was demonstrated for BeH_2, which has a far more complex structure than HeH^+, using 6 qubits with superconducting circuits by IBM

group [20]. In Fig. 4.1, we show the experimental results obtained from superconducting system [20].

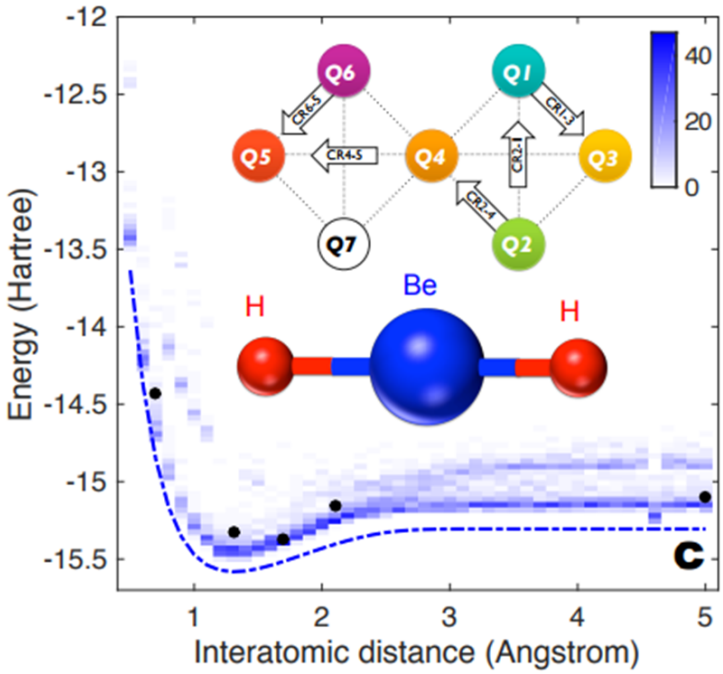


Figure 4.1: Ground state energy of BeH_2 depending on the inter-atomic distance, obtained using the VQE with a superconducting system. Black circles represent experimental results, the dotted line shows the exact theoretical curve, and density plots correspond to numerical simulations of this experiment. This figure is reproduced from Ref. [20].

Furthermore, the VQE can also be used for finding an approximate solution of classical optimization problems such as MaxCut. This is called the quantum approximate optimization algorithm [23].

It is worth noting that the VQE is inherently robust to coherent errors, such as qubit over rotation [22]. Suppose the ansatz space for the VQE is described by the unitary operator $U_A(\vec{\theta})$. Due to the effect of the noise, the actual operation is described by some different unitary operator $\tilde{U}_A(\vec{\theta})$. If there exists the parameter $\vec{\theta} + \vec{\alpha}$ such that $||U_A(\vec{\theta}) - \tilde{U}_A(\vec{\theta} + \vec{\alpha})|| < \epsilon$ for sufficiently small $\epsilon > 0$, $\tilde{U}_A(\vec{\theta} + \vec{\alpha})$ also gives the correct ground state energy, hence the parameter to give the ground state just changes to $\vec{\theta}_{\text{opt}} + \vec{\alpha}$ from $\vec{\theta}_{\text{opt}}$.

Furthermore, consider the case the ansatz should conserve some quantity, such as the total electron number of the molecule [22]. If an error changes the conserved quantity, the argument discussed above cannot be applied. However, this problem can be resolved by optimising the modified cost function

$$\langle\Phi(\vec{\theta})|\, H + \sum_j \beta_j (Q_j - q_j I)^2 \,|\Phi(\vec{\theta})\rangle\,, \tag{4.7}$$

where $\{Q_j\}$ is the set of the operators for conserved quantities, q_j is the corresponding ideal expectation value of Q_j, and β_j is the penalty coefficient which should be sufficiently large. For instance, if the ansatz space of interest is a subspace of the eigenspace of the fixed electron number such as unity, we should set $q_j = 1$. Now, if the state gets in the wrong subspace whose electron number is 2, the cost function increases by β_j. Therefore, the parametrised state tries to avoid subspace other than $q_j = 1$. Consequently, we can explore the parameter space whose electron number is unity, i.e., the eignesubspace of electron number operator, whose eigenvalue is unity.

Moreover, we can also utilise quantum error mitigation techniques to suppress physical and even algorithmic error, which we will consider in a later section [24, 28, 29, 30].

4.1.2 Real and imaginary time evolution quantum simulator

There are also variational quantum algorithms for simulating real and imaginary time evolution, introduced in Ref. [24] and [46], respectively. The real time evolution is described by the Schrödinger equation,

$$\frac{d\,|\psi(t)\rangle}{dt} = -iH\,|\psi(t)\rangle\,, \tag{4.8}$$

with hermitian Hamiltonian H. Instead of directly simulating the real time dynamics with the Hamiltonian simulation algorithms [2, 67, 68, 69], the variational method assumes that the quantum state $|\psi(t)\rangle$ is prepared by a parametrised quantum circuit, $|\varphi(\vec{\theta}(t))\rangle = U_N(\theta_N)\dots U_k(\theta_k)\dots U_1(\theta_1)\,|\bar{0}\rangle$ with each gate $U_k(\theta_k)$ controlled by the real parameter θ_k. Here, we denote $\vec{\theta} = (\theta_1, \theta_2, \dots, \theta_N)$.

There are three variational principles which can be used in classical simulation of quantum dynamics in order to map the equation of the wave function to the parameters: The Dirac and Frenkel variational principle [70, 71], McLachlan's variational principle [72], and the time-dependent variational principle [73, 74]. The solution of the Dirac and Frenkel variational principle involve complex numbers, although parameters for gates in a quantum variational algorithm are real. The time-dependent variational principle gives the unstable evolution of the parameters, and no physically feasible equation is obtained when it is applied to the simulation of the density matrix. McLachlan's principle is the most consistent variational principle for the cases we will mention in this section, so that we use McLachlan's variational principle [75].

According to McLachlan's variational principle [72], the real time dynamics of $|\psi(t)\rangle$ can be mapped to the evolution of the parameters $\vec{\theta}(t)$ by minimising the distance between the ideal evolution and the evolution induced of the parametrised trial state [24],

$$\delta \| (\partial/\partial t + iH)\,|\varphi(\vec{\theta}(t))\rangle \,\| = 0, \tag{4.9}$$

where $\| |\varphi\rangle \| = \langle\varphi|\varphi\rangle$ is the norm of $|\varphi\rangle$, and

$$\begin{aligned}
&\|(\partial/\partial t + iH)\,|\varphi(\vec{\theta}(t))\rangle\,\| = ((\partial/\partial t + iH)\,|\phi(\vec{\theta}(t))\rangle)^{\dagger}(\partial/\partial t + iH)\,|\phi(\vec{\theta}(t))\rangle) \\
&= \sum_{k,j} \frac{\partial\,\langle\varphi(\vec{\theta}(t))|}{\partial\theta_k}\frac{\partial\,|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_j}\dot{\theta}_k^*\dot{\theta}_j \\
&+ i\sum_{k} \frac{\partial\,\langle\varphi(\vec{\theta}(t))|}{\partial\theta_k} H\,|\varphi(\vec{\theta}(t))\rangle\,\dot{\theta}_k^* - i\sum_{k}\langle\varphi(\vec{\theta}(t))|\,H\frac{\partial\,|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_k}\dot{\theta}_k \\
&+ \langle\varphi(\vec{\theta}(t))|\,H^2\,|\varphi(\vec{\theta}(t))\rangle\,.
\end{aligned} \tag{4.10}$$

Suppose $\dot{\theta}_k$ can be complex, then we have

$$\begin{aligned}
\delta\|(\partial/\partial t + iH)\,|\varphi(\vec{\theta}(t))\rangle\,\| = \sum_k \Bigg(&\left(\sum_j \frac{\partial\,\langle\varphi(\vec{\theta}(t))|}{\partial\theta_k}\frac{\partial\,|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_j}\dot{\theta}_j + i\frac{\partial\,\langle\varphi(\vec{\theta}(t))|}{\partial\theta_k}H\,|\varphi(\vec{\theta}(t))\rangle\right)\delta\dot{\theta}_k^*, \\
&+\left(\sum_j \frac{\partial\,\langle\varphi(\vec{\theta}(t))|}{\partial\theta_j}\frac{\partial\,|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_k}\dot{\theta}_j^* - i\,\langle\varphi(\vec{\theta}(t))|\,H\frac{|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_k}\right)\delta\dot{\theta}_k\Bigg).
\end{aligned} \tag{4.11}$$

Now, assuming $\dot{\theta}_k$ is real, we have

$$\delta\|(\partial/\partial t + iH)\,|\varphi(\vec{\theta}(t))\rangle\,\| = \sum_k \Bigg(\sum_j \left(\frac{\partial\,\langle\varphi(\vec{\theta}(t))|}{\partial\theta_k}\frac{\partial\,|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_j} + \frac{\partial\,\langle\varphi(\vec{\theta}(t))|}{\partial\theta_j}\frac{\partial\,|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_k}\right)\dot{\theta}_j\delta\dot{\theta}_k \tag{4.12}$$

$$+ i\left(\frac{\partial\,\langle\varphi(\vec{\theta}(t))|}{\partial\theta_k}H\,|\varphi(\vec{\theta}(t))\rangle - \langle\varphi(\vec{\theta}(t))|\,H\frac{\partial\,|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_k}\right)\delta\dot{\theta}_k\Bigg) = 0. \tag{4.13}$$

Thus, we obtain

$$\sum_j M_{k,j}\dot{\theta}_j = V_k, \tag{4.14}$$

with coefficients

$$\begin{aligned}
M_{k,j} &= \mathrm{Re}\left(\frac{\partial\,\langle\varphi(\vec{\theta}(t))|}{\partial\theta_k}\frac{\partial\,|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_j}\right), \\
V_k &= \mathrm{Im}\left(\langle\varphi(\vec{\theta}(t))|\,H\frac{\partial\,|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_k}\right).
\end{aligned} \tag{4.15}$$

It is worth noting that, in order to get optimal performance, we need an extra parameter corresponding to the (meaningless) global phase [75]. We can also derive similar equations by using the Dirac and Frenkel variational principle and the time-dependent variational principle.

The imaginary time evolution is used as an efficient optimisation routine to find the ground state of molecular Hamiltonian, or the Hamiltonian in which the solution of the optimisation problem is encoded [46]. For imaginary time evolution, the normalised Wick-rotated Schrödinger equation is obtained by replacing $t = i\tau$ in Eq. (4.8) [76],

$$\frac{d\left|\psi(\tau)\right\rangle}{d\tau} = -(H - \langle H\rangle)\left|\psi(\tau)\right\rangle. \tag{4.16}$$

Applying a similar procedure for real time evolution, the imaginary time evolution is mapped to the evolution of the parameters via McLachlan's principle,

$$\delta\|(\partial/\partial\tau + H - \langle H\rangle)\left|\varphi(\vec{\theta}(\tau))\right\rangle\| = 0. \tag{4.17}$$

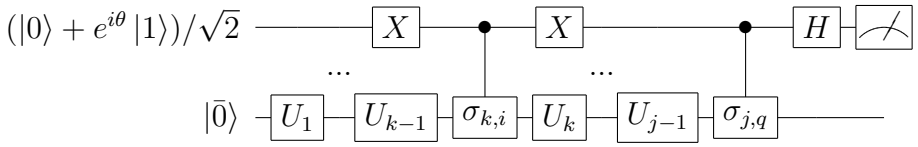
The evolution of the parameters is

$$\sum_j M_{k,j}\dot{\theta}_j = C_k, \tag{4.18}$$

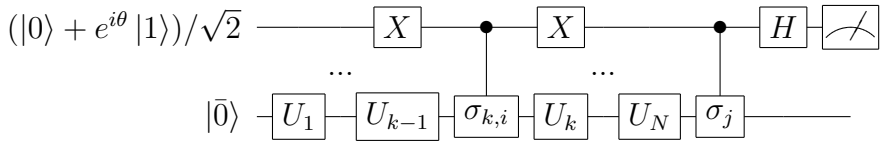
with M given in Eq. (4.15) and C defined by

$$C_k = -\mathrm{Re}\left(\left\langle\varphi(\vec{\theta}(\tau))\right| H\frac{\partial\left|\varphi(\vec{\theta}(\tau))\right\rangle}{\partial\theta_k}\right). \tag{4.19}$$

The M, V, and C terms can be efficiently measured with quantum circuits. Considering gate based circuits, the derivative of the parameterised gate can be



(a)



(b)

Figure 4.2: Quantum circuits that evaluate (a) $\mathrm{Re}(e^{i\theta}\langle\bar{0}|U_{k,i}^{\dagger}U_{j,q}|\bar{0}\rangle)$ and (b) $\mathrm{Re}(e^{i\theta}\langle\bar{0}|U_{k,i}^{\dagger}\sigma_j U|\bar{0}\rangle)$ [24, 46, 75].

expressed as

$$\frac{\partial U_k}{\partial \theta_k} = \sum_i g_{k,i} U_k \sigma_{k,i}, \tag{4.20}$$

where $\sigma_{k,i}$ are unitary operators and $g_{k,i}$ are complex coefficients. The derivative of the trial state can be written as

$$\frac{\partial |\varphi(\vec{\theta}(t))\rangle}{\partial \theta_k} = \sum_i g_{k,i} U_{k,i} |\bar{0}\rangle , \tag{4.21}$$

where

$$U_{k,i} = U_N U_{N-1} \cdots U_{k+1} U_k \sigma_{k,i} \cdots U_2 U_1. \tag{4.22}$$

The $M_{k,j}$ terms can be expressed as

$$M_{k,j} = \sum_{i,q} \mathrm{Re}\left(g_{k,i}^{*} g_{j,q} \langle\bar{0}| U_{k,i}^{\dagger} U_{j,q} |\bar{0}\rangle \right). \tag{4.23}$$

Similarly, considering a sparse Hamiltonian with decomposition $H = \sum_j f_j \sigma_j$, $f_j \in$

$\mathbb{R}$, we have C_k and V_k as

$$\begin{aligned} V_k &= \sum_{i,j} \mathrm{Re}\left(i g^*_{k,i} f_j \langle \bar{0} | U^\dagger_{k,i} \sigma_j U | \bar{0} \rangle \right), \\ C_k &= -\sum_{i,j} \mathrm{Re}\left(g^*_{k,i} f_j \langle \bar{0} | U^\dagger_{k,i} \sigma_j U | \bar{0} \rangle \right), \end{aligned} \tag{4.24}$$

All the M, C, and V terms can be written in the form

$$a\mathrm{Re}\left(e^{i\theta} \langle \bar{0} | \mathcal{V} | \bar{0} \rangle \right),$$

where $a, \theta \in \mathbb{R}$ depend on the coefficients, and $\mathcal{V}$ is a unitary operator of either $U^\dagger_{k,i} U_{j,q}$ or $U^\dagger_{k,i} \sigma_j U$. We can calculate M, C, and V by using the quantum circuit shown in Fig. 4.2.

Then, the flow of the algorithm for simulating the real time evolution is

1. Determine $\vec{\theta}(0)$

2. Send it to the quantum computer and calculate $M_{k,j}$ and V_k.

3. Send the calculated $M_{k,j}$ and V_k to classical computer and obtain $\dot{\vec{\theta}}(0)$ by solving Eq. (4.15).

4. We have $\vec{\theta}(\delta t) = \vec{\theta}(0) + \delta t \dot{\vec{\theta}}(0)$

5. Send $\vec{\theta}(\delta t)$ to the quantum computer and calculate $M_{k,j}$ and V_k.

By iterating process 3 - 5, we can obtain the state at time t after real time evolution. By changing V_k to C_k, we can simulate the imaginary time evolution.

We show the numerical simulation result of the variational imaginary simulation in Fig. 4.3 for H_2 with 2 qubits [46]. It successfully simulates the exact imaginary time evolution. Also, it has been verified that imaginary time evolution was useful for finding the ground state of LiH with 6 qubits compared with other classical optimisation methods, with regard to the convergence time for optimisation and the probability for discovering the ground state [46]. In the simualtion, the hardware efficient ansatz was used.

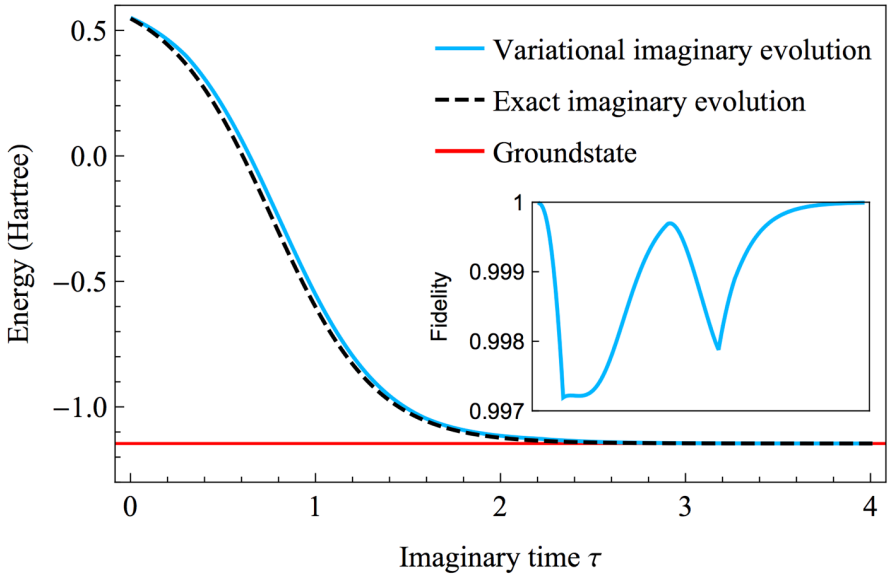


Figure 4.3: Numerical simulation of the energy of H_2 under imaginary time simulation. The blue line denote the variational imaginary evolution, with the dashed black line corresponding to the exact imaginary time evolution. The red line is the exact ground state energy of H_2. The inset figure shows the fidelity of the variational imaginary evolution and the exact imaginary evolution. The plots are reproduced from Ref. [46].

Chapter 5

Quantum error mitigation

5.1 Error mitigation techniques

The idea and techniques about quantum error mitigation described in this chapter are built upon by author's research as explained in chapter 7. It is essential to suppress the computation errors in quantum algorithms. Due to imperfections of gate operations, such as decoherence, errors can accumulate in the quantum circuit, so that the noisy output state ρ_{noise} becomes

$$\rho_{\text{noise}} = \mathcal{N}_{N_{\text{tot}}} \circ \mathcal{G}_{N_{\text{tot}}} \circ \cdots \circ \mathcal{N}_1 \circ \mathcal{G}_1(\rho_{\text{init}}), \tag{5.1}$$

where the ideal operation can be expressed as $\rho_{\text{ideal}} = \mathcal{G}_N \circ \mathcal{G}_{N-1} ... \mathcal{G}_1(\rho_{\text{init}})$. Here, $\mathcal{N}_k$ is the noise channel accompanying the k^{th} ideal gate operation $\mathcal{G}_k$, N_{tot} is the number of gates, and ρ_{init} is the initial input state for the quantum circuit. As explained in section 3.2, the noisy output state can be corrected with fault tolerant error correction that utilises extra qubits to detect errors and correct the state. However, fault tolerant error correction is considerably costly and is hard to realise with current quantum hardware as seen in the previous section. Therefore, it is desirable in hybrid quantum/classical algorithms to use all qubits as logical qubits without the conventional error correction technique, as the number of qubits is restricted. The

hybrid quantum/classical algorithms rely on the ability to calculate the expectation value of some observables using quantum processors [19, 24, 75, 46, 22]. For noisy intermediate-scale quantum devices with a restricted number of qubits, quantum error mitigation methods mentioned in this section require no extra qubits and can suppress errors for expectation values of observables with simple post-processing of different runs of the quantum circuits. More concretely, although we cannot recover the ideal output state ρ_{ideal} with quantum error mitigation technique, we can estimate the ideal expectation value $\langle \hat{A}_{\text{ideal}} \rangle = \text{Tr}(\rho_{\text{ideal}} \hat{A})$ for observables $\hat{A}$ [24, 28, 29].

5.2 Extrapolation technique

Now, we explain the Richardson extrapolation error mitigation methods [24, 28, 29]. These methods were proposed by Li & Benjamin and an IBM group, independently [24, 28]. We show the schematic figure for showing the linear extrapolation (or first order Richrdson extrapolation) in Fig. 5.2.

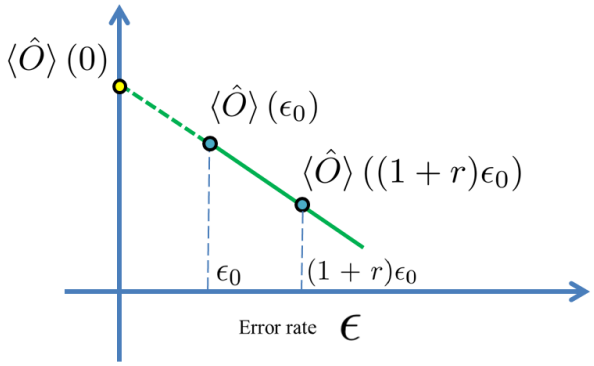


Figure 5.1: The schematic figure showing extrapolation technique.

For a noisy circuit with error strength ε, the noisy output state ρ_ε deviates from the ideal noiseless state. For example, for stochastic errors

$$\mathcal{N}_k = (1-p)\mathcal{I} + p\mathcal{N}'_k, \tag{5.2}$$

we can set $\varepsilon \propto p$, where $\mathcal{I}$ is the identity map, $\mathcal{N}_k'$ is a noise map, and $p \in [0,1]$ denotes the strength of the noise. The noisy expectation value $\langle \hat{A} \rangle (\varepsilon)$ can be regarded as a function of ε and it can be Taylor expanded according to different orders of ε,

$$\langle \hat{A} \rangle (\varepsilon) = \langle \hat{A} \rangle (0) + \sum_{j=1}^{n} A_j \varepsilon^j + O(\varepsilon^{n+1}), \tag{5.3}$$

where A_j is independent of ϵ and $\langle \hat{A} \rangle (0)$ is the noiseless expectation value. For a considerably small ε, we have $\langle \hat{A} \rangle (\varepsilon) \approx \langle \hat{A} \rangle (0)$. With several different noisy expectations $\langle \hat{A} \rangle (\varepsilon)$ for different ε, error mitigation is able to infer the noiseless expectation value $\langle \hat{A} \rangle (0)$ to a higher accuracy.

Suppose the expectation value of the observable $\langle \hat{A} \rangle (\varepsilon)$ is measured for several rescaled noise rates $a_j \varepsilon$ with $a_0 = 1 < a_1 < a_2 ... < a_n$, which can be achieved by increasing the physical noise, by rescaling the Hamiltonian parameters and evolution time of the system [28]. We will explain the method to boost the physical error rate later. Then we can estimate $\langle \hat{A} \rangle (0)$ by Richardson extrapolation method [77],

$$\begin{aligned} \langle \hat{A} \rangle_{est} (0) &= \sum_{i=0}^{n} \gamma_i \langle \hat{A} \rangle (a_i \varepsilon), \\ &= \sum_{i=0}^{n} \gamma_i \langle \hat{A} \rangle (0) + \sum_{j=1}^{n} A_j \varepsilon^j \sum_{i=0}^{n} \gamma_i a_i^j + O(\varepsilon^{n+1}) \\ &= \langle \hat{A} \rangle (0) + O(\varepsilon^{n+1}), \end{aligned} \tag{5.4}$$

where γ_i is chosen such that $\sum_{i=0}^{n} \gamma_i = 1$, $\sum_{i=0}^{n} \gamma_i a_i^j = 0$ for $j = 1, 2, \ldots, n$, so that higher order terms of ε cancels out as $\sum_{j=1}^{n} A_j \varepsilon^j \sum_{i=0}^{n} \gamma_i a_i^j = 0$. By extrapolating n points with different error strengths, we can accurately estimate $\langle \hat{A} \rangle (0)$ and suppress the error to $O(\varepsilon^{n+1})$. When $n = 1$, we call it linear or two-point extrapolation; when $n = 2$, we call it 2nd order Richardson or three-point extrapolation.

The variance of the estimation $\langle \hat{A} \rangle_{est}(0)$ is

$$\mathrm{Var}(\langle \hat{A} \rangle_{est}(0)) = \sum_{i=0}^{n} \gamma_i^2 \mathrm{Var}(\langle \hat{A} \rangle_{est}(a_i \varepsilon)). \tag{5.5}$$

Suppose the variance $\mathrm{Var}(\langle \hat{A} \rangle_{est}(a_i \varepsilon))^2$ are similar for different error strengths, the variance of the estimation $\langle \hat{A} \rangle_{est}(0)$ is $\sum_{i=1}^{n} \gamma_i^2$ times larger than the variance of each measurement $\hat{A}(\varepsilon)$. Therefore, to achieve the same shot noise of each measurement $\hat{A}(\varepsilon)$, we need to run the circuit $\sum_{i=1}^{n} \gamma_i^2$ more times and we denote

$$\Gamma_{phys} = \sum_{i=1}^{n} \gamma_i^2 \tag{5.6}$$

as the cost of physical error mitigation. Note that, it is numerically confirmed that Γ_{phys} generally increases exponentially to n in Ref. [78]. We can thus only choose a small constant value of n in practice in order to avoid such an exponentially increasing cost.

For example, when we estimate $\langle \hat{A} \rangle_{est}(0)$ by using linear extrapolation with error rates ε and $r\varepsilon$ $(r > 1)$, $\langle \hat{A} \rangle_{est}(0) = r/(r-1) \langle \hat{A} \rangle(\epsilon) - 1/(r-1) \langle \hat{A} \rangle(r\epsilon)$, and assuming $\mathrm{Var}\big[\langle \hat{A} \rangle(\epsilon) \big] = \mathrm{Var}\big[\langle \hat{A} \rangle(r\epsilon) \big]$

$$\mathrm{Var}\big[\langle \hat{A} \rangle_{est}(0) \big)] = \frac{1 + r^2}{(1-r)^2} \mathrm{Var}\big[\hat{A}(\varepsilon) \big], \tag{5.7}$$

thus we obntain

$$\Gamma_{phys} = \frac{1 + r^2}{(1-r)^2}, \tag{5.8}$$

which is a monotonically decreasing function of r. The resource cost can be reduced with a larger r, however with an increase of the estimation error. Therefore, one needs to optimise r by taking into account the shot noise due to finite samples and the error due to the extrapolation method.

Firstly, in Fig. 5.2, we show the result from the IBM team [28] where the extrapolation enhances the accuracy of the expectation value of a randomly chosen Pauli operator from a certain quantum circuit, for three types of noise: depolarising noise, mixture of damping and dephasing noise, and non-Markovian noise, where each qubit is coupled to its single qubit bath b_i via a Hamiltonian $V_i = 1/2X_i \otimes X_{b_i} + 1/2Z_{b_i}$. We can see that the more the sampling point n is, the better accuracy we obtain. However, note that as we increase n, the variance increases exponentially, therefore there is a tradeoff between the accuracy of the expectation value and the sampling error.

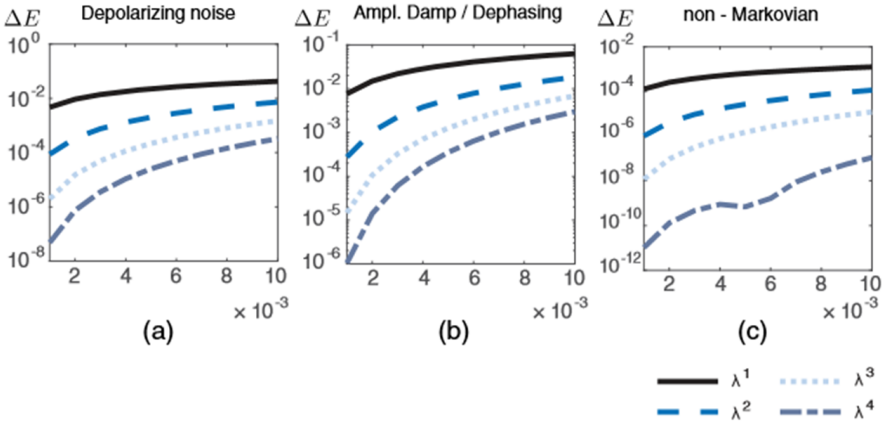


Figure 5.2: The performance of extrapolation technique for three types of noise models: depolarising noise, mixture of damping and dephasing noise, and non-Markovian noise, where qubits are coupled with some qubit bath. The vertical axis shows the deviation from the ideal value, and horizontal axis is noise rate. n of λ^n in the figure denote the number of sample points for extrapolation. The larger n is, the smaller the deviation becomes dramatically. This figure is reproduced from Ref [28].

Also, Li & Benjamin applied linear extrapolation with the real time quantum simulator [24]. We show how the efficient variational quantum simulator plus extrapolation scheme works in the regime $\langle \hat{A} \rangle (\epsilon)$ behaves linearly in Fig. 5.2. Three different simulation methods are compared. The simulated system is three qubit 2D Ising model with transverse fields. The metric of the computation error is the trace distance from the ideal state. Trotterrisation of an evolution operator is a standard

method to simulate the dynamics of quantum systems, and physical errors accumulate as shown in Fig. 5.2. Also, while the errors accumulate in the variational real time simulation algorithm, we can combine it with extrapolation technique to mitigate the error in a low level.

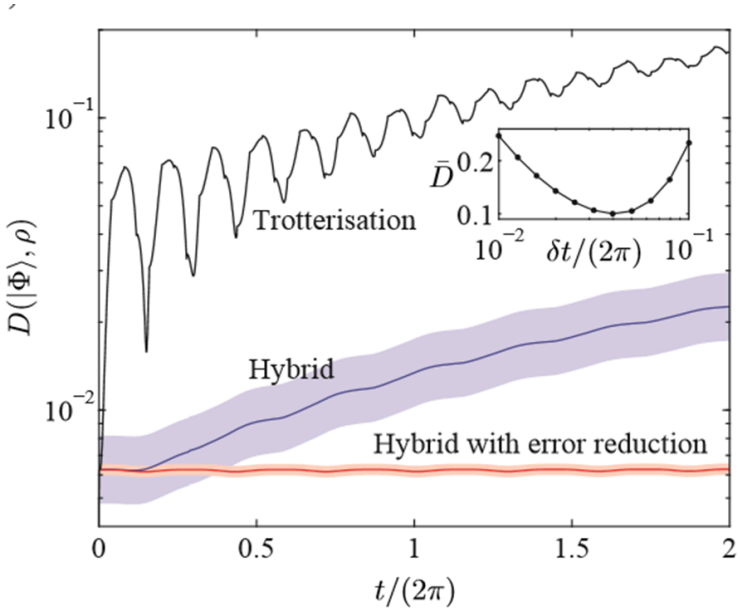


Figure 5.3: Numerical simulations of trace distance of ideal state and simulated states, using conventional Trotterization, an efficient variational quantum simulator (hybrid), and efficient variational quantum simulator plus extrapolation technique (hybrid with error reduction) in the presence of depolarizing channel, whose error rate is 0.01 percent for single qubit gates and 0.1 percent for 2 qubits gate. Vertical axis is trace distance and horizontal axis is time. It is obvious that the hybrid algorithm with error mitigation can suppress the error. This figure is reproduced from Ref. [24].

Recently, the error mitigation technique using the Richardson extrapolation was implemented experimentally by an IBM group [79]. By using the noisy data and extrapolating them, they could reproduce the ground state energies of the H_2 (with 2 qubits) and LiH (with 4 qubits) molecules, utilising the hardware efficient ansatz with the superconducting circuit qubits. Also, the ground state of an anti-ferromagnetic four-qubit Heisenberg model on a square lattice with a transversal magnetic field was simulated. The result obtained from the extrapolation tech-

nique is fed to the optimisation routine and this process is iterated. As a classical optimisation method, simultaneous perturbation stochastic approximation (SPSA) algorithm was used.

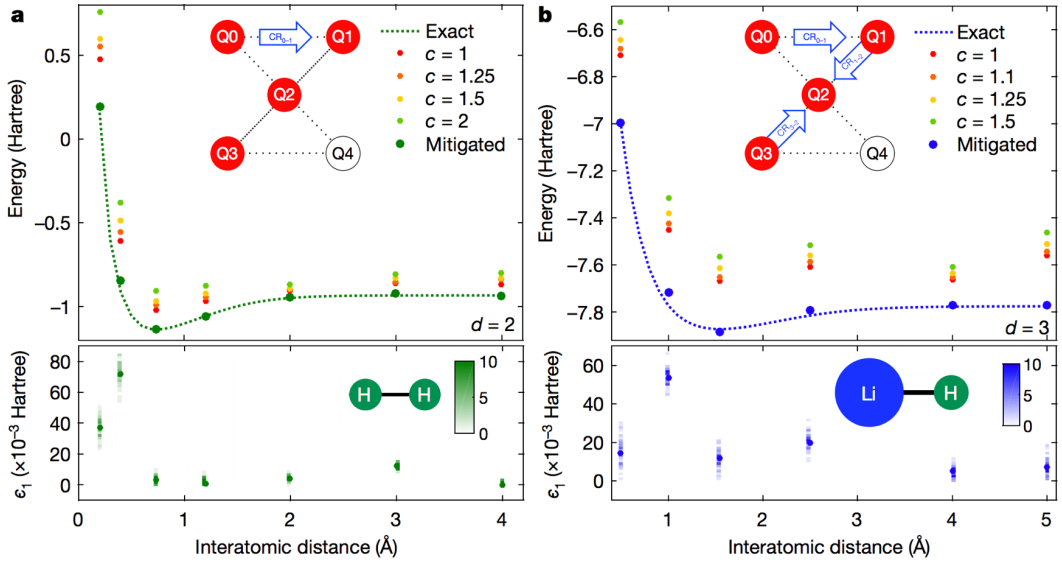


Figure 5.4: The experimental demonstration of extrapolation. The result of energies (a) H_2 and (b) LiH, depending on the interatomic distance. The inset shows the connectivity of qubits. we can see that more accurate energy can be obtained due error mitigation. For (a), the depth of the quantum circuit is 2, and for (b) 3. This figure is reproduced from [79].

5.2.1 The method to boost the physical error rate

By rescaling the Hamiltonian parameters and the evolution time, we can effectively boost the error rate [28]. We outline how this can be done in a system where the quantum gates are implemented by applying control pulses as in the superconducting system in Ref [28]. Now, we assume the quantum system (also including the system's quantum circuits) is described by stochastic equation for density matrix

$$\frac{d}{dt}\rho_\lambda = -i[H(t), \rho_\lambda] + \lambda\mathcal{L}(\rho_\lambda), \tag{5.9}$$

where $H(t)$ is a time dependent Hamiltonian, $\mathcal{L}$ is a stochastic effect, which can be time dependent, and λ is a noise strength. The goal is effectively increasing a noise strength λ to $c\lambda$, where $c > 1$. Now, we rescale the Hamiltonian to $\frac{1}{c}\, H(\frac{t}{c})$, which is experimentally achieved by reducing the pulse strength to $1/c$ times and widening the pulse width by c times. Now, suppose the state ρ_λ evolves in this setup, and we have

$$\frac{d}{dt}\rho'_\lambda = -i\left[\frac{1}{c}H\left(\frac{t}{c}\right), \rho'_\lambda\right] + \lambda\mathcal{L}(\rho'_\lambda). \tag{5.10}$$

Now we show $\rho'_\lambda(ct) = \rho_{c\lambda}(t)$, which means by evolving the state under rescaled Hamiltonian for c times longer, we can effectively boost the error by factor of c. We assume $\mathcal{L}$ is independent of rescaling and $H(t)$. Now we define $\rho_a(t) := \rho'_\lambda(ct)$, set $t' = ct$ and we have

$$\begin{aligned}
\frac{d}{dt}\rho_a(t) &= \frac{dt'}{dt}\frac{\partial}{\partial t'}\rho'_\lambda(t')|_{t'=ct} && (5.11)\\
&= c\Big\{ -i\Big[\ \frac{1}{c}H\left(\frac{t'}{c}\right), \rho'_\lambda(t')\Big] + \lambda\mathcal{L}(\rho'_\lambda(t'))\Big\}\Big|_{t'=ct} && (5.12)\\
&= -i[H(t), \rho'_\lambda(ct)] + c\lambda\mathcal{L}(\rho'_\lambda(ct)) && (5.13)\\
&= -i[H(t), \rho_a(t)] + c\lambda(\rho_a(t)). && (5.14)
\end{aligned}$$

On the other hand, the equation $\rho_{c\lambda}(t)$ follows is

$$\frac{d}{dt}\rho_{c\lambda}(t) = -i[H(t), \rho_{c\lambda}(t)] + c\lambda\mathcal{L}(\rho_{c\lambda}(t)). \tag{5.15}$$

$$\tag{5.16}$$

Also $\rho_a(0) = \rho_{c\lambda}(0)$, and we can see $\rho_a(t)$ and $\rho_{c\lambda}(t)$ has the same time derivative equation. Then we have $\rho'_\lambda(ct) = \rho_{c\lambda}(t)$.

5.3 Quasi-probability method

An IBM group also proposed another type of error mitigation technique [28], called quasi-probability method or probabilistic error cancellation method. The main idea is to simulate the ideal unitary processes by randomly varying the noisy gates. Suppose that formally the ideal unitary process $\mathcal{U}$ can be expressed using noisy operations $\mathcal{K}_i$

$$\mathcal{U} = \sum_i \eta_i \mathcal{K}_i \tag{5.17}$$

$$= \gamma \sum_i p_i \mathrm{sgn}(\eta_i) \mathcal{K}_i \tag{5.18}$$

where $\sum_i \eta_i = 1$, $\gamma = \sum_i |\eta_i|$, $p_i = \frac{|\eta_i|}{\gamma}$, and generally $\gamma \geq 1$. Here, $\mathcal{U}$ and $\mathcal{K}_i$ are superoperators, and $\mathcal{U}(\rho) = U\rho U^\dagger$, where U is a unitary operator corresponding to $\mathcal{U}$. Thus, by randomly generating noisy operation $\mathcal{K}_i$ with probability p_i, and attaching the sign $\mathrm{sgn}(\eta_i)$ to the measurement result when measuring some observable, averaging the results, and multiplying γ to that value, we can obtain the error free expectation value of the observable corresponding to the ideal unitary process $\mathcal{U}$. Note that the variance becomes γ^2 times greater, and we need to have γ^2 greater samples to have the same accuracy as the ideal case, which is the cost of this method.

For example, suppose that the gate operation $\mathcal{U}$ is affected by depolarizing errors as $\mathcal{D}\mathcal{U}$. The aim is to remove the effect of the noisy operator $\mathcal{D}$, and this can be done by applying $\mathcal{D}^{-1}$ mathematically. Now, the depolarising channel can be written as

$$\mathcal{D}(\rho) = (1 - \frac{3}{4}p)\rho + \frac{p}{4}(X\rho X + Y\rho Y + Z\rho Z), \tag{5.19}$$

and the inverse channel can be exressed as

$$\mathcal{D}^{-1}(\rho) = (1 + \frac{3p}{4(1-p)})\rho - \frac{p}{4(1-p)}(X\rho X + Y\rho Y + Z\rho Z) \tag{5.20}$$

$$= \gamma_{\mathcal{D}^{-1}}[p_1\rho - p_2(X\rho X + Y\rho Y + Z\rho Z)], \tag{5.21}$$

where $\gamma_{\mathcal{D}_{-1}} = (p+2)(2-2p) > 1$, $p_1 = (4-p)/(2p+4)$, and $p_2 = p/(2p+4)$.

Therefore, $\mathcal{U}$ can be described as

$$\mathcal{U} = \mathcal{D}^{-1}\mathcal{D}\mathcal{U} \tag{5.22}$$

$$= \gamma_{\mathcal{D}^{-1}}[p\mathcal{I}\mathcal{D}\mathcal{U} - p_2(\mathcal{X}\mathcal{D}\mathcal{U} + \mathcal{Y}\mathcal{D}\mathcal{U} + \mathcal{Z}\mathcal{D}\mathcal{U})], \tag{5.23}$$

where $\mathcal{I}$, $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{Z}$ are superoperators for I ,X, Y, and Z operators respectively. Then, from this expression, quasi-probability operation is constructed. We show the schematic figure for the construction of the quasi-priobability method in Fig. 5.5.

It has been shown that amplitude damping error can be suppressed using the identity, S gate, S^{-1}, and the initialization to $|0\rangle$. For two qubits gate errors, the necessary operations to suppress the error can be expressed as the tensor product of the operations for single qubit gate.

Suppose that this error mitigation technique is applied in a quantum circuit, and these error mitigation operations are applied after each gate. In this case, the overhead coefficient γ_{gate} can be expressed as

$$\gamma_{\text{gate}} = \Pi_{i=1}^{N_{\text{tot}}}\gamma_i \tag{5.24}$$

where γ_i is the overhead coefficient for error suppression operation after i the gate, and N_{tot} is the total number of the gates. In Fig. 5.6, we show the result of probabilistic error cancellation technique.

Note that this technique needs the full knowledge of nature of noise associated

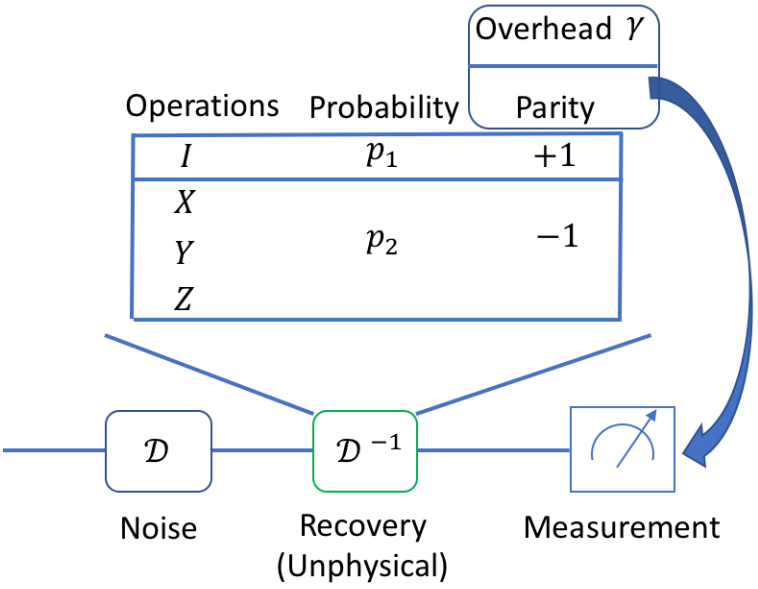


Figure 5.5: The schematic figure for quasi-probability method in the case of depolarising channel. We randomly generate I, and X, Y, Z with probabilities p_1 and p_2, and when X, Y, Z is generated, we multiply -1 to the measurement result and store it. We repeat the experiment and calculate the average value of the values obtained. Finally, we multiply the overhead coefficient γ to the average value, which approximates the ideal expectation value free from errors.

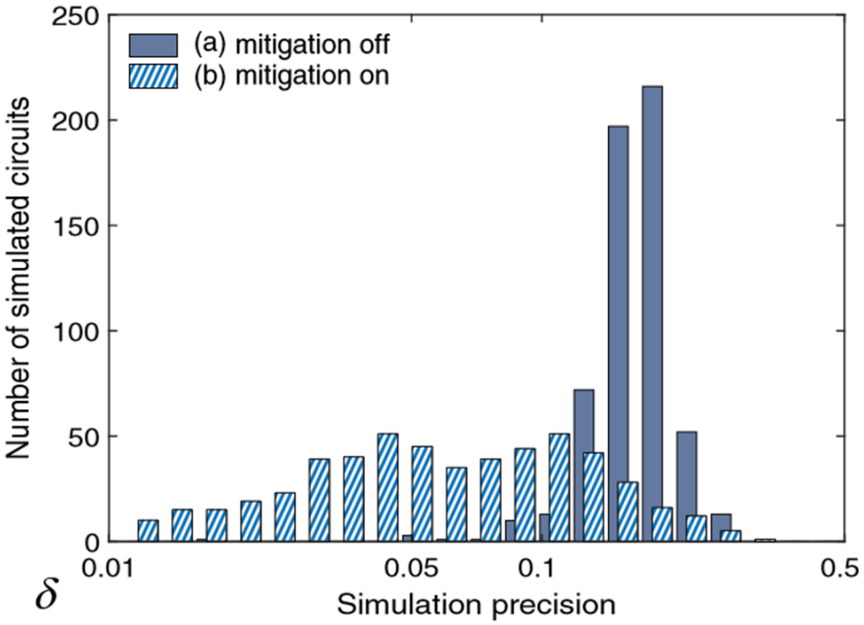


Figure 5.6: The performance of probabilistic error cancellation. The vertical axis is the number of simulated circuits and horizontal axis is the deviation from the ideal value. When mitigation is on, while the mean of deviation become closer to 0, the variance become broader due to cost. The simulated circuit is randomly generated ideal Clifford+T circuits on $n = 6$ qubits with the depth $d = 20$. This figure is reproduced from Ref [28].

with each gate, via process tomography. However, there are errors for preparing the initial state and measurements, therefore the conventional process tomography leads to imperfections in this error mitigation method. Also, there is a restriction that this scheme cannot be used for non-Markovian noises such as correlated errors. Furthermore, in the IBM group paper, only depolarising errors and amplitude damping errors were mentioned. Also, it is worth mentioning that they implicitly assume that error suppression operations are perfect, (for example, I, X, Y, Z operations for depolarising errors,) however, there is necessarily some noise to these operations. Considering these points, this method has to be generalised to other types of noise and noises to the error suppression operations should be taken into account so that this technique would be utilised in real experiments. This topic will be addressed in Chapter 8.

5.4 Other quantum error mitigation methods

5.4.1 Quantum subspace expansion

The quantum subspace expansion technique Ref. [80, 62] is designed to mitigate errors in VQE and also it enables the calculation of the excited energy eigenstates. This method can considerably suppress stochastic errors such as depolarising and dephasing errors. By using VQE, we can find an approximate ground state $|\psi_0\rangle$. Such a ground state may deviate from the true ground state $|\psi_{GS}\rangle$ due to errors in the whole process. For example, when $|\psi_{GS}\rangle = X_1 |\psi_0\rangle$, we can simply apply a X operation to $|\psi_0\rangle$ to get the correct ground state. However, in general, we do not know which error occurred to the state, so we should instead consider expansion of the state in a subspace $\{|\sigma_i\psi_0\rangle\}$, with Pauli matrix σ_i. Then, one can measure the matrix representation of the Hamiltonian in the subspace,

$$H_{ij} = \langle\psi_0| \sigma_i H \sigma_j |\psi_0\rangle . \tag{5.25}$$

As the subspace states are not orthogonal to each other, we should also measure the overlap matrix

$$S_{ij} = \langle\psi_0| \sigma_i\sigma_j |\psi_0\rangle . \tag{5.26}$$

By solving the generalised eigenvalue problem

$$HC = SCE, \tag{5.27}$$

with eigenvectors C and diagonal matrix of eigenvalues E, we can obtain the error mitigated spectra of the Hamiltonian, including the excited states. This would be true in general when the chosen subspace $\{|\sigma_i\psi_0\rangle\}$ can represent the full Hilbert space. However, this is unclear when there is a noise to the measurement of Pauli operators σ_i. Also, it requires exponentially large number of Pauli operators, and in practice we have to choose suitable set of Pauli operators. In Ref [62], this method was experimentally demonstrated and it successfully discovered the spectrum of H_2.

5.4.2 Stabiliser based method

The stabiliser based error mitigation technique uses the information related to the conserved quantities [81, 82, 83]. When we use the spin and particle number conserving ansatz, if the change of such preserving numbers is detected, we can say there is an error in the quantum circuit. We define the parity operators such that $\hat{P}_N |\psi\rangle = (-1)^N |\psi\rangle$, and $\hat{P}_{N_{\uparrow\downarrow}} |\psi\rangle = (-1)^{N_{\uparrow\downarrow}} |\psi\rangle$, where N is the number of particles and $N_{\uparrow\downarrow}$ are the number of up and down spins, respectively. If the eigenvalues of parity operators change, an error exists, and this detection procedure is similar to stabiliser measurement in quantum error correction. If the error is found, the state can be discarded, and the state can be prepared again, to reduce the error. We can implement the stabiliser measurement of $\hat{P}_N$ and $\hat{P}_{N_{\uparrow\downarrow}}$ by adding ancilla qubits to the register qubits, or by taking additional measurements and post-processing.

5.4.3 Individual error reduction

In Ref. [84], another type of error mitigation technique was proposed, which uses the expectation value obtained from the state with all the noise sources active, and expectation values from the state where the each error source is reduced, in the scenario that there are multiple noise sources. Suppose the error process after each quantum gate is described by Lindblad master equation,

$$\frac{d\rho}{dt} = L(\rho) = \sum_i L_i(\rho), \tag{5.28}$$

where L_i describes some noise process. Now, we compute

$$\tilde{\rho}(T) = \rho(T) - \sum_{j=1}^{m} \frac{1}{g_j}(\rho(T) - \rho_j(T)), \tag{5.29}$$

where $\rho(T)$ is the state after applying gates under noise process described by Eq. (5.28), T is the total evolution time and $\rho_j(T)$ is the state which evolves under the noise process where L_j is removed to the ratio g_j. (If $g_j = 1$, the noise process L_j is completely removed.) Now, the following relationship can be shown

$$\tilde{\rho}(T) = \rho_{\text{ideal}}(T) + O(\tau^2), \tag{5.30}$$

where $\rho_{\text{ideal}}(T)$ is the ideal output state in the quantum circuit, free from errors and τ is the evolution time for the noise process each after the gate. The first order error $O(\tau)$ is completely removed. Hence, when we would like to obtain the expectation value $\text{Tr}(\rho_{\text{ideal}}(T)\hat{A})$ for an observable $\hat{A}$, we can estimate it by computing the expectation value of $\hat{A}$ for $\rho(T)$ and $\rho_i(T)$, and linearly combining the results, following Eq. (5.29). As this method assumes a nearly perfect removal of noise with error correction, it is relatively unrealistic on current quantum devices compared with other error mitigation methods.

Chapter 6

Variational quantum algorithm for evaluating excited states

This is the first research chapter, describing work performed by the author and coworkers. The research has been published as Ref. [25]. The knowledge of excited states of molecular and materials systems allows us calculate photodissociation rates and absorption bands [85, 86]. They are useful for designing and predicting the behaviour of solar cells [87, 88]. Also, it is desirable to calculate the excited state energy and states for chemical reaction analysis, necessary for creating new drugs, and new methods of mass production of industrially useful materials [59, 89]. There are some proposals for evaluating excited state energies based on the VQE, such as the quantum subspace expansion method [80, 62] and witness-assisted variational eigenspectra solver (WAVES) [90]. However, each method has limitations. For the quantum subspace expansion method, many additional measurements are required, and WAVES needs a controlled e^{-iHt} evolution, implying deep quantum circuit is necessary.

In Ref. [25], we proposed a quantum algorithm which can evaluate the energy levels of a Hamiltonian sequentially, by using imaginary time evolution and the shallow swap test [91, 92] for evaluating the overlap of two input wave functions [25].

The steps are as follows: by using the recently proposed variational imaginary time evolution simulator [46], we evaluate the ground state. Then, feeding information of the ground state to the quantum circuit with the shallow swap test circuit, we can effectively penalise the ground state from the original Hamiltonian, and hence we can obtain the first excited state. By repeating this process, we can sequentially evaluate the eigenenergies and eigenstates of the Hamiltonian. The advantage of this method is that this algorithm only utilises shallow depth circuits. This method paves the way to deeper analysis of the structure of quantum Hamiltonians and systems.

The author of the thesis conceived the idea, and the coauthor Tyson Jones implemented the numerical simulations.

6.1 Evaluations of excited energy using imaginary time evolution

Here, we explain about our new variational algorithm for finding the energy spectrum. This method uses the imaginary time evolution for optimisation, and the shallow swap test for penalising the ground state to find the excited states. A similar method was proposed simultaneously by Higott *et al.*, [26], and they used the conventional VQE. We briefly review the variational imaginary time evolution algorithm below [46]. We introduced real and imaginary time evolution earlier in section 4.1.2. The ideal imaginary time evolution follows

$$\frac{d\left|\psi(\tau)\right\rangle}{d\tau} = -(H - \langle H(\tau)\rangle)\left|\psi(\tau)\right\rangle . \tag{6.1}$$

When we simulate this equation by using a variational ansatz

$$|\varphi(\theta_1(\tau), \theta_2(\tau), \theta_3(\tau)..., \theta_M(\tau))\rangle := |\varphi(\vec{\theta}(\tau))\rangle , \tag{6.2}$$

equations of the parameters are

$$\sum_j \mathcal{M}_{ij}\dot{\theta}_j = \mathcal{V}_i \tag{6.3}$$

where

$$\mathcal{M}_{ij} = \mathrm{Re}\left(\frac{\partial \langle\varphi(\vec{\theta}(\tau))|}{\partial\theta_j}\frac{\partial |\varphi(\vec{\theta}(\tau))\rangle}{\partial\theta_i}\right) \tag{6.4}$$

$$\mathcal{V}_i = \mathrm{Re}\left(\langle\varphi(\vec{\theta}(\tau))| H\frac{\partial |\varphi(\vec{\theta}(\tau))\rangle}{\partial\theta_i}\right). \tag{6.5}$$

Therefore, $\dot{\vec{\theta}} = \mathcal{M}^{-1}\mathcal{V}$. However, in some cases $\mathcal{M}$ is unstable to directly calculate the inverse matrix, such that the result becomes unsmooth. We can use Tikhonov regularisation [93] to minimise

$$\|\mathcal{V} - \mathcal{M}\dot{\vec{\theta}}\| + \lambda\|\dot{\vec{\theta}}\| \tag{6.6}$$

where there is a trade off between keeping $\dot{\vec{\theta}}$ small and smoothness, depending on the choice of λ. The norm used here is Euclidean norm.

Here, we discuss the number of measurements required for the conventional VQE and imaginary time. In order to obtain the gradient vector in VQE, we need to have $O(N_{\mathcal{V}}N_H N_p)$ measurements, where $N_{\mathcal{V}}$ is the number of measurements to achieve required precision for each element of the gradient vector, N_H is the number of terms of the Hamiltonian and N_p is the number of parameters in the ansatz. The gradient vector in VQE is the same as $\mathcal{V}$, therefore to populate $\mathcal{V}$, we also need to have $O(N_{\mathcal{V}}N_H N_p)$ measurements. For imaginary time evolution, we need to have $O(N_{\mathcal{M}}N_p^2)$ measurements to populate $\mathcal{M}$, where $N_{\mathcal{M}}$ is the number of measurements to achieve a required precision for each element of $\mathcal{M}$.

Now, suppose we can replace the Hamiltonian H with $H + a\,|\tilde{g}\rangle\langle\tilde{g}|$ in variational imaginary time evolution, where $|\tilde{g}\rangle$ is an estimate of the ground state previously

discovered with imaginary time evolution, a is a positive value sufficiently larger than the energy gap. This corresponds to penalising the state $|\tilde{g}\rangle$ energetically. Now the ground state for the new Hamiltonian is no longer $|\tilde{g}\rangle$ but $|\tilde{e}_1\rangle$ (the first excited state). When we hope to obtain $|\tilde{e}_2\rangle$, we replace the Hamiltonian with $H + a\,|\tilde{g}\rangle\langle\tilde{g}| + b_1\,|\tilde{e}_1\rangle\langle\tilde{e}_1|$. Generally, if we can prepare the effective hamiltonian $H + a\,|\tilde{g}\rangle\langle\tilde{g}| + \sum_{j=1}^{N} b_j\,|\tilde{e}_j\rangle\langle\tilde{e}_j|$, we can obtain $N+1$ th excited state $|\tilde{e}_{N+1}\rangle$.

To simulate the imaginary time evolution governed by the effective Hamiltonian $H + a\,|\tilde{g}\rangle\langle\tilde{g}|$, we should calculate the term $\mathcal{V}$ vector such as

$$\mathcal{V}_i = \mathrm{Re}\left(\frac{\partial \langle\varphi(\vec{\theta}(\tau))|}{\partial\theta_i} H\,|\varphi(\vec{\theta}(\tau))\rangle + a\frac{\partial \langle\varphi(\vec{\theta}(\tau))|}{\partial\theta_i}\,|\tilde{g}\rangle\langle\tilde{g}|\varphi(\vec{\theta}(\tau))\rangle \right) \tag{6.7}$$

Practically, we do not directly modify the Hamiltonian, as that requires full tomography of the state vector, which is exponentially costly. The second term can be rewritten as

$$a\mathrm{Re}\left(\frac{\partial \langle\varphi(\vec{\theta}(\tau))|}{\partial\theta_i}\,|\tilde{g}\rangle\langle\tilde{g}|\varphi(\vec{\theta}(\tau))\rangle \right) = \frac{a}{2}\frac{\partial}{\partial\theta_i}|\,\langle\varphi(\vec{\theta}(\tau))|\tilde{g}\rangle\,|^2 \tag{6.8}$$

$$\approx \frac{a}{2}\frac{|\,\langle\varphi(\theta_i+\delta\theta_i)|\tilde{g}\rangle\,|^2 - |\,\langle\varphi|\tilde{g}\rangle\,|^2}{\delta\theta_i}, \tag{6.9}$$

with small $\delta\theta_i$, and each term can be evaluated with the swap test, which evaluates the overlap of the two input wavefunctions.

In [91, 92], an algorithm using constant depth circuits and post processing for calculating the overlap of wavefunctions was introduced, (which was constructed with a machine learning approach). Our method for finding excited states can use this shallow swap test. The state overlap of the state ρ and σ can be evaluated as follows. The circuit is shown in Fig. 6.1. We apply CNOT gates, by setting qubits which express ρ as control qubits, and qubits for σ as target qubits. Then, Hadamard gates are applied transversally to the qubits for ρ. Measurement of an observable $\bigotimes_{n=1}^{N} C_n$ corresponds to the measurement of the overlap of two input

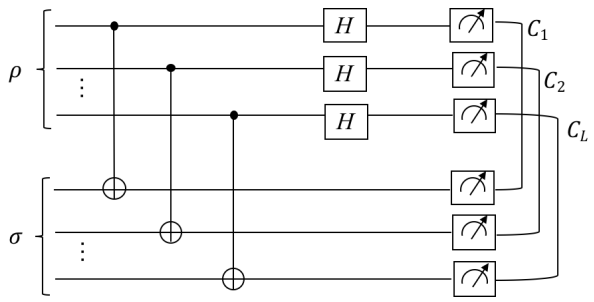


Figure 6.1: Schematic of swap test using a shallow depth circuit. This circuit evaluates the overlap of two input density matrix ρ and σ, shown in the figure.

states, where

$$C_n = |0\rangle\langle 0|^n_\rho \otimes |0\rangle\langle 0|^n_\sigma + |1\rangle\langle 1|^n_\rho \otimes |0\rangle\langle 0|^n_\sigma + |0\rangle\langle 0|^n_\rho \otimes |1\rangle\langle 1|^n_\sigma - |1\rangle\langle 1|^n_\rho \otimes |1\rangle\langle 1|^n_\sigma . \tag{6.10}$$

($|0\rangle^n_{\rho(\sigma)}$ denotes that the n th pair qubit which belongs to $\rho(\sigma)$, and N is the number of pairs.) This can be accomplished using following post processing. Firstly we consider the measurement of C_1. If the measurement result is $|0\rangle^1_\rho \otimes |0\rangle^1_\sigma$, $|0\rangle^1_\rho \otimes |1\rangle^1_\sigma$ and $|1\rangle^1_\rho \otimes |0\rangle^1_\sigma$, we assign measurement result $c_1 = 1$, and when the result is $|1\rangle^1_\rho \otimes |1\rangle^1_\sigma$, we assign $c_1 = -1$. Similarly, we measure C_n and obtain c_n. Then, the measurement result of the observable $\bigotimes_{n=1}^N C_n$ can be obtained as $\prod_{n=1}^N c_n$. By repeating this process and averaging over the results, we can evaluate the overlap function $\mathrm{Tr}(\rho\sigma)$.

The advantage of this scheme over the original swap test is that the total number of the gates is $2N$. ($2N$ is also the number of qubits in the quantum circuit. The conventional swap test circuit requires $23N_q - 21$ gates [94, 31], where N_q is the number of qubits including the ancilla qubit, as we will mention in the later section.) The CNOT and Hadamard gates can be applied in parallel, which means the quantum algorithm can be implemented with the constant depth circuit.

Another method to compute the overlap is proposed in Ref. [26]. Suppose $|\tilde{g}\rangle = U(\vec{\theta}_g)\,|\bar{0}\rangle$, where $U(\vec{\theta})$ is the ansatz unitary. Now, the overlap term $|\,\langle\tilde{g}|\varphi(\vec{\theta})\rangle\,|^2 =$

$|\langle\bar{0}|U^{\dagger}(\vec{\theta_g})U(\vec{\theta})|\bar{0}\rangle|^2$ can be computed as, by applying $U(\vec{\theta})$ and inverse operation of $U(\vec{\theta_g})$ to $|\bar{0}\rangle$, and measure the probability to obtain $|\bar{0}\rangle$. This method does not require SWAP test and halves the number of qubits, however, requires double the depth of the quantum circuit.

6.2 Numerical simulation results

Here, we explain about the numerical simulation results, which were implemented by Tyson Jones. Now, we applied our new methods to two systems: 3SAT problem Hamiltonian and LiH Hamiltonian, with two types of ansätze; the low depth circuit ansatz [95] and the compact ansatz [46]. The low depth circuit ansatz is a chemically motivated ansatz. The compact ansatz is firstly introduced in Ref. [46], and successfully discovered the ground state of LiH. The number of parameters in both two ansätze scale linearly.

6.2.1 Benchmark by evaluating the spectrum of 3SAT Hamiltonian

The Boolean satisfiability problem is finding the set of variables which are either True or False and can be encoded into 1 and 0 respectively, satisfying the condition that all clauses given in propositional formula are true [96]. Each variable in clauses is with or without negation (turning True into False, or vice versa), and if there is at least one true valuable in each clause, the clause becomes true. For example, suppose the problem to find the set of variables as follows:

$$(x_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_2) = \text{True}. \tag{6.11}$$

where $\bar{x}$ denotes the negation of the variable x, $\vee$ and $\wedge$ are OR and AND, meaning $y_1 \vee y_2$ outputs 1 when either y_1 or y_2 is 1, otherwise 0, and $y_1 \wedge y_2 = y_1 y_2$. In this

problem, $x_1 = 1$ (True) and $x_2 = 0$ (False) is the solution.

Now, any clause (e.g. $x_1 \vee x_2 \vee \bar{x}_3$) is only violated by one specific set of values (in this example, $(x_1, x_2, x_3) = (0, 0, 1) = (F, F, T)$). Suppose we introduce numbers n_1, n_2, n_3 with $n_i = +1$ if Boolean value x_i appears in direct form (i.e. as x_i), and $n_i = -1$ if x_i appears in negated form (i.e. as $\bar{x}_i$). Now we use qubit mapping True $\rightarrow |1\rangle$, False $\rightarrow |0\rangle$, and notice

$$H = \frac{1}{2^3}(1 + n_1 Z_1)(1 + n_2 Z_2)(1 + n_3 Z_3) \tag{6.12}$$

has zero energy for all z-basis states except the state corresponding to the violated state.

When we have several clauses, we can map each clause in the form of Eq. (6.12), and the Hamiltonian should be linear combination of each Hamiltonian.

Now, we applied our new algorithms for finding the spectrum of the 3SAT Hamiltonian. It is worth noting that for 3SAT problems, the spectrum of the Hamiltonian is in general of no interest, and evaluating the ground state is our goal. Therefore, finding the spectrum and the excited states is not so necessary for 3SAT problems, and this is simply a benchmark of this algorithm by using 3SAT Hamiltonian. 3SAT Hamiltonian has an equally spaced spectrum and each energy has a large number of degeneracies.

We randomly generated 3SAT problems with 16 to 18 variables and approximately 4.27 clauses per variable which maximises the chance that there will be a unique satisfying solution. We select two such cases and used these to demonstrate our quantum algorithm. We show the evolution of the expectation value of the energy $\langle \varphi(\vec{\theta}(\tau))| H |\varphi(\vec{\theta}(\tau))\rangle$ in Fig. 6.2, and verified our algorithm successfully discovered the spectrum. Note the order that the spectrum is found is not always in order. For example, for qubit number 16, the right figure of Fig. 6.2, the first excited state is discovered before the ground state.

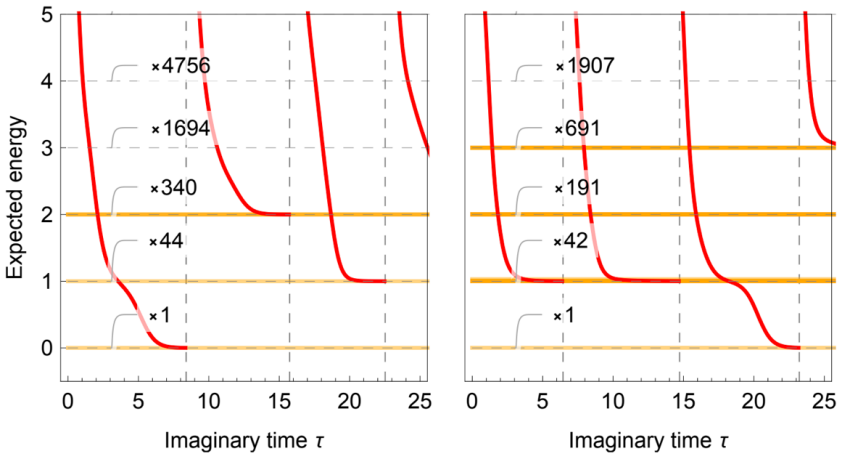


Figure 6.2: The expectation value of the energy when we implement imaginary time evolution with energy penalisation for 3SAT problem. The left figure is the Hamiltonian with 18 qubits and the right figure with 16 qubits. It is not necessarily that the energy spectrum is found in order. The evolution of the expected energy is discontinuous at some points, and this is because the Hamiltonian is updated to include the penalisation for already discovered states. Labels in the figures indicate the number of degenerate states. The figures were produced by Tyson Jones.

6.2.2 Search of the spectrum of LiH

Now we move on to the simulation of LiH molecule. To obtain the 2$^{\text{nd}}$ quantised Hamiltonian, we chose STO-3G basis, where three gaussian functions are used to approximate Slater type orbital for expressing the distribution of electrons [58, 59]. In this case, LiH has 12 spin orbitals as $2 \times (\{1S\}^{\text{H}} + \{1S, 2S, 2P_x, 2P_y, 2P_z\}^{\text{Li}})$. These 12 orbitals can be mapped to 10 qubits by using Bravi-Kitaev encoding [61] and restricting to non-ionic states with four electrons. For some numerical results, we further reduced the number of orbitals and the number of qubits is reduced to 6. This reduction would give a nonphysical spectrum, although it is still useful for testing our algorithm.

By examining the natural molecules orbital basis, we can reduce the number of orbitals. This basis diagonalises the single-particle reduced density matrix (1-RDM). Now, we remove the diagonal component whose occupation is close to 1 or 0, as these orbitals are just fulfilled or empty. Then we remove the fermionic operators

corresponding to the removed orbitals [64, 46]. Now, we map the 2^{nd} quantised Hamiltonian into qubit form by using some encoding method. In our 6 qubit and 10 qubit simulation, we have removed two qubits by using the conservation of spin number and electron number. We used Open Fermion, an open source package for quantum computational chemistry [97], to carry out all these processes.

As a result of the transformation mentioned above, the Hamiltonian for LiH molecule can be written as the linear combination of Pauli operators with coefficients

$$H = \sum_{j}^{M} h_j \prod_{i} \sigma_i^j, \tag{6.13}$$

where σ_i^j is one of I, X, Y, Z Pauli operators. M scales as $O(N_o^4)$, where N_o is the number of orbitals. The Hamiltonian includes terms which are associated with all the qubits, such as $XYYXYY$ for 6 qubit LiH Hamiltonian.

Our algorithm also worked for 6 qubit LiH Hamiltonian. We show the result in Fig. 6.3. As a comparison, we also show the result obtained from the case that the optimisation method is conventional gradient descent method, which implies imaginary time evolution is superior method for optimisation. In the case of gradient descent, noneigenstates are found, and subsequently the Hamiltonian is modified incorrectly. Also, we show the energy spectrum of the converged states in Fig. 6.4. It is clear that the state is trapped in noneigenstates in the case of gradient descent.

Next, we simulated the dependence of the spectrum of LiH on the interatomic distance by using 10-qubit Hamiltonian. We used 70 and 145 parameters for the compact and the low depth ansätze, respectively. The compact ansatz shows good agreement with the exact spectrum, even though the number of parameters is much fewer than the Hilbert space size $2^{10} = 1024$. The reason the accuracy for the ground state is not so good might be insufficient ability to generate the ground state. When we used low depth ansatz with 145 parameters, the performance has been improved in accuracy and discovery of degenerate states. In both cases, around interatomic

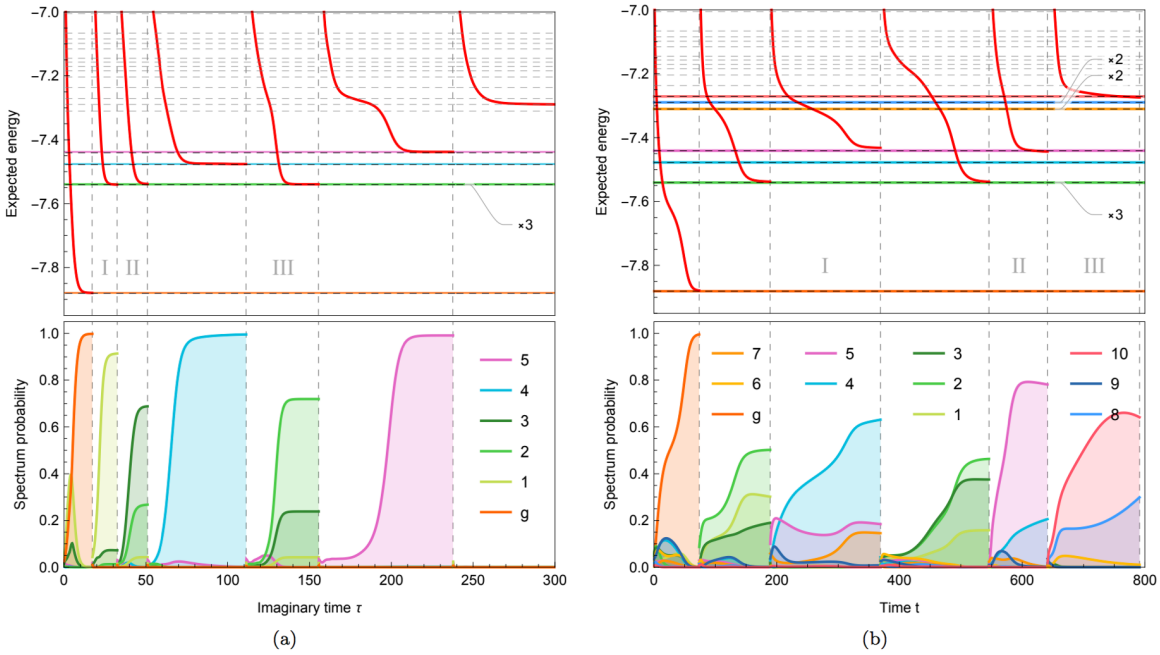


Figure 6.3: The expectation value of the energy when we implement the imaginary time evolution algorithm with energy penalisation for the 6-qubit LiH Hamiltonian. We used the low depth ansatz with 56 parameters. The top figures show the expectation value of the energy. Again, the evolution of the expected energy is discontinuous at some points, and this is because the Hamiltonian is updated to have the penalisation for already discovered states. Horizontal dashed and coloured lines correspond to exact eigenvalues of the Hamiltonian. The bottom plots are population of the eigenstate included in $|\varphi(\vec{\theta}(\tau))\rangle$. (a) Imaginary time. we verified that the trial state converges to the eigenstate (other regions than I, II, III), or superposition of the degenerated eigenstates (Region I, II, III). (b) Gradient descent. In regions I, II and III, the state converge to the noneigen state. The figures were produced by Tyson Jones.

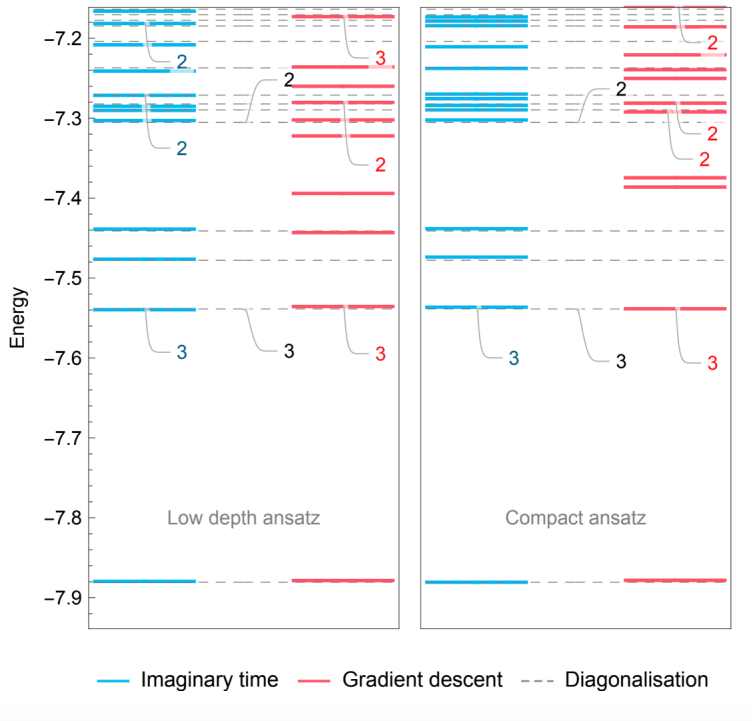


Figure 6.4: The 6-qubit LiH Hamiltonian spectrum at the converged point. We simulated this by using low depth ansatz and compact ansatz, with 56 and 42 parameters, respectively. We see that imaginary time performs significantly better whereas gradient descent generates an unreliable spectrum. The figures were produced by Tyson Jones.

distance $l \approx 2.5$Å, the accuracy decreases, which also has been observed in recent VQE experiments. The intuition is, at around $l \approx 2.5$Å, the system is undergoing bond breaking, and the static correlation and dynamic correlation are competing, leading to a highly entangled state, which is difficult to be produced by the ansatz with insufficient power.

6.3 The effect of shot noise

The shot noise affects the solution of Eq. (6.3). Now, the solution $\dot{\vec{\theta}}$ in Eq. (6.3) can be written as

$$\dot{\vec{\theta}} = (\mathcal{M}_0 + \delta\mathcal{M})^{-1}(\mathcal{V}_0 + \delta\mathcal{V}), \tag{6.14}$$

where $\mathcal{M}_0$ and $\mathcal{V}_0$ are noise free $\mathcal{M}$ matrix and $\mathcal{V}$ vector, $\delta\mathcal{M}$ and $\delta\mathcal{V}$ originate from errors, e.g., physical errors such as dephasing and shot noise. Here we focus on the effect of shot noise. Assuming the effect of shot noise is sufficiently small, i.e., each matrix component pf $\delta\mathcal{M}$ is small enough, and the singular value of it is much smaller than that of $\mathcal{M}_0$, so that $\|\mathcal{M}_0^{-1}\delta\mathcal{M}\| \leq \|\mathcal{M}_0^{-1}\|\|\delta\mathcal{M}\| \approx 0$ (the norm used here is the Frobenius norm). Now,

$$\begin{aligned}
\dot{\vec{\theta}} &= \big(\mathcal{M}_0(I + \mathcal{M}_0^{-1}\delta\mathcal{M})\big)^{-1}(\mathcal{V}_0 + \delta\mathcal{V}) \\
&= (I + \mathcal{M}_0^{-1}\delta\mathcal{M})_0^{-1}\mathcal{M}^{-1}(\mathcal{V}_0 + \delta\mathcal{V}) \\
&\approx (I - \mathcal{M}_0^{-1}\delta\mathcal{M})(\mathcal{M}_0^{-1}\mathcal{V}_0 + \mathcal{M}_0^{-1}\delta\mathcal{V}) \\
&\approx \dot{\vec{\theta}}_0 + \mathcal{M}_0^{-1}\delta\mathcal{V} - \mathcal{M}_0^{-1}\delta\mathcal{M}\dot{\vec{\theta}},
\end{aligned} \tag{6.15}$$

where $\dot{\vec{\theta}}_0$ is the error free ideal solution of Eq. (6.3), and we used $(I - \mathcal{M}_0^{-1}\delta\mathcal{M})(I + \mathcal{M}_0^{-1}\delta\mathcal{M}) = I - (\mathcal{M}_0^{-1}\delta\mathcal{M})^2 \approx I$, thus $(I + \mathcal{M}_0^{-1}\delta\mathcal{M})^{-1} \approx I - \mathcal{M}_0^{-1}\delta\mathcal{M}$. The second and third term indicate the effect of shot noise. Then, the effect of the shot

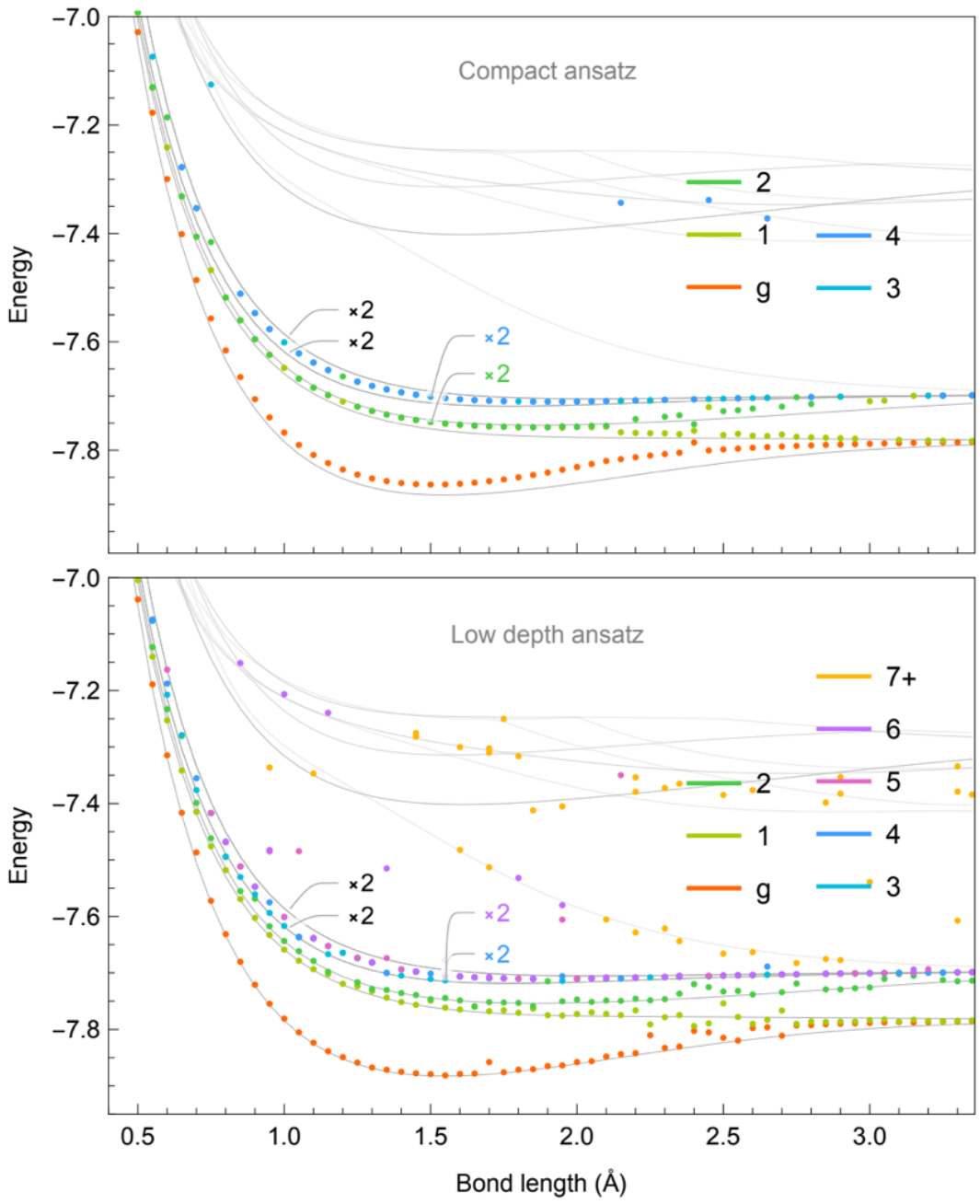


Figure 6.5: The 10-qubit LiH Hamiltonian spectrum versus interatomic distance. We simulated this by using low depth ansatz and compact ansatz, with 70 and 145 parameters for 10000 and 40000 iterations at each bond length, respectively. The line labels indicate the degeneracy for both the true and discovered states. The figures were produced by Tyson Jones.

noise can be written as

$$\begin{aligned} \|\delta\dot{\vec{\theta}}\| &= \|\dot{\vec{\theta}} - \dot{\vec{\theta}}_0\| && (6.16)\\ &= \|\mathcal{M}_0^{-1}\delta\mathcal{V} - \mathcal{M}_0^{-1}\delta\mathcal{M}\dot{\vec{\theta}}\| && (6.17)\\ &\leq \|\mathcal{M}_0^{-1}\|\|\delta\mathcal{V}\| + \|\mathcal{M}_0^{-1}\|^2\|\mathcal{V}_0\|\|\delta\mathcal{M}\|. && (6.18) \end{aligned}$$

Now, we introduce the number of measurements N_{m}, and by using the coefficients for the simulated Hamiltonian $H = \sum_j f_j\sigma_j$ and ones introduced in Eq. (4.21),

$$\begin{aligned} \|\delta\mathcal{M}\| &\approx \frac{2\left[\sum_{k,j}\left(\sum_{i,q}|g_{k,i}^{*}g_{j,q}|\right)^2\right]^{\frac{1}{2}}}{\sqrt{N_{\mathrm{m}}}},\\ \|\delta\mathcal{V}\| &\approx \frac{2\left[\sum_{k}\left(\sum_{i,j}|g_{k,i}^{*}f_{j}|\right)^2\right]^{\frac{1}{2}}}{\sqrt{N_{\mathrm{m}}}} + \frac{aN_{\mathrm{p}}}{2\delta\theta\sqrt{N_{\mathrm{m}}}}. \end{aligned} \tag{6.19}$$

Note that the second term of $\|\delta\mathcal{V}\|$ comes from the penalising previously discovered states by using the swap test, and N_{p} is the number of penalised states. Therefore, $\|\dot{\vec{\theta}}\|$ can be written as

$$\|\delta\dot{\vec{\theta}}\| \leq \frac{\Delta}{\sqrt{N_{\mathrm{m}}}} \tag{6.20}$$

where

$$\Delta = 2\left(\left[\sum_{k}\left(\sum_{i,j}|g_{k,i}^{*}f_{j}|\right)^2\right]^{\frac{1}{2}} + \frac{aN_{\mathrm{p}}}{4\delta\theta}\right)\|\mathcal{M}_0^{-1}\| + 2\left[\sum_{k,j}\left(\sum_{i,q}|g_{k,i}^{*}g_{j,q}|\right)^2\right]^{\frac{1}{2}}\|\mathcal{M}_0^{-1}\|^2\|\mathcal{V}_0\|. \tag{6.21}$$

In Ref. [24], it is shown the implementation error due to shot noise to update

parameter from $\dot{\vec{\theta}}(t)$ to $\dot{\vec{\theta}}(t+\delta t)$ is

$$D(|\psi^{(0)}(\vec{\theta}_0(t+\delta t))\rangle, |\psi(\vec{\theta}(t+\delta t))\rangle) = \sqrt{\delta\dot{\vec{\theta}}^T A \delta\dot{\vec{\theta}}\delta t^2 + O(\delta t^3)}, \tag{6.22}$$

$$A_{i,j} = \frac{\langle\psi(\vec{\theta}(t))|}{\partial\theta_j}\frac{|\psi(\vec{\theta}(t))\rangle}{\partial\theta_i} - \frac{\langle\psi(\vec{\theta}(t))|}{\partial\theta_j}|\psi(\vec{\theta}(t))\rangle\langle\psi(\vec{\theta}(t))|\frac{|\psi(\vec{\theta}(t))\rangle}{\partial\theta_i} \tag{6.23}$$

where $|\psi^{(0)}(\vec{\theta}_0(t+\delta t))\rangle$ denote corresponds to the noise free parameter update $\vec{\theta}_0(t) = \vec{\theta}(t) + \dot{\vec{\theta}}_0\delta t$, and $D(|\phi_1\rangle, |\phi_2\rangle)$ is the trace distance between two states. Therefore, the total implementation error due to the shot noise from time $t = 0$ to $t = T$ is

$$\begin{aligned} D_I &\lessapprox \sqrt{\|A\|_{\max}}\|\delta\dot{\vec{\theta}}\|_{\max}T \\ &= \sqrt{\|A\|_{\max}}\frac{\Delta_{\max}T}{\sqrt{N_{\mathrm{m}}}} \end{aligned} \tag{6.24}$$

Thus, in order to limit the effect of the shot noise to ϵ_s as

$$\epsilon_s \leq \sqrt{\|A\|_{\max}}\frac{\Delta_{\max}T}{\sqrt{N_{\mathrm{m}}}}, \tag{6.25}$$

we need to have $N_{\mathrm{m}} = \|A\|_{\max}\Delta_{\max}^2T^2/\epsilon_s^2$.

6.4 Discussion

We proposed a quantum algorithm for evaluating excited states with near term quantum devices, and it works well for 3SAT and LiH Hamiltonians. Our method is also compatible with the error mitigation techniques recently proposed, as it only uses shallow depth quantum circuit. Although we used our method to find excited states, we can also use this to escape from local minima where the trial state are trapped. We checked that the trial state is rarely trapped in noneigenstates of the Hamiltonian when we used imaginary time evolution as an optimisation method, therefore our method can be used to escape from local minima. In contrast, when we used gradient descent for optimisation, local minima tend to be noneigenstates of

the Hamiltonian, therefore our penalisation method with SWAP test is not suitable for avoiding local minima in the case of gradient descent.

There are other potential use for our method. It can be used for penalising the unwanted subspace, which breaks the symmetries for instance, when we know the parameters for such subspace a priori. We can even modify the Hamiltonian with our method for real time evolution. For example, we can eliminate or create energy degeneracies.

Also, it is worth mentioning that after the optimisation to find a certain excited state, the quantum device directly generates that excited state wavefunction, which is not necessarily true for other methods for finding spectrum, which will be mentioned later.

There are some points which should be still sought, such as propagation of physical errors, how the error of the ground state affect the search of excited state. Physical errors such as dephasing also impair the precision of the simulation. The effect can also be included in Eq. (6.15), and the similar argument to shot noise may be able to be discussed. It is important to mention that, we can use the error mitigation technique, at the cost of increasing the number of samples needed.

Also, the errors to the parameters to the penalised already found states to discover new excited states should be discussed. We found that the state trapped in the local minimum is likely to be close to the eigenstates of the Hamiltonian, therefore the effect of error is expected to be relatively low when we use the imaginary time simulation as the optimisation method, compared with other optimisation methods.

The resource estimation and the investigation of the performance in the real experimental setups to include the effect of physical noise, the error of previously found states to be penalised, and the error mitigation is left to the future work.

6.5 Other variational algorithms for finding molecule spectra

We conclude this chapter with a brief look at other methods for finding spectra, and the strength/weakness of those methods. There are other variational quantum algorithms for finding spectra, such as witness-assisted variational eigenspectra solver (WAVES) [90], the folded spectrum method [22], the quantum subspace expansion method (QSE) [80, 62], and contraction VQE method [98, 99].

WAVES is a quantum algorithm for finding spectra by combining VQE and phase estimation [90]. Although WAVES has been implemented experimentally, it needs to realise the controlled time evolution operator e^{-iHt}, therefore it is unsuitable for near-term quantum devices.

The folded spectrum method minimises $(H - \alpha I)^2$ instead of H, and if α is close to the eigenvalue of H, we can obtain approximation of excited states [22]. To evaluate H^2 is very costly: as $O(N^4)$ measurements are required for H in chemistry problems, $O(N^8)$ measurements are necessary for measuring H^2, where N is the number of orbitals. Also, it is hard to know suitable α a priori. Therefore, this method is also not ideal.

The quantum subspace expansion represents Hamiltonian in a subspace $a_i^\dagger a_j \left|\psi_0\right\rangle$ for all possible i, j, where $a_i^\dagger$ and a_j are fermionic operators and $\left|\psi_0\right\rangle$ is an approximated ground state [80, 62]. The excited states are found by solving

$$H_{QSE} C = S^{QSE} C E, \tag{6.26}$$

where H^{QSE} is the Hamiltonian projected to the subspace and expressed as

$$H^{QSE}_{i,j,k,l} = \left\langle\psi_0\right| a_i a_j^\dagger H a_k^\dagger a_l \left|\psi_0\right\rangle . \tag{6.27}$$

Also, as the subspace states are not orthogonal to each other, the overlap matrix

$$S^{QSE}_{i,j,k,l} = \langle\psi_0|\, a_i a_j^\dagger a_k^\dagger a_l \,|\psi_0\rangle \tag{6.28}$$

should be calculated. QSE requires additional measurements than conventional VQE for evaluating H^{QSE} and S^{QSE}. Also, in this method the quantum device does not directly produce the wavefunction of excited states.

Contraction VQE method rely on minimisation of the cost function for several input states

$$\mathcal{C}(\vec{\theta}) = \sum_{j=0}^{k} \langle\psi_j|\, U^\dagger(\vec{\theta}) H U(\vec{\theta}) \,|\psi_j\rangle \tag{6.29}$$

where $\{|\psi_i\rangle_{i=0}^k\}$ ($\langle\psi_j|\psi_i\rangle = \delta_{i,j}$. Note that as unitary operator conserves the orthonormality [98, 99]. When the optimisation is converged, in the ideal case, $\{U(\vec{\theta_0})\,|\psi_j\rangle\}\forall j$ spans the subspace up to the k th excited state. It is important to obtain the excited state from this subspace. In subspace-search variational quantum eigensolver (SSVQE), an optimisation method similar to the conventional VQE is used [98]. On the other hand, in multistate contracted variant of VQE (MC-VQE) [99], classical diagonalisation for Hamiltonian matrix obtained in the subspace is used, which is similar to QSE. Contraction VQE enables us to obtain the spectrum at the same time. Therefore, all the states are evaluated to a similar accuracy. However, as the cost function involves multiple states, the landscape of the energy becomes complex, rendering the optimisation hard.

Considering these points, although our method for finding spectra needs to have double the depth or the number of qubits, it has advantages such as, the compatibility with near-term quantum devices, the moderate increase of the terms to be measured, the direct generation of the wavefunction from the quantum device, and the simpler landscape of the energy to be optimised. Therefore, we may be

able to conclude our method is the most suitable for near-term quantum devices.

Chapter 7

Variational quantum algorithm for general processes

7.1 Introduction

In previous chapters, we saw variational quantum algorithms can be utilised for efficiently finding energy spectra [19, 100, 101, 66, 102, 22, 27, 26, 90] and simulating real time Schrödinger evolution [24, 103] of classically intractable many-body systems. Although quantum circuits are unitary operations, the variational algorithm is not limited to energy minimisation and unitary processes and it can be used to simulate dissipative imaginary time evolution that cannot be straightforwardly mapped to unitary gates [46, 75].

In Ref. [27], we study the capability of variational quantum algorithms and show that they are not limited to these problems. First, we propose a variational quantum algorithm for simulating the generalised time evolution defined in Eq. (7.1). Our algorithm can be considered as a unified framework for simulating general time evolutions, which incorporates real and imaginary time evolutions as special cases [24, 46, 75]. We can even simulate non-Hermitian quantum mechanics [104, 105, 106] that describes nonequilibrium processes [107], parity-time

symmetric Hamiltonians [108, 109], open quantum systems [110], etc.

Next, we consider problems of linear systems of equations and matrix-vector multiplications, which are vital in many fields including machine learning and optimisation [111, 112]. Although various algorithms have been developed for linear systems of equations with universal quantum computers [113, 114, 115, 116, 117, 118, 119, 120], which have profound applications in quantum machine learning [121, 122, 123, 124], they generally necessitates a long depth circuit that uses a fault tolerant error correction.

Now, we introduce variational quantum algorithms for these two tasks. For general sparse matrices, we show that these two tasks can be interpreted as a generalised time evolution, which can be variationally simulated. For special matrices that are products of small matrices that only involve a constant number of qubits, the solutions can be even more easily found only with variational real and imaginary time evolution.

Finally, we combine the variational algorithms for non-Hermitian evolution and matrix multiplication to simulate the evolution of open quantum systems [33, 125, 34]. Under the description of the stochastic Schrödinger equation, the evolution of open quantum systems can be regarded as an ensemble of wave functions that undergo a continuous measurement induced from the environment [126, 34]. The evolution of each wave function has two processes that can be both simulated with variational algorithms: continuous evolution under the generalised time evolution with the system Hamiltonian combined with the damping effect due to continuous measurement, and discontinuous jumps according to the measurement results. The continuous process can be described by the generalised time evolution, and the jump process is a matrix-vector multiplication process.

Simulating the evolution of general open quantum systems is of great importance for understanding any quantum system that interacts with an environment. Existing quantum algorithms [127, 128, 129, 130, 131, 132] for simulating open quantum

systems generally require deep quantum circuits. As our algorithm is compatible with shallow circuits, it can be implemented with near-term quantum devices. The research described in this chapter has appeared online [27].

Analytical results were derived by the author, with input from Xiao Yuan. Numerical results were obtained by the author, but Xiao Yuan also contributed to it.

7.2 Generalised time evolution

The simulation of general processes rather than Schrödinger equation and Wick rotated Schrödinger equation for imaginary time evolution have been enabled in Ref. [27]. We first introduce variational quantum simulation of generalised time evolution,

$$\frac{d}{dt}\left|v(t)\right\rangle = \left|dv(t)\right\rangle, \tag{7.1}$$

with $\left|dv(t)\right\rangle = \sum_j A_j(t)\left|v_j'(t)\right\rangle$. Here, $A_j(t)$ is a time dependent general sparse (non-Hermitian) operator, $\left|v(t)\right\rangle$ is the system state, and each of $\left|v_j'(t)\right\rangle$ can be either $\left|v(t)\right\rangle$ or any known efficiently preparable state on a quantum circuit. The states $\left|v(t)\right\rangle$ and $\left|v_j'(t)\right\rangle$ can be (un)normalised states $\left|v(t)\right\rangle = \alpha(t)\left|\psi(t)\right\rangle$, $\left|v_j'(t)\right\rangle = \alpha_j'(t)\left|\psi_j'(t)\right\rangle$ with normalisation factors $\alpha(t)$ and $\alpha_j'(t)$, respectively. In practice, we assume that $A_j(t)$ can be decomposed as a linear combination of Pauli operators $A_j(t) = \sum_i \lambda_{i,j}(t)\sigma_i$ with complex coefficients λ_i and a polynomial (to the system size) number of tensor products of Pauli matrices σ_i.

In variational quantum simulation, we parametrise the quantum state as $\left|v(\vec{\theta}(t))\right\rangle = \alpha(\vec{\theta}_0(t))\left|\varphi(\vec{\theta}_1(t))\right\rangle$ with $\vec{\theta} := (\vec{\theta}_0, \vec{\theta}_1)$. Then we can project the original evolution to the evolution of the parameters via McLachlan's principle [72],

$$\delta\| \left(\partial/\partial t \left|v(\vec{\theta}(t))\right\rangle - \sum_j A_j(t)\left|v_j'(t)\right\rangle \right)\| = 0, \tag{7.2}$$

where $\| |\psi\rangle \| = \langle\psi|\psi\rangle$.

By parametrising $|v(t)\rangle$ and $|v'_j(t)\rangle$ as $|v(\vec{\theta}(t))\rangle$ and $|v'(\vec{\theta}'_j(t))\rangle$, with McLachlan's principle, we have

$$\delta \left\| \sum_i \frac{\partial |v(\vec{\theta}(t))\rangle}{\partial \theta_i} \dot{\theta}_i - \sum_j A_j(t) |v'(\vec{\theta}'_j(t))\rangle \right\| = 0. \tag{7.3}$$

This is equivalent to

$$\begin{aligned} &\frac{\partial}{\partial \dot{\theta}_k} \left\| \sum_i \frac{\partial |v(\vec{\theta}(t))\rangle}{\partial \theta_i} \dot{\theta}_i - \sum_j A_j(t) |v'(\vec{\theta}'_j(t))\rangle \right\| \\ &= \frac{\partial}{\partial \dot{\theta}_k} \left(\sum_i \frac{\partial \langle v(\vec{\theta}(t))|}{\partial \theta_i} \dot{\theta}_i - \sum_j \langle v'(\vec{\theta}'_j(t))| A_j^\dagger(t) \right) \\ &\left(\sum_l \frac{\partial |v(\vec{\theta}(t))\rangle}{\partial \theta_l} \dot{\theta}_l - \sum_j A_j(t) |v'(\vec{\theta}'_j(t))\rangle \right) = 0 \ \forall k. \end{aligned} \tag{7.4}$$

Hence, we have

$$\begin{aligned} &\sum_j \left(\frac{\partial \langle v(\vec{\theta}(t))|}{\partial \theta_k} \frac{\partial |v(\vec{\theta}(t))\rangle}{\partial \theta_j} + \frac{\partial \langle v(\vec{\theta}(t))|}{\partial \theta_j} \frac{\partial |v(\vec{\theta}(t))\rangle}{\partial \theta_k} \right) \dot{\theta}_j \\ &= \sum_j \frac{\partial \langle v(\vec{\theta}(t))|}{\partial \theta_k} A_j |v'_j(\vec{\theta}'_j(t))\rangle + h.c, \end{aligned} \tag{7.5}$$

which leads to

$$\sum_j \tilde{M}_{k,j} \dot{\theta}_j = \tilde{V}_k, \tag{7.6}$$

By substituting $|v(\vec{\theta}(t))\rangle = \alpha(\vec{\theta_0}(t)) |\varphi(\vec{\theta_1}(t))\rangle$ and $|v'(\vec{\theta}'_j(t))\rangle = \alpha'(\vec{\theta}'_{0j}(t)) |\varphi(\vec{\theta}'_{1j}(t))\rangle$,

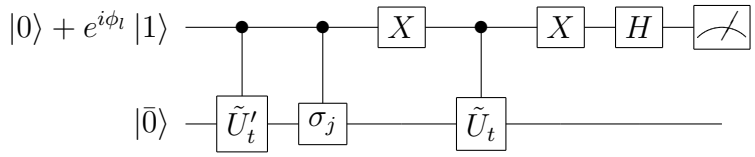


Figure 7.1: The quantum circuit for evaluating $\tilde{V}_k$.

we have

$$\begin{aligned}
\tilde{M}_{k,j} &= \mathrm{Re}\left(|\alpha(\vec{\theta}_0(t))|^2 \frac{\partial \langle\varphi(\vec{\theta}_1(t))|}{\partial\theta_k} \frac{\partial |\varphi(\vec{\theta}_1(t))\rangle}{\partial\theta_j}\right) \\
&+ \mathrm{Re}\left(\frac{\partial\alpha^*(\vec{\theta}_0(t))}{\partial\theta_k}\alpha(\vec{\theta}_0(t)) \langle\varphi(\vec{\theta}_1(t))| \frac{\partial |\varphi(\vec{\theta}_1(t))\rangle}{\partial\theta_j}\right) \\
&+ \mathrm{Re}\left(\frac{\partial\alpha^*(\vec{\theta}_0(t))}{\partial\theta_j}\alpha(\vec{\theta}_0(t)) \langle\varphi(\vec{\theta}_1(t))| \frac{\partial |\varphi(\vec{\theta}_1(t))\rangle}{\partial\theta_k}\right) \\
&+ \mathrm{Re}\left(\frac{\partial\alpha(\vec{\theta}_0(t))}{\partial\theta_k}\frac{\partial\alpha^*(\vec{\theta}_0(t))}{\partial\theta_j}\right), \\
\tilde{V}_k &= \mathrm{Re}\left(\frac{\partial\alpha^*(\vec{\theta}_0(t))}{\partial\theta_k}\alpha'(\vec{\theta}'_{0j}(t)) \langle\varphi(\vec{\theta}_1(t))| A_j(t) |\varphi'_j(\vec{\theta}'_{1j}(t))\rangle\right). \\
&+ \sum_j \mathrm{Re}\left(\alpha^*(\vec{\theta}_0(t))\alpha'(\vec{\theta}'_{0j}(t))\frac{\partial \langle\varphi(\vec{\theta}_1(t))|}{\partial\theta_k} A_j(t) |\varphi'_j(\vec{\theta}'_{1j}(t))\rangle\right).
\end{aligned} \tag{7.7}$$

Each element of $\tilde{M}$ can be efficiently computed with the quantum circuit in Fig. 4.2. $\tilde{V}$ can be computed as follows. We denote $\tilde{U}_t$ and $\tilde{U}'_t$ to be the unitary circuit to prepare $|\varphi(\vec{\theta}_1(t))\rangle = \tilde{U}_t|\bar{0}\rangle$ and $|\varphi'_j(\vec{\theta}'_{1j}(t))\rangle = \tilde{U}'_t|\bar{0}\rangle$. Replacing $A_j(t) = \sum_i \lambda^i_j(t)\sigma_i$, the first term of each $\tilde{V}_k$ can be written in a form of $\sum_l a_j \mathrm{Re}\left(e^{i\phi_j}\langle\bar{0}|\tilde{U}^\dagger_t\sigma_j\tilde{U}'_t|\bar{0}\rangle\right)$ with $a_j, \phi_j \in \mathbb{R}$ determined by α, α', and λ. Each term in the summation can be efficiently computed using the quantum circuit shown in Fig. 7.1.

Real and imaginary time evolution are special class of this general evolution with $|dv(t)\rangle = -iH|v(t)\rangle$ and $|dv(t)\rangle = (-H - \langle v(t)|H|v(t)\rangle)|v(t)\rangle$, respectively [24, 46, 75]. Suppose $|v(t)\rangle = |\varphi(\vec{\theta}(t))\rangle$, then $\tilde{M} = \mathrm{Re}\left(\frac{\partial\langle\varphi(\vec{\theta}(t))|}{\partial\theta_i}\frac{\partial|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_j}\right)$ is the same for both real and imaginary time evolution; and $\tilde{V}$ is $\mathrm{Im}\left(\frac{\partial\langle\varphi(\vec{\theta}(t))|}{\partial\theta_i}H|\varphi(\vec{\theta}(t))\rangle\right)$ and $-\mathrm{Re}\left(\frac{\partial\langle\varphi(\vec{\theta}(t))|}{\partial\theta_i}H|\varphi(\vec{\theta}(t))\rangle\right)$ for real and imaginary time evolution, respectively. To

measure $\tilde{M}$ or $\tilde{V}$, we can employ an extra ancillary qubit with controlled operations applied between the ancilla and the system state. It is worth mentioning that it was recently shown that the ancilla and the constant number of controlled operations can be reduced [133].

7.3 Variational algorithms for linear algebra

Next we consider solving linear systems of equations and matrix-vector multiplications. For a sparse square matrix $\mathcal{M}$ and an (unnormalised) state vector $|v_0\rangle$, we hope to find

$$|v_{\mathcal{M}^{-1}}\rangle = \mathcal{M}^{-1}|v_0\rangle \quad \text{or} \quad |v_{\mathcal{M}}\rangle = \mathcal{M}|v_0\rangle, \tag{7.8}$$

for these two cases respectively. We first focus on matrix-vector multiplication by introducing a dynamical process that evolves the initial vector $|v_0\rangle$ to the target state $|v_{\mathcal{M}}\rangle$. One of the possible evolution paths is via a linear extrapolation between $|v_0\rangle$ and $|v_{\mathcal{M}}\rangle$ as

$$|v(t)\rangle = E(t)|v_0\rangle\,, \tag{7.9}$$

with $E(t) = t/T \cdot \mathcal{M} + \big(1 - t/T\big)I$, $|v(0)\rangle = |v_0\rangle$, and $|v(T)\rangle = |v_{\mathcal{M}}\rangle$. When we set $\mathcal{M} = e^{-iHT}$, it corresponds to Hamiltonian simulation.

We also consider linear extrapolation between normalised states. Given the evolution via linear extrapolation, the time derivative equation of $|v(t)\rangle$ is

$$\frac{\partial}{\partial t}|v(t)\rangle = G|v(0)\rangle\,, \tag{7.10}$$

with $G = \big(\mathcal{M} - I\big)/T$. This corresponds to the case with $A(t) = G$ and $|v'(t)\rangle = |v(0)\rangle$ in Eq. (7.1), which can be variationally simulated. We also consider the case where we are only interested in the normalised final state $|\psi(t)\rangle = \mathcal{M}|v_0\rangle / \sqrt{\|\mathcal{M}|v_0\rangle\|}$.

By extrapolating from $|v_0\rangle / \sqrt{\| |v_0\rangle \|}$ to $|\psi(t)\rangle$, we can similarly have an evolution of the state $|\psi(t)\rangle$ as

$$|\psi(t)\rangle = E'(t) |\psi_0\rangle , \tag{7.11}$$

with

$$E'(t) = N(t)\bigg(\frac{t}{T}\mathcal{M} + (1 - \frac{t}{T})I\bigg), \tag{7.12}$$

and a normalisation factor

$$N(t) = \frac{1}{\sqrt{\bigg\| \bigg(\frac{t}{T}\mathcal{M} + (1 - \frac{t}{T})I\bigg) |\psi_0\rangle \bigg\|}}. \tag{7.13}$$

The normalisation factor $N(t)$ can be measured from the expectation values of $\mathcal{M}^\dagger + \mathcal{M}$ and $\mathcal{M}^\dagger\mathcal{M}$ for $|\psi_0\rangle$. Given the definition of the state $|\psi(t)\rangle$ at time t, the corresponding derivative equation is

$$\frac{d}{dt} |\psi(t)\rangle = \frac{\dot{N}(t)}{N(t)} |\psi(t)\rangle + N(t)G |\psi(0)\rangle . \tag{7.14}$$

Such an equation is also described by the generalised time evolution equation with $|v(t)\rangle = |\psi(t)\rangle$, $A_1(t) = \frac{\dot{N}(t)}{N(t)}I$, $A_2(t) = N(t)G$, $|v_1'(t)\rangle = |\psi(t)\rangle$ and $|v_2'(t)\rangle = |\psi(0)\rangle$.

We now introduce a second method for realising matrix-vector multiplication using only real and imaginary time evolution. This method assumes an efficient singular value decomposition of $\mathcal{M}$ as $\mathcal{M} = UDV$, with unitary matrices U, V and diagonal matrix D with non-negative entries. Even when an efficient singular value decomposition is possible, the multiplication of a matrix to a quantum state is not physical process, which is nontrivial. In addition, as the state has exponentially large Hilbert space in the system size, the multiplication of a matrix is also classically hard, because to store the information of the state vector on classical computer is already inefficient. Suppose the unitary matrices U and

V can be represented by $U = \exp(-iH^U T^U)$ and $V = \exp(-iH^V T^V)$ with time T^U and T^V, respectively. Then the multiplication of U and V can be implemented by evolving the state with Hamiltonian H^U and H^V via variational real time simulation, where $H^U = -\sum_j \lambda_j / T^U \left|\lambda_j\right\rangle \left\langle\lambda_j\right|$ and similarly for V, given a spectral decomposition of U is expressed as $U = \sum_j e^{i\lambda_j} \left|\lambda_j\right\rangle \left\langle\lambda_j\right|$ with $\lambda_j \in \mathbb{R}$. To realise the diagonal matrix D, we first define a corresponding Hamiltonian H^D, such that $D \approx \exp(-H^D T^D)$. Suppose $D_k = \sum_j a_j \left|j\right\rangle \left\langle j\right|$, and the Hamiltonian H^D is given by $-H^D T = \sum_{a_j \neq 0} \log(a_j) \left|j\right\rangle \left\langle j\right| - \alpha \sum_{a_j = 0} \left|j\right\rangle \left\langle j\right|$, with properly chosen large constant α satisfying $\alpha \gg \log(a_j)$. Therefore, we can define an unnormalised imaginary time evolution $\frac{d\left|v(\tau)\right\rangle}{d\tau} = -H^D \left|v(\tau)\right\rangle$, so that the initial vector $\left|v_0\right\rangle$ is evolved to $D\left|v_0\right\rangle$ from $\tau = 0$ to $\tau = T$. This second method is easier to implement; it requires the singular value decomposition of $\mathcal{M}$ and evolution operator for the unitary and diagonal matrices. Although singular value decomposition cannot be implemented efficiently for a general large matrix, when considering only products of matrices that only involve a few qubits, i.e., $\mathcal{M} = \mathcal{M}_1 \otimes \cdots \otimes \mathcal{M}_L$ with $\mathcal{M}_i$ acting on a small constant number of qubits, singular value decomposition can be computed efficiently. We will shortly show that this variational algorithm is particularly useful for simulating the jump process of the stochastic Schrödinger equation.

We now discuss the solution of the linear equation $\mathcal{M}\left|v_{\mathcal{M}^{-1}}\right\rangle = \left|v_0\right\rangle$ with invertible matrix $\mathcal{M}$. We consider the extrapolation between $\left|v_0\right\rangle$ and $\left|v_{\mathcal{M}^{-1}}\right\rangle$ as

$$E(t)\left|v(t)\right\rangle = \left|v_0\right\rangle, \tag{7.15}$$

with $E(t) = t/T \cdot \mathcal{M} + \big(1 - t/T\big)I$, $\left|v(0)\right\rangle = \left|v_0\right\rangle$, and $\left|v(T)\right\rangle = \left|v_{\mathcal{M}^{-1}}\right\rangle$. By differenciating both sides of Eq. (7.15) with t, the derivation equation of $\left|v(t)\right\rangle$ is $E(t)\frac{\partial}{\partial t}\left|v(t)\right\rangle = -G(t)\left|v(t)\right\rangle$ with $G(t) = \big(\mathcal{M} - I\big)/T$. Although this is slightly different from the generalised time evolution of Eq. (7.1), we can still simulate it with the variational method by assuming $\left|v(t)\right\rangle = \left|\varphi(\vec{\theta}(t))\right\rangle$. Now, we derive the

time derivative equation of parameters for solving linear equations. We parametrise the state $|v(\vec{\theta}(t))\rangle$. According to McLachlan's principle [72], we have

$$\delta\|E(t)\frac{d}{dt}|v(\vec{\theta}(t))\rangle + G(t)|v(t)\rangle\| \tag{7.16}$$

$$= \delta\|E(t)\sum_j \dot{\theta}_j \frac{\partial}{\partial\theta_j}|v(\vec{\theta}(t))\rangle + G(t)|v(\vec{\theta}(t))\rangle\| = 0, \tag{7.17}$$

which is equivalent to

$$\frac{\partial}{\partial\dot{\theta}_k}\|E(t)\sum_j \dot{\theta}_j \frac{\partial}{\partial\theta_j}|v(\vec{\theta}(t))\rangle + G(t)|v(\vec{\theta}(t))\rangle\| \tag{7.18}$$

$$= \frac{\partial}{\partial\dot{\theta}_k}\Bigg[\Bigg(\sum_j \dot{\theta}_j\Bigg(\frac{\partial}{\partial\theta_j}\langle v(\vec{\theta}(t))|\Bigg)E^\dagger(t) + \langle v(\vec{\theta}(t))|G^\dagger(t)\Bigg) \tag{7.19}$$

$$\Bigg(E(t)\sum_l \dot{\theta}_l \frac{\partial}{\partial\theta_l}|v(\vec{\theta}(t))\rangle + G(t)|v(\vec{\theta}(t))\rangle\Bigg)\Bigg] = 0.$$

Then we obtain

$$\sum_j \tilde{M}_{k,j}\dot{\theta}_j = \tilde{V}_k, \tag{7.20}$$

where

$$\tilde{M}_{k,j} = \mathrm{Re}\Bigg(\frac{\partial\langle v(\vec{\theta}(t))|}{\partial\theta_k}E^\dagger(t)E(t)\frac{\partial|v(\vec{\theta}(t))\rangle}{\partial\theta_j}\Bigg) \tag{7.21}$$

$$\tilde{V}_k = -\mathrm{Re}\Bigg(\frac{\partial\langle v(\vec{\theta}(t))|}{\partial\theta_k}E^\dagger(t)G(t)|v(\vec{\theta}(t))\rangle\Bigg). \tag{7.22}$$

It is important to note that $\tilde{V}_k$ in Eq. (7.22) can be efficiently computed with the quantum circuit shown in Fig. 4.2. $\tilde{M}_{k,j}$ can written as

$$\tilde{M}_{k,j} = \sum_{i,q,l}\beta_l \mathrm{Re}\Bigg(g^*_{k,i}g_{j,q}\langle\bar{0}|U^\dagger_{k,i}\sigma_l U_{j,q}|\bar{0}\rangle\Bigg) \tag{7.23}$$

where we set $E^\dagger(t)E(t) = \sum_l \beta_l\sigma_l$, and σ_l is a Pauli operator. The quantum circuit

to compute this value is shown in Fig. 7.2,

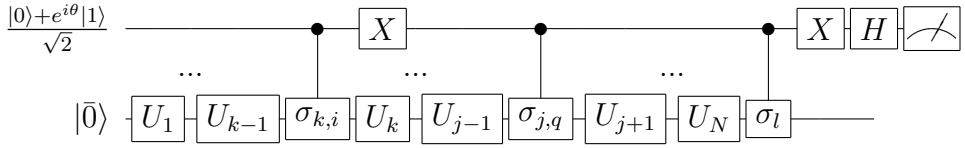


Figure 7.2: Quantum circuits that evaluate $\mathrm{Re}\left(g^*_{k,i} g_{j,q} \langle\bar{0}| R^\dagger_{k,i} \sigma_l R_{j,q} |\bar{0}\rangle \right)$.

Therefore we can calculate $|v_{\mathcal{M}^{-1}}\rangle = \mathcal{M}^{-1} |v_0\rangle$ by solving Eq. (7.20) and evolving the parameters accordingly.

7.4 Open system simulation

Now, we apply the developed variational algorithms to simulate the stochastic Schrödinger equation of open quantum systems. It is important to note that this approach differs from that in Ref. [75], where the evolution of the Lindblad master equation is directly simulated using a method that requires two copies of the states. Meanwhile, our scheme here only necessitates one copy of the state. Suppose the dynamics of open quantum systems is described by the Lindblad master equation,

$$\frac{d}{dt}\rho = -i[H, \rho] + \mathcal{L}\rho. \tag{7.24}$$

Here, the system Hamiltonian is H and the interaction with the environment $\mathcal{L}\rho$ is described by $\mathcal{L}\rho = \sum_k \frac{1}{2}(2L_k\rho L_k^\dagger - L_k^\dagger L_k\rho - \rho L_k^\dagger L_k)$, with Lindblad operators L_k. Although the Lindblad master equation directly evolves the density matrix, it is equivalent to the stochastic Schrödinger equation when we average the trajectory of each pure state evolved under continuous measurements [126, 34]. As the measurement process is stochastic, the wave function has a stochastic evolution.

We can obtain the stochastic Schrödinger equation for each single trajectory

$|\psi_c(t)\rangle$ as follows, given the Lindblad master equation

$$\begin{aligned} d\,|\psi_c(t)\rangle = &\left(-iH - \frac{1}{2}\sum_k (L_k^\dagger L_k - \langle L_k^\dagger L_k\rangle)\right)|\psi_c(t)\rangle\, dt \\ &+ \sum_k \left[\left(\frac{L_k\,|\psi_c(t)\rangle}{||L_k\,|\psi_c(t)\rangle\,||} - |\psi_c(t)\rangle\right) dN_k\right], \end{aligned} \tag{7.25}$$

where $d\,|\psi_c(t)\rangle = |\psi_c(t+dt)\rangle - |\psi_c(t)\rangle$, and dN_k is a random number taking either 0 or 1 which satisfies $dN_k dN_{k'} = \delta_{kk'} dN_k$ and $E[dN_k] = \langle\psi_c(t)|\, L_k^\dagger L_k\,|\psi_c\rangle\, dt$. We can interpret this process as a positive observable valued measurement $\{O_0 = I - \sum_k L_k^\dagger L_k dt,\, O_k = L_k^\dagger L_k dt\}$ happens at each time t. For measurement outcome O_k, the state discontinuously jumps to $L_k\,|\psi_c(t)\rangle\,/\|L_k\,|\psi_c(t)\rangle\,\|$ with probability $E[dN_k]$. and the total jump probability is $\gamma(t) = \sum_k E[dN_k]$. For outcome O_0 with probability $1-\gamma(t)$, we have $dN_k = 0$, $\forall k$ and the state evolves under the generalised time evolution with operator

$$A = -iH - \frac{1}{2}\sum_k (L_k^\dagger L_k - \langle L_k^\dagger L_k\rangle). \tag{7.26}$$

Here, $-iH$ corresponds to the conventional real time Schrödinger evolution with Hamiltonian H and the other terms can be understood as a normalised damping process. Therefore, the whole process is composed of two parts: the continuous process governed by the first term and the quantum jump process described by the second term in Eq. (7.25).

Now, we show the method to the simulate stochastic Schrödinger equation using the Monte Carlo method. Suppose the state experiences jump process at time t, then the probability $p(t+\tau)$ that the state does not jump until time $t+\tau$ is

$$p(t+\tau) = e^{-\Gamma(t,\tau)}, \tag{7.27}$$

with $\Gamma(t,\tau) = \int_t^{t+\tau} \gamma(t')dt'$. When a jump occurs at time t, a uniform random number $q \in [0,1]$ is generated. Then the time of the next jump is determined by

accumulating time τ until we have $p(t+\tau) = q$. When jump happens, a random number $q' \in [0,1]$ is generated to determine which jump operator to apply at each timestep. The state is updated to $L_k |\psi_c(t)\rangle / \| L_k |\psi_c(t)\rangle \|$ if $q' \in [\tilde{\gamma}_{k-1}(t), \tilde{\gamma}_k(t)]$, where

$$\tilde{\gamma}_k(t) = \frac{\sum_{l=1}^{k} \langle\psi_c(t)| L_l^\dagger L_l |\psi_c(t)\rangle}{\sum_{l=1}^{N_L} \langle\psi_c(t)| L_l^\dagger L_l |\psi_c(t)\rangle}, \tag{7.28}$$

and N_L is the number of the Lindblad operators. Considering discretised time with initial state $|\psi_c(0)\rangle$, the stochastic Schrödinger equation from time 0 to T can be simulated as follows.

Algorithm 1 Stochastic evolution equation

1: Set $\Gamma = 0$ and generate a random number $q \in [0,1]$.
2: **for** $t = 0 : dt : T$ **do**
3: **if** $e^{-\Gamma} \geq q$ **then**
4: Evolve the state $|\psi_c(t)\rangle$ under A in Eq. (7.26).
5: Calculate $\gamma(t) = \sum_k \langle\psi_c(t)| L_k^\dagger L_k |\psi_c\rangle \, dt$.
6: Update $\Gamma = \Gamma + \gamma(t)$.
7: **else**
8: Calculate $\tilde{\gamma}_k(t)$ in Eq. (7.28)
9: Generate a random number $q' \in [0,1]$.
10: **if** $q' \in [\tilde{\gamma}_{k-1}(t), \tilde{\gamma}_k(t)]$ **then**
11: Update $|\psi_c(t)\rangle$ to $L_k |\psi_c(t)\rangle / \| L_k |\psi_c(t)\rangle \|$.
12: Reset $\Gamma = 0$ and randomly generate $q \in [0,1]$.

Now we show how to simulate the stochastic Schrödinger equation, Algorithm 1, with the variational quantum algorithms developed in this work. Suppose the state $|\psi_c(t)\rangle$ at time t can be represented by the parametrised state $|\phi_c(\vec{\theta}(t))\rangle$ prepared by a quantum computer. We can simulate step 4, i.e., the evolution under operator A defined in Eq. (7.26), with the algorithm for generalised time evolution. Specifically, we can evolve the parameters according to Eq. (7.6) with $\tilde{M}_{k,j} = \mathrm{Re}\left(\frac{\partial\langle\varphi(\vec{\theta}(t))|}{\partial\theta_k} \frac{\partial|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_j}\right)$, $\tilde{V}_k = \mathrm{Re}\left(\langle\varphi(\vec{\theta}(t))| \left(-iH - (L - \langle L\rangle)\right) \frac{\partial|\varphi(\vec{\theta}(t))\rangle}{\partial\theta_k}\right)$, and $L = \frac{1}{2}\sum_k L_k^\dagger L_k$. Note that, the evaluation of $\tilde{V}_k$ for the simulation of the continuous part can be implemented with the same quantum circuit for the evaluation of $\tilde{V}_k$ for

real and imaginary time, but the phase of the ancilla qubit has to be appropriately set. The quantum circuit has only two controlled operations, and can be decomposed such that the ancilla is efficiently eliminated. The values of $\gamma(t)$ and $\tilde{\gamma}_k(t)$ at step 5 and 8, respectively, can be efficiently measured. The jump at step 11 can be realised by the variational algorithms for matrix-vector multiplication. Especially, when considering L_k as a product of operators acting on each qubit, it can be efficiently realised also with the singular value decomposition method, which uses the shallow depth circuit. In practice, we consider sparse Hamiltonian and Lindblad operators L_k, therefore all the measurements can be efficiently evaluated. As the evaluation of $\tilde{V}_k$ can also be implemented with the shallow quantum circuit, this algorithm for the simulation of stochastic Schrödinger equation does not rely on deep quantum circuit, and is compatible near-term quantum devices.

7.5 Resource estimation for simulating stochastic Schrödinger equation

Now, we have the discussion about resource estimation of this method. One trajectory of the stochastic Schrödinger equation consists of the continuous evolution and jump processes. The resource cost of the continuous evolution is similar to the one for real time evolution discussed in Ref. [24], which is shown to be polynomial to the evolution time and system size. For the jump processes, the resource cost of each jump is therefore also polynomial in the evolution time and system size, as we simulate each jump with singular value decomposition method, which is composed of real and imaginary time evolution.

Now we discuss how many jump processes occur on average in the simulation of the stochastic Schrödinger equation. The averaged number of jump events

$n_{\text{jump}}(t)dt$ during time from t to $t+dt$ is

$$n_{\text{jump}}(t)dt = \langle\psi_c(t)| \sum_k L_k^\dagger L_k |\psi_c(t)\rangle \, dt. \tag{7.29}$$

Therefore, the average number of jump events from $t=0$ to $t=T$ is

$$\begin{aligned} N_{\text{jump}} &= \int_0^T \langle\psi_c(t)| \sum_k L_k^\dagger L_k |\psi_c(t)\rangle \, dt, \\ &\leq T \left\| \sum_k L_k^\dagger L_k \right\|_2, \\ &\leq T \sum_k \left\| L_k^\dagger L_k \right\|_2, \end{aligned} \tag{7.30}$$

where $\|L\|_2$ is the largest singular value of L (operator norm). For physical systems, we generally have $\|L_k^\dagger L_k\|_\infty$ and the number of Lindblad terms equal $O(\text{Poly}(n))$, where n is the system size and $\text{Poly}(n)$ is a polynomial function of n. Therefore, the averaged number of jumps is

$$N_{\text{jump}} = O(T \cdot \text{Poly}(n)). \tag{7.31}$$

The number of jump events is much fewer when considering the case where each Lindblad operator only locally acts on a constant subsystem. That is, we assume that $L_k^\dagger L_k$ has orthogonal support to each other and $\|L_k^\dagger L_k\|_2 = O(1)$. Now, we expand the state $|\psi_c(t)\rangle = \sum_i c_i |\phi_i\rangle$, where $|\phi_i\rangle$ is the superposition state of states in the support of $L_i^\dagger L_i$. In this case, we have

$$\begin{aligned} \langle\psi_c(t)| \sum_k L_k^\dagger L_k |\psi_c(t)\rangle &= \sum_{i,j,k} c_i c_j^* \langle\phi_j| L_k^\dagger L_k |\phi_i\rangle \\ &= \sum_k |c_k|^2 \langle\phi_k| L_k^\dagger L_k |\phi_k\rangle \\ &\leq \sum_k |c_k|^2 \|L_k^\dagger L_k\|_2 = O(1). \end{aligned} \tag{7.32}$$

Therefore, $N_{\text{jump}} = O(T)$.

To simulate the stochastic Schrödinger equation, it is also necessary to sample different random trajectories. When we measure an observable O and hope to suppress the sampling error to $\epsilon = 1/\sqrt{M}$, we need M random samples. Therefore, the overall cost should also be multiplied by M, but it is worth noting that every trajectory can be computed in parallel.

7.6 Numerical simulation

Now we show an example of variational quantum simulation of a three-qubit dissipative 1D Ising model with a transverse field and open boundary, which is discussed in Refs. [134, 135, 136, 137]. With the Lindblad master equation as in Eq. (7.24), the Hamiltonian is $H_I = J/4 \sum_{i=1}^{2} Z_i Z_{i+1} + h_X \sum_{i=1}^{3} X_i + h_Z \sum_{i=1}^{3} Z_i$ with Pauli matrices X_i, Y_i, and Z_i on the i^{th} spin. The Lindblad terms are $L_i = \sqrt{\gamma}\sigma_i^-$, where $\sigma_i^- = |0\rangle\langle 1|_j$ are the lowering operator acting on the i^{th} spin. In our simulation, we set $J = 1$, $h_X = 1$, $h_Z = 0$, and $\gamma = 1$. We set the initial state to $|\varphi(0)\rangle = |0\rangle_1 |0\rangle_2 |0\rangle_3$, and we simulate the evolution from $t = 0$ to $t = 10$. In our simulation, we utilised the Hamiltonian ansätz [138] as shown in Fig. 7.3 with ten parameters.

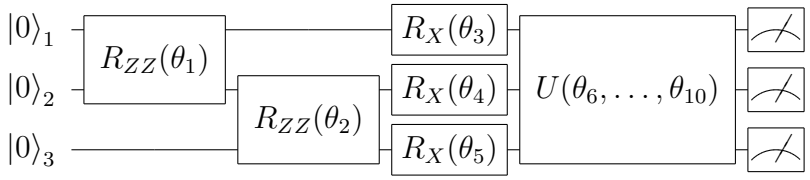


Figure 7.3: Hamiltonian ansätz in our numerical simulation. The single qubit gate is defined by $R_X(\theta_i) = e^{-i\theta_i X}$ and the two qubit gate is $R_{ZZ}(\theta_i) = e^{-i\theta_i Z \otimes Z}$. The last gate $U(\theta_6, \ldots, \theta_{10})$ is a repetition of the first five gates with five different parameters. In total, there are ten parameters.

To simulate quantum events induced by Lindblad operators, we use the singular value decomposition method. We decompose the jump operator σ^- as $|0\rangle\langle 1| =$

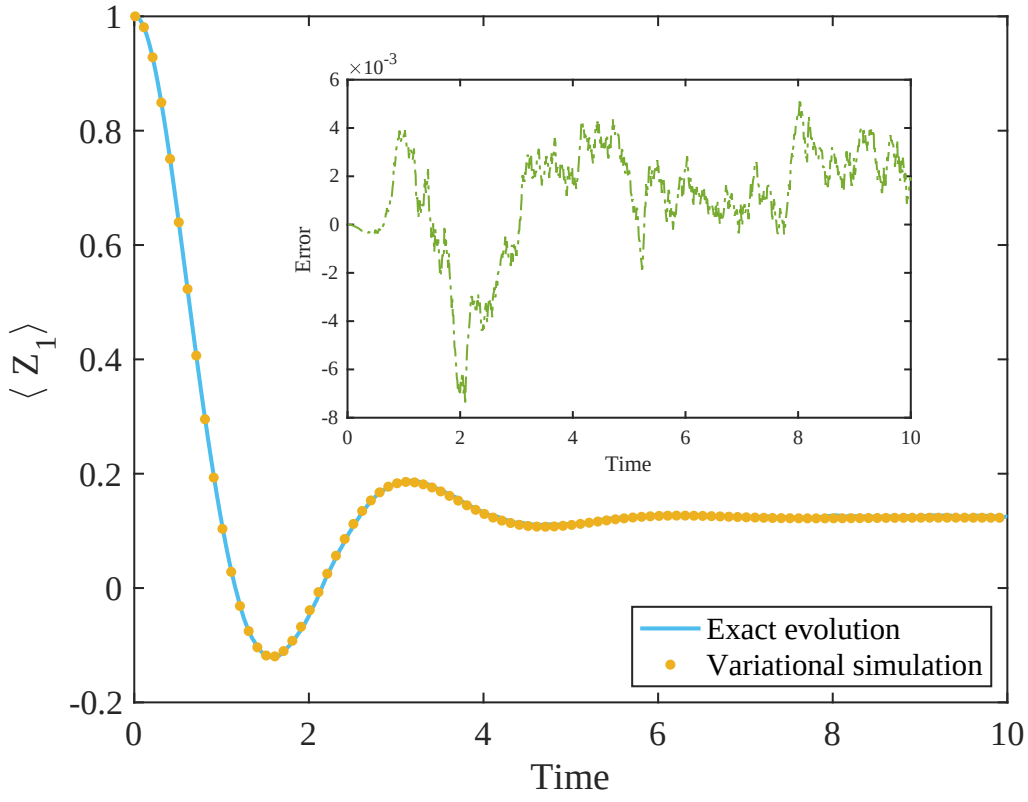


Figure 7.4: The comparison between the exact evolution and the variational simulation from $t = 0$ to $t = 10$. The Y axis is the average of the Z_1 operator. The time step is $\delta t = 0.01$. The result of the variational algorithm is averaged over $N_{\text{trial}} = 8 \times 10^4$ times. The inset shows the difference between the exact evolution and the variational simulation result.

$|0\rangle\langle 0| X$. Suppose $|0\rangle\langle 1| = UDV$, we can see that $U = I$, $D = |0\rangle\langle 0|$, $V = X$. To realise the V operator, we set $H^V = X$ and $T^V = \pi/2$ such that $X = \exp(-iH^V T^V)$. Then we evolve the state under Hamiltonian H^V for time T^V with time step $\delta t^V = 0.01$ to have $X|\varphi(\vec{\theta})\rangle$. For implementing $D = |0\rangle\langle 0|$, we set $H^D = |1\rangle\langle 1|$ and $T^D = 10$ so that $D \approx \exp(-H^D T^D)$. Then we realise $D|\varphi(\vec{\theta})\rangle / \| D|\varphi(\vec{\theta})\rangle \|$ by using the normalised variational imaginary time evolution with total time T^D and time step $\delta t^D = 0.1$.

In Fig. 7.4, we compare the simulation result of the variational algorithm with the exact solution. We measure the average value of Z_1 and we can see that the variational simulation result shows a good agreement with the exact solution with a deviation less than 10^{-2}.

7.7 Discussion

To summarise, we generalised the variational quantum simulation method so that general processes can be simulated, including the generalised time evolution, matrix-vector multiplication, and the evolution of open quantum systems. Our algorithm for simulating the generalised time evolution can be applied to simulate non-Hermitian quantum mechanics [104, 105, 106] such as nonequilibrium processes [107] and parity-time symmetric Hamiltonians [108, 109]. Especially, in Ref. [109], it is shown that a quantum state can evolve to the target state faster by using non-Hermitian parity-time symmetric Hamiltonians than the case with Hermitian Hamiltonians. Therefore, for designing faster quantum computing algorithms, our variational quantum algorithm for simulating the generalised time evolution may also be useful. Meanwhile, the proposed algorithms are compatible with NISQ hardwares and can be further combined with the recently proposed quantum error mitigation techniques [24, 81, 82, 28, 29, 84, 79], which we will discuss in the following chapters.

Chapter 8

Practical error mitigation

8.1 Introduction

In chapter 5, we reviewed quantum error mitigation (QEM) techniques to suppress physical errors, i.e. the extrapolation method based on Richardson extrapolation [77, 24, 28] and a quasi-probability method [28]. Although these methods are very important in that they showed it is possible to mitigate errors on quantum computers without quantum error correction and they are compatible with near-term quantum devices and hybrid quantum-classical algorithms, each method has disadvantages and was not practical to apply in experiments. For example, the extrapolation method based on Richardson extrapolation, e.g. linear extrapolation, cannot tolerate high error rate. We will demonstrate this fact later, and our new extrapolation method, i.e. exponential extrapolation significantly outperforms linear extrapolation. Moreover, there were limitations with the quasi-probability method when it was first proposed by an IBM group, say, it could only be applied to restricted types of noise models, such as depolarising noise and amplitude damping, and, although we have to know the noise model explicitly via tomography, the suitable tomography method for this method was not clarified. By introducing a linearly independent universal operation set, it is shown that general error mod-

els can be mitigated. In addition, we found that by using gate set tomography (GST) [139, 140], we can even eliminate the state preparation and measurement error accompanying the tomography process, and full elimination of the impact of localised Markovian errors has been enabled.

As a test case, we numerically simulated SWAP test circuit up to 19 qubits by using high performance quantum circuit simulator QuEST [141], and demonstrated that exponential extrapolation and improved quasi-probability method in the regime where the average number of error events in the quantum circuit is close to unity. We also confirmed that by using the current error rate for ion trap systems [15, 16], mitigating errors for SWAP test up to approximately 80 qubits is feasible, paving the way to quantum supremacy regime.

The author was solely responsible for the work in sections 8.10, 8.11, 8.12, 8.13 and 8.14, while he jointly derived the results in sections 8.5, 8.6, 8.7, 8.8, and 8.9 with coauthor Ying Li. Section 8.10 is predominantly the work of coauthor Ying Li but it is included here for completeness. Any data purely due to a coauthor is labelled as such.

8.2 Error mitigation

Firstly, we again formulate QEM (especially as to quasi-probability method) so that we can discuss it clearly in later sections. We consider computing the expectation value of an observable in a state (the final state of a quantum circuit) using a quantum computer, as the error mitigation can mitigate error of the expectation value, not the quantum state itself like quantum error correction as explained in chapter 5. It is typical that, for a number of quantum algorithms and subroutines [24, 19, 22, 25, 26, 142], the desired output is the expected value of a qubit or qubits – one example is the SWAP-test [94, 91] itself, which is a component of algorithms including the recently-introduced auto encoder [143], and several proposed hybrid

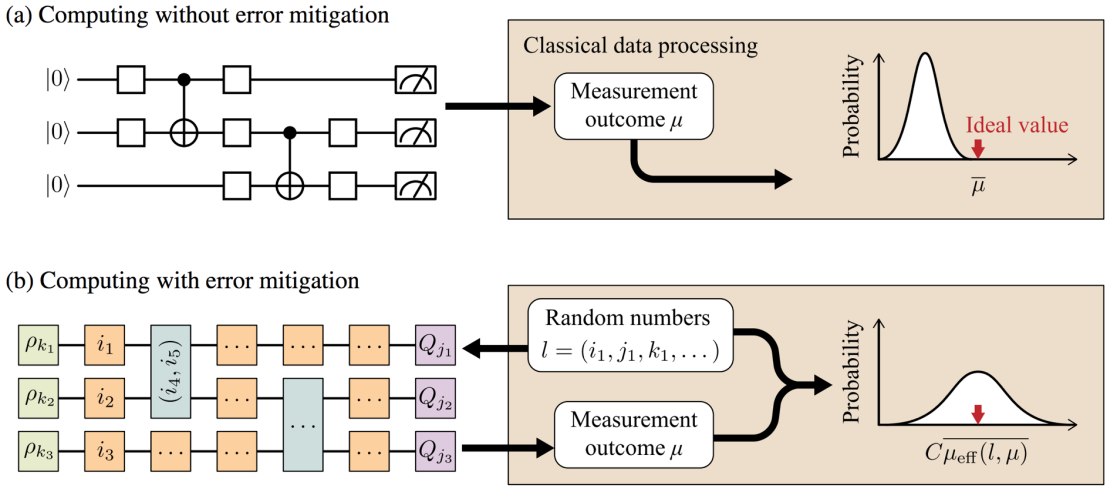


Figure 8.1: Quantum computing of the expected value of an observable (a) without quantum error mitigation (QEM) and (b) with QEM. In QEM circuit (b), each operation (including the memory operation) in the original circuit (a) is replaced by an operation depending on the corresponding random numbers [see Fig. 8.2(a)].

algorithms for simulating chemical or materials systems. In the case that we do not utilise QEM as shown in Fig. 8.1(a), the quantum circuit is repeated many times, and the measurement outcome μ of each time is collected. Then, we can calculate the average $\overline{\mu}$ as our best estimate of the expected value. Given that the number of repetitions is finite, the value of $\overline{\mu}$ is a random variable with an associated distribution. Because the implementation of the quantum circuit is imperfect, it is likely that the distribution of $\overline{\mu}$ is not even centered at the ideal value, i.e. the exact expected value when the quantum circuit is perfectly implemented without error.

When we use QEM as shown in Fig. 8.1(b), instead of the original quantum circuit, we implement a set of modified circuits. The scheme depicted in the figure is relevant to the quasi-probability method for QEM, but can also apply to the extrapolation method as a means to deliberately boost errors. Each modified circuit is determined by a set of random numbers l. The distribution of random numbers, i.e. modified circuits, depends on the error model, which is measured using GST before the quantum computing. In each run of the quantum experiment, firstly the random number set l is generated, then depending on l a specific circuit is

implemented, and finally the measurement outcome μ is collected. Rather than calculating the average $\overline{\mu}$, we use both l and μ to calculate the average of an effective outcome $\mu_{\text{eff}}(l,\mu)$, which will be given explicitly later. If QEM is successful, the distribution of $C\overline{\mu_{\text{eff}}(l,\mu)}$ is centered at the ideal value, but the distribution is wider than $\bar{\mu}$ because of the factor C, which is greater than 1 and can be efficiently computed. Thus only error due to the statistical fluctuation remains, although it is amplified. By repeating the quantum experiment enough times, we can obtain an accurate computing result of the expected value.

In Sec. 8.4, we explicitly give the effective outcome $C\overline{\mu_{\text{eff}}(l,\mu)}$. Modified circuits and their distribution are given in Sec. 8.8.

8.3 Pauli transfer matrix and notation for states, operators and operations

We use the notation commonly used in quantum tomography (e.g. in Refs. [139, 140]). In quantum theory, a quantum state is usually represented by a density matrix ρ, and an observable is represented by a Hermitian operator Q. The expected value of the observable quantity in the state is $\langle Q\rangle = \mathrm{Tr}(Q\rho)$. An operation is a map on the space of states, $\mathcal{O}(\rho) = \sum_k E_k \rho E_k^\dagger$, expressed in the Kraus form.

Because an operation is a linear map, we can always express the operation $\mathcal{O}$ as a square matrix, e.g. using the Pauli transfer matrix representation, acting on the state expressed as a column vector $|\rho\rangle\rangle$. Similarly, an observable can be expressed as a row vector $\langle\langle Q|$, and the expected value is $\langle Q\rangle = \langle\langle Q|\rho\rangle\rangle$. Throughout this chapter, we use the Pauli transfer matrix representation.

The explanation of Pauli transfer matrix is as follows. A state ρ can be expressed as a real column vector

$$|\rho\rangle\rangle = \begin{bmatrix} \cdots & \rho_\sigma & \cdots \end{bmatrix}^{\mathrm{T}}, \tag{8.1}$$

where the vector element is

$$\rho_\sigma = \mathrm{Tr}(\sigma\rho), \tag{8.2}$$

$\sigma \in \{I, \sigma^{\mathrm{x}}, \sigma^{\mathrm{y}}, \sigma^{\mathrm{z}}\}^{\otimes n}$ is a Pauli operator, and $d = 2^n$ is the dimension of the Hilbert space. Similarly, an observable (i.e. Hermitian operator) Q can be expressed as a real row vector

$$\langle\langle Q| = \begin{bmatrix} \cdots & Q_\sigma & \cdots \end{bmatrix}, \tag{8.3}$$

where the vector element is

$$Q_\sigma = d^{-1}\mathrm{Tr}(\sigma Q). \tag{8.4}$$

Here, we use notations $\langle\langle\cdot|$ and $|\cdot\rangle\rangle$ to denote real row and column vectors, respectively. A physical operation $\mathcal{O}$ (i.e. $\mathcal{O}(\rho) = \sum_k E_k \rho E_k^\dagger$) can be expressed as a real square matrix

$$\mathcal{O}_{\sigma,\tau} = d^{-1}\mathrm{Tr}[\sigma\mathcal{O}(\tau)], \tag{8.5}$$

where $\sigma, \tau \in \{I, \sigma^{\mathrm{x}}, \sigma^{\mathrm{y}}, \sigma^{\mathrm{z}}\}^{\otimes n}$ are Pauli operators. If $\rho' = \mathcal{O}(\rho)$, we have $|\rho'\rangle\rangle = \mathcal{O}|\rho\rangle\rangle$.

In the Pauli transfer matrix representation, vectors representing states or observables and matrices representing operations are all real. For n qubits, vectors and matrices are 4^n-dimensional. The expected value of the observable Q in the state ρ going through a sequence of operations $\mathcal{O}_1, \ldots, \mathcal{O}_N$ reads as follows:

$$\mathrm{Tr}[Q\mathcal{O}_N \circ \cdots \circ \mathcal{O}_1(\rho)] = \langle\langle Q|\mathcal{O}_N \cdots \mathcal{O}_1|\rho\rangle\rangle.$$

8.4 Quantum computing by sampling circuits

We suppose that the initial state is $\rho^{(0)}$, which goes through a sequence of operations $\mathcal{O}_1^{(0)}, \ldots, \mathcal{O}_N^{(0)}$, and in the final state the observable $Q^{(0)}$ is measured. Each time the experimentalist implements this circuit, the measurement returns an eigenvalue of $Q^{(0)}$, and the probability distribution of eigenstates is determined by the final state. By repeating such a circuit for many times, they can estimate the expected value $\langle Q^{(0)} \rangle = \langle\langle Q^{(0)} | \mathcal{O}_{\text{tot.}}^{(0)} | \rho^{(0)} \rangle\rangle = \text{E}[\mu^{(0)}]$, where $\mathcal{O}_{\text{tot.}}^{(0)} = \mathcal{O}_N^{(0)} \cdots \mathcal{O}_1^{(0)}$, and $\mu^{(0)}$ is the measurement outcome. Generally in this chapter we will use the superscript 0 to denote the ideal noise-free realisation of a state, operation or observable quantity.

In the case that the quantum computation has errors, the actual initial state is ρ, actual operations are $\mathcal{O}_1, \ldots, \mathcal{O}_N$, and the actually measured observable is Q. As a result, the estimation of the expected value converges to $\langle Q \rangle = \langle\langle Q | \mathcal{O}_{\text{tot.}} | \rho \rangle\rangle$ rather than $\langle Q^{(0)} \rangle$. Here, $\mathcal{O}_{\text{tot.}} = \mathcal{O}_N \cdots \mathcal{O}_1$, and we have assumed that errors are Markovian, i.e., errors associated with different gates are totally independent.

Now, we discuss the idea of the quasi-probability method introduced by the IBM team [28] which we outilined in chapter 5 by using the notation we introduced here. One can mitigate the effect of errors by sampling from a set of (real, error-burdened) circuits, each labelled $\mathcal{O}_{\text{tot.}}^{(l)}$ for $l = 1, 2...$, provided that their outputs satisfy

$$\langle Q^{(0)} \rangle = \sum_l q_l \langle\langle Q^{(l)} | \mathcal{O}_{\text{tot.}}^{(l)} | \rho^{(l)} \rangle\rangle.$$

Ref. [28] describes how the real numbers $\{q_l\}$ which represent quasi-probabilities can be efficiently derived given specific error models, assuming that the experimentalist has full knowledge of the model. Note that each $\mathcal{O}_{\text{tot.}}^{(l)}$ denotes the total operation composed by a sequence of operations in the l^{th} circuit.

We can use the Monte Carlo method to compute $\langle Q^{(0)} \rangle$. We note that $\langle\langle Q^{(l)} | \mathcal{O}_{\text{tot.}}^{(l)} | \rho^{(l)} \rangle\rangle = \text{E}[\mu^{(l)}]$, where $\mu^{(l)}$ is the measurement outcome in the l^{th} circuit. Then $\langle Q^{(0)} \rangle = \sum_l |q_l| \text{E}[\text{sgn}(q_l)\mu^{(l)}]$. To compute $\langle Q^{(0)} \rangle$, we randomly choose a circuit to implement,

and the l^{th} circuit is chosen with the probability $p_l = |q_l|/C$, where $C = \sum_l |q_l|$. Then, the computing result is given by the expected value of effective measurement outcomes, i.e. $\langle Q^{(0)} \rangle = C\mathrm{E}[\mu_{\text{eff}}]$, where the effective outcome is $\mu_{\text{eff}} = \mathrm{sgn}(q_l)\mu^{(l)}$ if the l^{th} circuit is chosen to be implemented, and $\mu^{(l)}$ is the outcome directly obtained in the l^{th} circuit.

8.5 Per-operation error correction

We can correct errors in each operation using the quasi-probability method, which will be the primary focus for the following several sections. We can also apply the quasi-probability method for suppressing errors for preparing the inital state and measurement, which was not discussed in Ref. [28]. We suppose that we have a set of initial states satisfying $|\rho^{(0)}\rangle\rangle = \sum_{l_{\text{in}}} q_{l_{\text{in}}}^{[\text{in}]} |\rho^{(l_{\text{in}})}\rangle\rangle$, and a set of operations satisfying $\mathcal{O}_i^{(0)} = \sum_{l_i} q_{l_i}^{[i]} \mathcal{O}_i^{(l_i)}$ for each error-free operation $\mathcal{O}_i^{(0)}$, and a set of observables satisfying $\langle\langle Q^{(0)}| = \sum_{l_{\text{out}}} q_{l_{\text{out}}}^{[\text{out}]} \langle\langle Q^{(l_{\text{out}})}|$. Then, by applying error mitigation, we have a computation result as

$$\begin{aligned} \langle Q^{(0)} \rangle \;=\; & \sum_{l_{\text{in}}} \sum_{l_1} \cdots \sum_{l_N} \sum_{l_{\text{out}}} q_{l_{\text{in}}}^{[\text{in}]} q_{l_1}^{[1]} \cdots q_{l_N}^{[N]} q_{l_{\text{out}}}^{[\text{out}]} \\ & \times \langle\langle Q^{(l_{\text{out}})} | \mathcal{O}_N^{(l_N)} \cdots \mathcal{O}_1^{(l_1)} | \rho^{(l_{\text{in}})} \rangle\rangle. \end{aligned} \tag{8.6}$$

When we sample circuits to compute $\langle Q^{(0)} \rangle = C\mathrm{E}[\mu_{\text{eff}}]$, the initial state is $|\rho^{(l_{\text{in}})}\rangle\rangle$ with probability $p_{l_{\text{in}}}^{[\text{in}]} = q_{l_{\text{in}}}^{[\text{in}]}/C_{\text{in}}$, the i^{th} operation is $\mathcal{O}_i^{(l_i)}$ with probability $p_{l_i}^{[i]} = |q_{l_i}^{[i]}|/C_i$, and the observable is $\langle\langle Q^{(l_{\text{out}})}|$ with probability $p_{l_{\text{out}}}^{[\text{out}]} = q_{l_{\text{out}}}^{[\text{out}]}/C_{\text{out}}$. Here, $C_\alpha = \sum_{l_\alpha} |q_{l_\alpha}^{[\alpha]}|$, and $C = C_{\text{in}} C_1 \cdots C_N C_{\text{out}}$ accordingly. To calculate μ_{eff}, we use $\mathrm{sgn}(q_{l_{\text{in}}}^{[\text{in}]} \cdots q_{l_{\text{out}}}^{[\text{out}]}) = \mathrm{sgn}(q_{l_{\text{in}}}^{[\text{in}]}) \cdots \mathrm{sgn}(q_{l_{\text{out}}}^{[\text{out}]})$.

8.6 Variance amplification in quasi-probability decomposition

As quasi-probabilities take negative values, it amplifies the variance of the expected value of the observable. We consider the case that $Q^{(l)}$ is a Pauli operator (maybe with error) and the measurement results are denoted by ± 1, hence the probability distribution is binomial. The standard deviation of the average of outcomes in the Monte Carlo calculation is $\sigma = C\sqrt{(1 - \mathrm{E}[\mu_{\mathrm{eff}}]^2)/N_{\mathrm{r}}} \leq C/\sqrt{N_{\mathrm{r}}}$, where N_{r} is the total number of samples, i.e. the total number of circuits of all kinds which the experimentalist performs is N_{r}. We compare this to the error-free computing, i.e. the ideal original circuit $\langle\langle Q^{(0)}|\mathcal{O}^{(0)}_{\mathrm{tot.}}|\rho^{(0)}\rangle\rangle$ is repeated for $N_{\mathrm{r}}^{(0)}$ times to estimate $\langle Q^{(0)}\rangle$. For the error-free computing, the standard deviation is given by $\sigma^{(0)} = \sqrt{(1 - \mathrm{E}[\mu^{(0)}]^2)/N_{\mathrm{r}}^{(0)}}$. Therefore, to achieve the same accuracy, i.e. $\sigma = \sigma^{(0)}$, the error-mitigated computation requires $N_{\mathrm{r}}/N_{\mathrm{r}}^{(0)} = (C^2 - \langle Q^{(0)}\rangle^2)/(1 - \langle Q^{(0)}\rangle^2)$ times more samples than the error-free computation. Here, we have used the fact that the error-mitigated computation and the error-free computation should converge to the same value of $\langle Q^{(0)}\rangle$, i.e. $\mathrm{E}[\mu^{(0)}] = C\mathrm{E}[\mu_{\mathrm{eff}}]$.

In order to limit the standard deviation to be $\sigma \sim \epsilon$, we can choose $N_{\mathrm{r}} \sim (C/\epsilon)^2$. Therefore, if the factor C is larger, the computation takes longer.

Because $C = C_{\mathrm{in}}C_1 \cdots C_N C_{\mathrm{out}}$ if errors are corrected for each operation, we call $C_\alpha - 1$ the cost for mitigating error in the corresponding operation. The overall cost therefore increases with the number of operations, thus it is important to reduce the operation number. For example, in a quantum computer where qubits are fully connected [144], operations for communication are not required, which may significantly reduce the cost.

1	$[I]$ (no operation)
2	$[\sigma^{\mathrm{x}}] = [R_{\mathrm{x}}]^2$
3	$[\sigma^{\mathrm{y}}] = [R_{\mathrm{x}}]^2[R_{\mathrm{z}}]^2$
4	$[\sigma^{\mathrm{z}}] = [R_{\mathrm{z}}]^2$
5	$[R_{\mathrm{x}}] = [\frac{1}{\sqrt{2}}(I + i\sigma^{\mathrm{x}})] = [H][S]^3[H]$
6	$[R_{\mathrm{y}}] = [\frac{1}{\sqrt{2}}(I + i\sigma^{\mathrm{y}})] = [R_{\mathrm{z}}]^3[R_{\mathrm{x}}][R_{\mathrm{z}}]$
7	$[R_{\mathrm{z}}] = [\frac{1}{\sqrt{2}}(I + i\sigma^{\mathrm{z}})] = [S]^3$
8	$[R_{\mathrm{yz}}] = [\frac{1}{\sqrt{2}}(\sigma^{\mathrm{y}} + \sigma^{\mathrm{z}})] = [R_{\mathrm{x}}][R_{\mathrm{z}}]^2$
9	$[R_{\mathrm{zx}}] = [\frac{1}{\sqrt{2}}(\sigma^{\mathrm{z}} + \sigma^{\mathrm{x}})] = [R_{\mathrm{z}}][R_{\mathrm{x}}][R_{\mathrm{z}}]$
10	$[R_{\mathrm{xy}}] = [\frac{1}{\sqrt{2}}(\sigma^{\mathrm{x}} + \sigma^{\mathrm{y}})] = [R_{\mathrm{x}}]^2[R_{\mathrm{z}}]$
11	$[\pi_{\mathrm{x}}] = [\frac{1}{2}(I + \sigma^{\mathrm{x}})] = [R_{\mathrm{z}}]^3[R_{\mathrm{x}}]^3[\pi][R_{\mathrm{x}}][R_{\mathrm{z}}]$
12	$[\pi_{\mathrm{y}}] = [\frac{1}{2}(I + \sigma^{\mathrm{y}})] = [R_{\mathrm{x}}][\pi][R_{\mathrm{x}}]^3$
13	$[\pi_{\mathrm{z}}] = [\frac{1}{2}(I + \sigma^{\mathrm{z}})] = [\pi]$
14	$[\pi_{\mathrm{yz}}] = [\frac{1}{2}(\sigma^{\mathrm{y}} + i\sigma^{\mathrm{z}})] = [R_{\mathrm{z}}]^3[R_{\mathrm{x}}]^3[\pi][R_{\mathrm{x}}]^3[R_{\mathrm{z}}]$
15	$[\pi_{\mathrm{zx}}] = [\frac{1}{2}(\sigma^{\mathrm{z}} + i\sigma^{\mathrm{x}})] = [R_{\mathrm{x}}][\pi][R_{\mathrm{x}}]^3[R_{\mathrm{z}}]^2$
16	$[\pi_{\mathrm{xy}}] = [\frac{1}{2}(\sigma^{\mathrm{x}} + i\sigma^{\mathrm{y}})] = [\pi][R_{\mathrm{x}}]^2$

Table 8.1: Sixteen basis operations. Gates $[R_{\mathrm{x}}]$ and $[R_{\mathrm{y}}]$ can be derived from $[H]$ and $[S]$, and other operations can be derived from $[\pi]$, $[R_{\mathrm{x}}]$ and $[R_{\mathrm{y}}]$.

8.7 Universal operation set

The set of operations including measurement and single-qubit Clifford gates is universal in computing expected values of observables. The relevant measurement operation reads $[\pi] = [\frac{1}{2}(I + \sigma^{\mathrm{z}})]$, which projects a qubit to the state $|0\rangle$. Here, $[U](\rho) = U\rho U^{\dagger}$ denotes a superoperator. Such a non-destructive measurement can be realised using a destructive measurement followed by initialising the qubit in the state $|0\rangle$. Single-qubit Clifford gates include the Hadamard gate $[H] = [\frac{1}{\sqrt{2}}(\sigma^{\mathrm{x}}+\sigma^{\mathrm{z}})]$, the phase gate $[S] = [\frac{1}{\sqrt{2}}(I - i\sigma^{\mathrm{z}})]$ and all other single-qubit Clifford gates can be derived from these two.

In Table 8.1, we list sixteen linearly independent operations that can be derived from the minimum universal operation set $\{[\pi], [H], [S]\}$. In the following, we use $\{\mathcal{B}_i^{(0)} | i = 1, \ldots, 16\}$ to denote these sixteen operations. As they are linearly independent, any single-qubit operation $\mathcal{O}$, which is a 4×4 real matrix, can be expressed as a linear combination of sixteen basis operations, i.e. $\mathcal{O} = \sum_{i=1}^{16} q_i \mathcal{B}_i^{(0)}$.

Similarly, multi-qubit operations can be expressed as a linear combination of tensor products of basis operations. Using the quasi-probability method, any computation of expected values of observables can be realised using this operation set.

We can also use the overcomplete basis operations to construct quasi-probability operations, such as 24 Clifford operations for a single qubit. In this case, we solve a L1 norm minimisation problem as [145]

$$\begin{aligned} &\text{minimise} \sum_i |q_i| \\ &\text{subject to } \mathcal{O} = \sum_i q_i \mathcal{B}_i, \end{aligned} \tag{8.7}$$

where the cost is $C = \sum_i |q_i|$. This optimisation problem can be efficiently solve by using, for example, CVX, Matlab Software for Disciplined Convex Programming [146]. By using this formalism, it may be possible to further reduce the cost for the quasi-probability decomposition, and we will leave the discussion about the reduction of the cost for future work.

These basis operations are universal, as one can verify by constructing a non-Clifford gate or an entangling gate. For instance, we can decompose T gate using our basis operations as $[T] = \frac{1}{2}[I] - \frac{\sqrt{2}-1}{2}[\sigma^{\mathrm{z}}] + \frac{\sqrt{2}}{2}[R^3_{\mathrm{z}}]$, and the corresponding cost $C = \sqrt{2}$. It is worth mentioning that $C = \sqrt{2}$ is the same as the value called robustness of magic, indicating the classical simulation overhead [145].

Another example is controlled-NOT gate, which reads

$$\Lambda_{\mathrm{X}} = \frac{I + \sigma^{\mathrm{z}}}{2} \otimes I + \frac{I - \sigma^{\mathrm{z}}}{2} \otimes \sigma^{\mathrm{x}}, \tag{8.8}$$

and the controlled-NOT gate can be decomposed as

$$\begin{aligned}[\Lambda_{\rm X}] &= \frac{1}{2}([I\otimes\sigma^{\rm x}]+[\sigma^{\rm z}\otimes I]-[I\otimes R_{\rm x}] \\ &\quad -[R_{\rm z}\otimes I]-[\sigma^{\rm z}\otimes R_{\rm x}]-[R_{\rm z}\otimes\sigma^{\rm x}]) \\ &\quad +[\sigma^{\rm z}\otimes\sigma^{\rm x}]+[R_{\rm z}\otimes R_{\rm x}]+[I\otimes\pi_{\rm x}] \\ &\quad +[\pi_{\rm z}\otimes I]-[\sigma^{\rm z}\otimes\pi_{\rm x}]-[\pi_{\rm z}\otimes\sigma^{\rm x}]. \end{aligned} \tag{8.9}$$

Then, the corresponding cost is given by $C = 9$.

However, these constructions would not be used in practice – it is not an efficient method to actually implement a desired T and controlled-NOT in the basic circuit, since the corresponding cost $C = \sqrt{2}$ and $C = 9$ would imply unacceptably steep exponential, leading to a huge sampling cost, as one would expect from e.g. Refs. [147, 148, 149]. Instead we rely on the assumption that the experimental system can directly implement a universal set of gates (including entangling and non-Clifford gates) with a reasonably high fidelity. Then rather than fully synthesising any of the basic gates using our basis, we need only compensate for slight imperfections. The cost for doing so, for each imperfect gate, is then $\sim C = 1 + \delta$ as we presently discuss.

Note that the measurement superoperator $[\pi]$ also means post-selection, i.e. if the outcome of the measurement corresponding to $[\pi]$ (which is not the final measurement on the observable $Q^{(l)}$) is $|1\rangle$ in a trial, the value of the observable $Q^{(l)}$ is noted as $\mu^{(l)} = 0$, but the trial is counted in the total number of samples in the Monte Carlo calculation. If $Q^{(l)}$ has two values ± 1, we can estimate the value of $\langle\langle Q^{(l)}|\mathcal{O}^{(l)}_{\rm tot.}|\rho^{(l)}\rangle\rangle$ by calculating $(N^{(l)}_{+1} - N^{(l)}_{-1})/(N^{(l)}_0 + N^{(l)}_{+1} + N^{(l)}_{-1})$. Here, we have supposed that the circuit is implemented for total $N^{(l)}_0 + N^{(l)}_{+1} + N^{(l)}_{-1}$ times; for $N^{(l)}_0$ times the circuit does not pass post-selections (i.e. $\mu^{(l)} = 0$), and for $N^{(l)}_{\pm 1}$ times the circuit passes all post-selections and reports $Q^{(l)} = \pm 1$ (i.e. $\mu^{(l)} = \pm 1$). It is the same when we compute $\langle Q^{(0)}\rangle$ using the Monte Carlo method. If the effect outcome

is $\mu_{\text{eff}} = 0$ with the probability P_0, then the standard deviation of the Monte Carlo calculation becomes $\sigma = C\sqrt{[(1-P_0)^2 - \mathrm{E}[\mu]^2]/(1-P_0)N_{\text{r}}} \leq C\sqrt{(1-P_0)/N_{\text{r}}}$.

As we have obtained the complete operation set in Table 8.1, we can use it in deriving the protocol that will compensate for errors. We focus on the case that errors are localised: An (error-free) operation that is applied on a set of qubits S is a $4^{|S|}$-dimensional real matrix, then the corresponding operation in real (i.e. error-burdened) $\mathcal{O}$ is also a $4^{|S|}$-dimensional real matrix operating on the same set of qubits. The overall operation on the entire system can be expressed as $I_{\bar{S}} \otimes \mathcal{O}$, where $I_{\bar{S}}$ is the identity acting on all other qubits. It is similar for the initialisation and measurement. If each qubit is initialised individually, the overall initial state is $\bigotimes_m |\rho_m\rangle\rangle$, where $|\rho_m\rangle\rangle$ is a 2-dimensional real vector representing the m^{th} qubit's initial state. Similarly, individual measurement of qubits implies that the overall measured observable is $\bigotimes_m \langle\langle Q_m|$, where $\langle\langle Q_m|$ is a 2-dimensional real vector representing the measured observable for the m^{th} qubit. In this case, a single-qubit operation with error can still be expressed using a 4×4 real matrix. We suppose that for a qubit, sixteen basis operations with errors are $\{\mathcal{B}_i | i = 1, \ldots, 16\}$, which are all 4×4 real matrices. When errors are not significant, these sixteen bases should still be linearly independent, i.e. the set of basis operations with errors is still universal.

To make this statement more precise, we consider the 16×16 real matrix

$$A = \begin{bmatrix} (\mathcal{B}_1)_{\bullet,1} & \cdots & (\mathcal{B}_{16})_{\bullet,1} \\ (\mathcal{B}_1)_{\bullet,2} & \cdots & (\mathcal{B}_{16})_{\bullet,2} \\ (\mathcal{B}_1)_{\bullet,3} & \cdots & (\mathcal{B}_{16})_{\bullet,3} \\ (\mathcal{B}_1)_{\bullet,4} & \cdots & (\mathcal{B}_{16})_{\bullet,4} \end{bmatrix}. \tag{8.10}$$

Here, $(\mathcal{B}_i)_{\bullet,j}$ denotes the j^{th} column of the matrix of the basis operation $\mathcal{B}_i$. Sixteen basis operations are linearly independent if the matrix A is invertible. We use $\epsilon_{\max} = \max\{\|\mathcal{B}_i - \mathcal{B}_i^{(0)}\|_{\max} | i = 1, \ldots, 16\}$ as the measure of the error severity in

basis operations. When $\epsilon_{\max} < \frac{1}{32}(13 - 3\sqrt{17}) \simeq 0.0351$, it can be proved that A is always invertible. We have the explanation as following.

For two real matrices $A^{(0)}$ and A and a non-zero real vector x, we have

$$\|A^{(0)}x\|_2 = \sqrt{x^{\mathrm{T}}A^{(0)\mathrm{T}}A^{(0)}x} \geq s_{\min}(A^{(0)})\|x\|_2, \tag{8.11}$$

where $\|x\|_2$ denotes the Euclidean norm of x, and $s_{\min}(A^{(0)})$ is the minimum singular value of $A^{(0)}$. We also have

$$\|(A - A^{(0)})x\|_2 \leq \|A - A^{(0)}\|_2\|x\|_2. \tag{8.12}$$

Therefore,

$$\begin{aligned} \|Ax\|_2 &\geq \|A^{(0)}x\|_2 - \|(A - A^{(0)})x\|_2 \\ &\geq (s_{\min}(A^{(0)}) - \|A - A^{(0)}\|_2)\|x\|_2, \end{aligned} \tag{8.13}$$

where $\|A\|_2$ denotes the largest singular value of A (an operator norm). If $\|A - A^{(0)}\|_2 < s_{\min}(A^{(0)})$, $\|Ax\|_2$ is always positive, thus A is invertible.

Now, A is the matrix formed by basis operations with error as defined in Eq. (8.10), and $A^{(0)}$ is the matrix formed by basis operations without error. Because $\det(A^{(0)})$ can be numerically calculated as $\det(A^{(0)}) = 16$, $A^{(0)}$ is invertible, i.e. basis operations without error are linearly independent. The minimum singular value is $s_{\min}(A^{(0)}) = \frac{1}{2}(13 - 3\sqrt{17})$. Because $\|A - A^{(0)}\|_2 \leq 16\|A - A^{(0)}\|_{\max} = 16\epsilon_{\max}$, the matrix A is invertible if $\epsilon_{\max} < \frac{1}{16}s_{\min}(A)$.

We remark that even if $\epsilon_{\max}$ exceeds the threshold, basis operations still tend to be linearly independent.

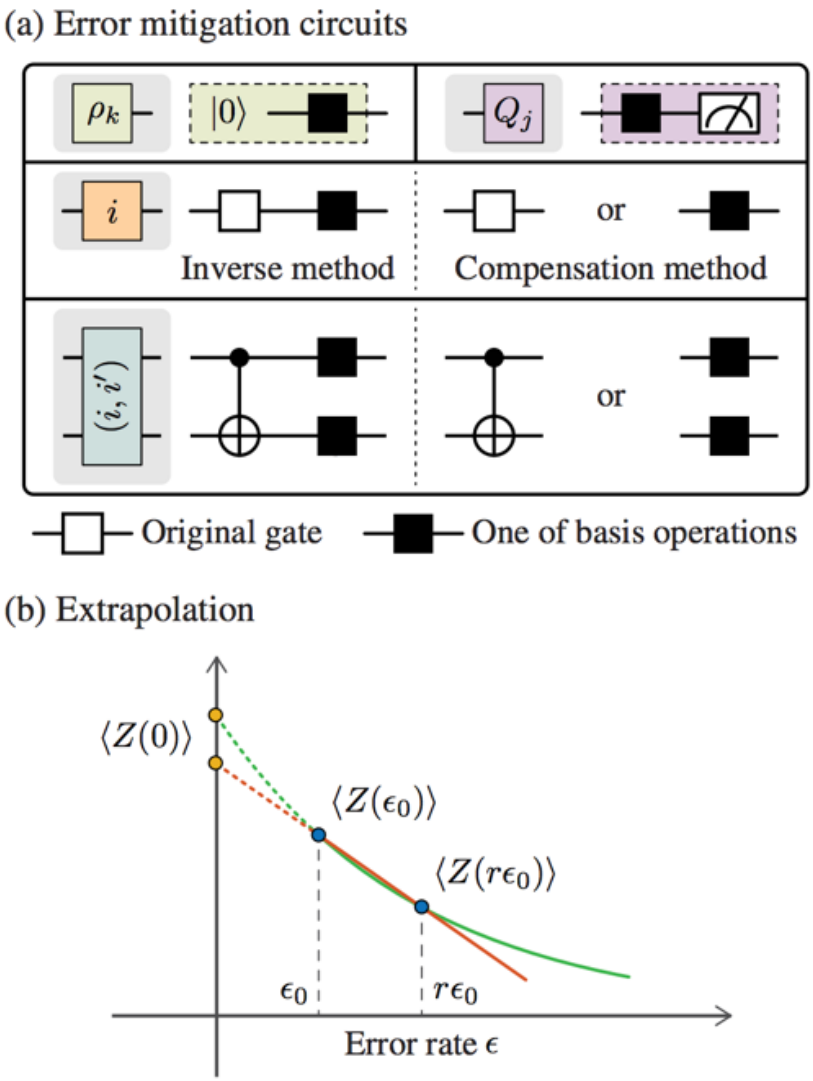


Figure 8.2: (a) Error-mitigation circuits. The choice of a basis operation is determined by the corresponding random number i, j or k. Original gate that is identity (memory operation) also has to be error-mitigated, unless memory error is negligible. In the compensation method, ether the original gate or basis operations are applied depending on the random number. (b) The schematic of the linear extrapolation (orange curve) and exponential extrapolation (green curve).

8.8 Error mitigation using basis operations

Given an operation with error $\mathcal{O}$, we can use sixteen basis operations to correct the error, i.e. realise the operation without error $\mathcal{O}^{(0)}$. There are two ways for correcting the error.

Compensation method. The operation $\mathcal{O}$ is close to $\mathcal{O}^{(0)}$. Therefore, we can keep the correct component of $\mathcal{O}$ and only decompose the error component using basis operations. We decompose the operation without error as $\mathcal{O}^{(0)} = \lambda\mathcal{O} + \sum_i q_i \mathcal{B}_i$, where λ is an arbitrary real number. If basis operations are linearly independent, the decomposition always exists, and there is only one solution of coefficients $\{q_i\}$ when λ is determined.

Inverse method. If the matrix $\mathcal{O}^{(0)}$ is invertible, we can express $\mathcal{O}$ as $\mathcal{O}^{(0)}$ followed by a noise process, i.e. $\mathcal{O} = \mathcal{N}\mathcal{O}^{(0)}$, where the noise operation $\mathcal{N} = \mathcal{O}\mathcal{O}^{(0)-1}$. In order to correct the error, we can decompose the inverse of the noise as $\mathcal{N}^{-1} = \mathcal{O}^{(0)}\mathcal{O}^{-1} = \sum_i q_i \mathcal{B}_i$. By applying the inverse of the noise after the operation $\mathcal{O}$, we can realise the operation without error, i.e. $\mathcal{O}^{(0)} = \mathcal{N}^{-1}\mathcal{O} = \sum_i q_i \mathcal{B}_i \mathcal{O}$. Similar to the compensation method, if basis operations are linearly independent, the decomposition always exists, and there is only one solution of coefficients $\{q_i\}$. However, the inverse method can only be applied if the matrix $\mathcal{O}$ is invertible.

For multi-qubit operations, the decomposition is performed using tensor products of basis operations. We consider the n-qubit operation $\mathcal{E}$. For each qubit, there is a set of basis operations $\{\mathcal{B}_{m,i} | i = 1, \ldots, 16\}$, where $m = 1, \ldots, n$ is the label of the qubit. For each set of basis operations, there is a matrix A as defined in Eq. (8.10). We use A_m to denote the matrix of the m^{th} qubit.

The operation $\mathcal{E}$ can be decomposed as

$$\mathcal{E} = \sum_{i_1=1}^{16} \cdots \sum_{i_n=1}^{16} q_{i_1,\ldots,i_n} \mathcal{B}_{1,i_1} \otimes \cdots \otimes \mathcal{B}_{n,i_n}. \tag{8.14}$$

Coefficients form a 16^n-dimensional vector

$$q = \begin{pmatrix} q_{1,1,\cdots,1,1} \\ \vdots \\ q_{1,1,\cdots,1,16} \\ \vdots \\ q_{16,16,\cdots,16,1} \\ \vdots \\ q_{16,16,\cdots,16,16} \end{pmatrix}, \tag{8.15}$$

Therefore, the decomposition is given by $q = (A_1 \otimes \cdots \otimes A_n)^{-1} E$, where E is a 16^n-dimensional vector corresponding to $\mathcal{E}$.

We choose the order of Pauli operators, i.e. the order of bases of Pauli transfer matrices $\{\mathcal{B}_{m,i} | i = 1, \ldots, 16\}$, as I, σ^{x}, σ^{y} and σ^{z} (which are also denoted as I, X, Y and Z, respectively). Then, to be consistent with $A_1 \otimes \cdots \otimes A_n$, we have

$$E = \begin{pmatrix} \mathcal{E}_{I_1 I_2 \cdots I_{n-1} I_n, I_1 I_2 \cdots I_{n-1} I_n} \\ \mathcal{E}_{I_1 I_2 \cdots I_{n-1} I_n, I_1 I_2 \cdots I_{n-1} X_n} \\ \vdots \\ \mathcal{E}_{I_1 I_2 \cdots I_{n-1} Z_n, I_1 I_2 \cdots I_{n-1} Y_n} \\ \mathcal{E}_{I_1 I_2 \cdots I_{n-1} Z_n, I_1 I_2 \cdots I_{n-1} Z_n} \\ \vdots \\ \mathcal{E}_{Z_1 Z_2 \cdots Z_{n-1} I_n, Z_1 Z_2 \cdots Z_{n-1} I_n} \\ \mathcal{E}_{Z_1 Z_2 \cdots Z_{n-1} I_n, Z_1 Z_2 \cdots Z_{n-1} X_n} \\ \vdots \\ \mathcal{E}_{Z_1 Z_2 \cdots Z_{n-1} Z_n, Z_1 Z_2 \cdots Z_{n-1} Y_n} \\ \mathcal{E}_{Z_1 Z_2 \cdots Z_{n-1} Z_n, Z_1 Z_2 \cdots Z_{n-1} Z_n} \end{pmatrix}. \tag{8.16}$$

Here, α_m ($\alpha = I, X, Y, Z$) is a Pauli operator of the m^{th} qubit.

Although basis operations are not entangling, we can use basis operations to

efficiently mitigate multi-qubit errors and errors that can entangle qubits. The example is decomposition of the controlled-NOT gate only using basis operations discussed in the previous section, which suffices to imply that any error in the form of the controlled-NOT gate can be mitigated using basis operations.

Initialisation and measurement errors can also be corrected using basis operations. Taking first the case of initialisation errors: If $|\rho\rangle\rangle$ is the error-burdened initial state, and it is a non-zero vector, we can always find a transformation $\mathcal{T}$ that satisfies $|\rho^{(0)}\rangle\rangle = \mathcal{T}|\rho\rangle\rangle$ where $|\rho^{(0)}\rangle\rangle$ is the error-free initial state. Thus by decomposing $\mathcal{T}$ using basis operations and applying it after the initialisation, we can prepare the noise free initial state. Actually, given an initial state that is close to $|0\rangle$, we can generate a complete set of linearly independent vectors $\{|\rho_k\rangle\rangle\}$ using basis operations. With these vectors, we can decompose the initial state without error as $|\rho^{(0)}\rangle\rangle = \sum_k q_k|\rho_k\rangle\rangle$. The state of a qubit is represented by a 4-dimensional real vector. To decompose the initial state of a qubit without error $|\rho^{(0)}\rangle\rangle$, we need four linearly independent initial states. If the qubit can be initialised in the state $|0\rangle$, we can choose the set of four states as $\{\rho_k^{(0)}\} = \{|0\rangle, |1\rangle, \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \frac{1}{\sqrt{2}}(|0\rangle + i\,|1\rangle)\}$. These four states can be obtained by applying basis-adjusting operations (Clifford gates) $\{[I], [R_{\mathrm{x}}], [R_{\mathrm{x}}]^2, [R_{\mathrm{z}}][R_{\mathrm{x}}]\}$ on the initial state $|0\rangle$. Because of the error in the state $|0\rangle$ and errors in basis-adjusting operations, the prepared four states $\{\rho_k\}$ are not exactly states $\{\rho_k^{(0)}\}$. When the overall error is small, states $\{\rho_k\}$ are still linearly independent.

We introduce the matrix $M^{\mathrm{in}}_{\sigma,k} = \langle\langle\sigma|\rho_k\rangle\rangle$, and $M^{\mathrm{in}(0)}$ is the matrix corresponding to $\{\rho_k^{(0)}\}$, i.e. M^{in} matrix free from state preparation noise. States $\{\rho_k\}$ are linearly independent if M^{in} is invertible. Similar to the analyse of the linear independence of basis operations (i.e. the invertibility of the matrix A), we have that M^{in} is always invertible if $\|M^{\mathrm{in}} - M^{\mathrm{in}(0)}\|_{\max} < \frac{1}{4}s_{\min}(M^{\mathrm{in}(0)}) = \frac{1}{8}\sqrt{\frac{5-\sqrt{17}}{2}} \simeq 0.0828$. The initial state without error is decomposed as $|\rho^{(0)}\rangle\rangle = \sum_{k=1}^{4} q_k|\rho_k\rangle\rangle$. Coefficients form a 4-dimensional column vector $q = [q_1\ q_2\ q_3\ q_4]^{\mathrm{T}}$. The decomposition is given by

$q = M^{\text{in}-1}|\rho^{(0)}\rangle\rangle$.

A similar argument can be applied to the corresponding result for measurement: For an observable $\langle\langle Q|$ there will be some $\langle\langle Q^{(0)}| = \langle\langle Q|\mathcal{T}$ where $\langle\langle Q^{(0)}|$ is the error-free quantity. If an observable is close to σ^{z} then a linearly independent set $\{\langle\langle Q_j|\}$ can be generated; then the error-free observable $\langle\langle Q^{(0)}| = \sum_j q_j \langle\langle Q_j|$. To decompose the observable of a qubit without error $\langle\langle Q^{(0)}|$, we need four linearly independent observables. If σ^{z} can be measured, we can choose the set of four observables as Pauli operators $\{Q_j^{(0)}\} = \{I, \sigma^{\text{x}}, \sigma^{\text{y}}, \sigma^{\text{z}}\}$. The operator I denotes a trivial measurement, whose outcome is always $+1$. Measurements of other three Pauli operators can be obtained by applying basis-adjusting operations (Clifford gates) $\{[I], [R_{\text{x}}], [R_{\text{z}}]^3[R_{\text{x}}][R_{\text{z}}]\}$ before the measurement of σ^{z}. Because of the error in the measurement of σ^{z} and errors in basis-adjusting operations, the measured observables $\{Q_j\}$ are not exactly $\{Q_j^{(0)}\}$. When the overall error is small, observables $\{Q_j\}$ are still linearly independent. We introduce the matrix $M^{\text{out}}_{j,\sigma} = \langle\langle Q_j|\sigma\rangle\rangle$, and $M^{\text{out}(0)}$ is the matrix corresponding to $\{Q_j^{(0)}\}$. Observables $\{Q_j\}$ are linearly independent if M^{out} is invertible. We have that M^{out} is always invertible if $\|M^{\text{out}} - M^{\text{out}(0)}\|_{\max} < \frac{1}{4} s_{\min}(M^{\text{out}(0)}) = \frac{1}{4}$. The initial state without error is decomposed as $\langle\langle Q^{(0)}| = \sum_{j=1}^4 q_j \langle\langle Q_j|$. Coefficients form a 4-dimensional row vector $q = [q_1\ q_2\ q_3\ q_4]$. The decomposition is given by $q = \langle\langle Q^{(0)}|M^{\text{out}-1}$.

Circuits for QEM are shown in Fig. 8.2(a). Given quasi-probabilities, we can compute the corresponding probability in sampling circuits as shown in Sec. 8.5.

Using the quasi-probability method, we can also increase the error in an operation, as required by the alternative error extrapolation method for QEM. Instead of decomposing the error-free operation $\mathcal{O}^{(0)}$ using $\mathcal{O}$ and basis operations, we can also decompose the error-boosted operation $\mathcal{O}_{\text{b}}(r) = (1-r)\mathcal{O}^{(0)} + r\mathcal{O}$ $(r > 1)$ using $\mathcal{O}$ and basis operations. It is similar for initial states and observables. We have noted that in the decomposition of an error-free operation, there are always some negative quasi-probabilities, i.e. the C factor is greater than 1, which leads to greater time

costs. But fortunately when we merely wish to decompose an error-boosted operation we can implement it without introducing negative quasi-probability, e.g. by boosting Pauli errors using Pauli gates [24]. Therefore, the sampling cost does not increase.

8.9 Quantum gate set tomography

We can measure a set of initial states $\{|\bar{\rho}_k\rangle\rangle\}$, observables $\{\langle\langle\bar{Q}_j|\}$ and operations $\{\bar{\mathcal{O}}_i\}$ (including basis operations) with gate set tomography (GST) [139, 140]. These vectors and matrices with the bar notation describe the actual physical system. Because there are errors in both initial states and observables, and initialisation and measurement errors cannot be distinguished, we may not obtain exactly these vectors and matrices describing the actual physical system. Instead, the vectors and matrices obtained using GST are $\{|\hat{\rho}_k\rangle\rangle\}$, $\{\langle\langle\hat{Q}_j|\}$ and $\{\hat{\mathcal{O}}_i\}$, which are estimations of $\{|\bar{\rho}_k\rangle\rangle\}$, $\{\langle\langle\bar{Q}_j|\}$ and $\{\bar{\mathcal{O}}_i\}$, respectively.

If we know $\{|\bar{\rho}_k\rangle\rangle\}$, $\{\langle\langle\bar{Q}_j|\}$ and $\{\bar{\mathcal{O}}_i\}$ because the physical system is well understood, we can directly use them in QEM. If our knowledge about the physical system is not enough, we can use GST to obtain $\{|\hat{\rho}_k\rangle\rangle\}$, $\{\langle\langle\hat{Q}_j|\}$ and $\{\hat{\mathcal{O}}_i\}$. We will show that, although the estimations may not be exact, we can exactly correct errors for the expected value of an observable of interest by using these estimations in QEM.

To measure a set of operations $\{\bar{\mathcal{O}}_1, \ldots, \bar{\mathcal{O}}_N\}$ on n qubits using GST, we need to choose a set of 4^n linearly independent initial states $\{\bar{\rho}_k\}$ and a set of 4^n linearly independent observables $\{\bar{Q}_j\}$. Given these initial states and observables, we measure expected values

$$\tilde{\mathcal{O}}_{j,k} = \langle\langle\bar{Q}_j|\bar{\mathcal{O}}|\bar{\rho}_k\rangle\rangle. \tag{8.17}$$

Here, $\bar{\mathcal{O}}$ is one of operations $\{\bar{\mathcal{O}}_1, \ldots, \bar{\mathcal{O}}_N\}$.

The matrix $\tilde{\mathcal{O}}$ is equivalent to $\bar{\mathcal{O}}$ up to a transformation. Because $\bar{\mathcal{O}}_{\sigma,\tau} = \langle\langle\sigma|\bar{\mathcal{O}}|\tau\rangle\rangle$ and $\sum_\sigma |\sigma\rangle\rangle\langle\langle\sigma| = I$ (the sum is taken over all Pauli operators), we have

$$\tilde{\mathcal{O}} = \bar{M}^{\mathrm{out}}\bar{\mathcal{O}}\bar{M}^{\mathrm{in}}, \tag{8.18}$$

where $\bar{M}^{\mathrm{in}}$ and $\bar{M}^{\mathrm{out}}$ are matrices defined as $\bar{M}^{\mathrm{in}}_{\sigma,k} = \langle\langle\sigma|\bar{\rho}_k\rangle\rangle$ and $\bar{M}^{\mathrm{out}}_{j,\sigma} = \langle\langle\bar{Q}_j|\sigma\rangle\rangle$. We remark that the effects of initialisation error and measurement error are included in $\bar{M}^{\mathrm{in}}$ and $\bar{M}^{\mathrm{out}}$, respectively. We cannot measure matrices $\bar{M}^{\mathrm{in}}$ and $\bar{M}^{\mathrm{out}}$ independently, therefore we cannot determine $\bar{\mathcal{O}}$ using GST. By taking the identity operation for $\bar{\mathcal{O}}$ (i.e. $\bar{\mathcal{O}} = I$) in Eq. (8.17), we can measure

$$g = \bar{M}^{\mathrm{out}}\bar{M}^{\mathrm{in}}. \tag{8.19}$$

The estimation of $\bar{\mathcal{O}}$ is given by

$$\hat{\mathcal{O}} = Tg^{-1}\tilde{\mathcal{O}}T^{-1} = T\bar{M}^{\mathrm{in}-1}\mathcal{O}\bar{M}^{\mathrm{in}}T^{-1}, \tag{8.20}$$

where g and $\tilde{\mathcal{O}}$ are obtained by measuring the expected values of observables, and T is an arbitrary invertible matrix. If $\bar{M}^{\mathrm{in}}$ and T are different, $\hat{\mathcal{O}}$ is different from $\bar{\mathcal{O}}$, but we set T so that it is similar to $\bar{M}^{\mathrm{in}}$. We usually set $T = M^{\mathrm{in}(0)}$, where $M^{\mathrm{in}(0)}$ is M^{in} matrix without state preparation noise. The estimations of states $|\bar{\rho}_k\rangle\rangle$ and observables $\langle\langle\bar{Q}_j|$ are given by

$$|\hat{\rho}_k\rangle\rangle = T_{\bullet,k} = T\bar{M}^{\mathrm{in}-1}|\bar{\rho}_k\rangle\rangle, \tag{8.21}$$

$$\langle\langle\hat{Q}_j| = (gT^{-1})_{j,\bullet} = \langle\langle\bar{Q}_j|\bar{M}^{\mathrm{in}}T^{-1}. \tag{8.22}$$

Here, $M_{\bullet,k}$ ($M_{j,\bullet}$) denotes the k^{th} column (j^{th} row) of the matrix M.

We introduce matrices $\hat{M}^{\mathrm{in}}$ and $\hat{M}^{\mathrm{out}}$ defined as $\hat{M}^{\mathrm{in}}_{\sigma,k} = \langle\langle\sigma|\hat{\rho}_k\rangle\rangle$ and $\hat{M}^{\mathrm{out}}_{j,\sigma} = \langle\langle\hat{Q}_j|\sigma\rangle\rangle$, respectively. Then $\hat{M}^{\mathrm{in}} = T$ and $\hat{M}^{\mathrm{out}} = gT^{-1}$.

For a sequence of operations $\bar{\mathcal{O}}_1, \ldots, \bar{\mathcal{O}}_N$, because $\hat{\mathcal{O}}_i$ and $\bar{\mathcal{O}}_i$ are similar matrices up to the same transformation independent of the operation (i.e. the index i), we have

$$\langle\langle \bar{Q}_j | \bar{\mathcal{O}}_N \cdots \bar{\mathcal{O}}_1 | \bar{\rho}_k \rangle\rangle = \langle\langle \hat{Q}_j | \hat{\mathcal{O}}_N \cdots \hat{\mathcal{O}}_1 | \hat{\rho}_k \rangle\rangle. \tag{8.23}$$

Therefore, although estimations $\{|\hat{\rho}_k\rangle\rangle, \langle\langle \hat{Q}_j|, \hat{\mathcal{O}}_i\}$ are in general different from their correspondences $\{|\bar{\rho}_k\rangle\rangle, \langle\langle \bar{Q}_j|, \bar{\mathcal{O}}_i\}$, they can always provide the correct prediction for the expected value of an observable in an initial state going through a sequence of operations.

Using GST estimations in QEM, the actual operations realised in this way differ from operations without error, but the computing result is correct. To correctly obtain $\langle\langle Q^{(0)} | \mathcal{O}^{(0)} | \rho^{(0)} \rangle\rangle$, we decompose the initial state, the observable and the operation using $\{|\hat{\rho}_k\rangle\rangle\}$, $\{\langle\langle \hat{Q}_j|\}$ and $\{\hat{\mathcal{O}}_i\}$ respectively. Here, $|\rho^{(0)}\rangle\rangle$, $\langle\langle Q^{(0)}|$, $\mathcal{O}^{(0)}$ and GST estimations are all known to us. The decompositions are $|\rho^{(0)}\rangle\rangle = \sum_k q_k |\hat{\rho}_k\rangle\rangle$, $\langle\langle Q^{(0)}| = \sum_j q_j \langle\langle \hat{Q}_j|$ and $\mathcal{O}^{(0)} = \sum_i q_i \hat{\mathcal{O}}_i$. Accordingly, we actually realise $|\bar{\rho}^{(0)}\rangle\rangle = \sum_k q_k |\bar{\rho}_k\rangle\rangle$, $\langle\langle \bar{Q}^{(0)}| = \sum_j q_j \langle\langle \bar{Q}_j|$ and $\bar{\mathcal{O}}^{(0)} = \sum_i q_i \bar{\mathcal{O}}_i$ in the physical system. We have $|\bar{\rho}^{(0)}\rangle\rangle = \bar{M}^{\text{in}} T^{-1} |\rho^{(0)}\rangle\rangle$, $\langle\langle \bar{Q}^{(0)}| = \langle\langle Q^{(0)}| T \bar{M}^{\text{in}-1}$ and $\bar{\mathcal{O}}^{(0)} = \bar{M}^{\text{in}} T^{-1} \mathcal{O}^{(0)} T \bar{M}^{\text{in}-1}$. Therefore, the physical system gives the computing result $\langle\langle \bar{Q}^{(0)} | \bar{\mathcal{O}}^{(0)} | \bar{\rho}^{(0)} \rangle\rangle = \langle\langle Q^{(0)} | \mathcal{O}^{(0)} | \rho^{(0)} \rangle\rangle$, and thus we can successfully obtain desired error-free output. This operation leads to the potential increase to the number of samples required, as shown in Fig. 8.3 and discussed in the caption.

We would like to remark that, when errors in actual operations are small, errors in estimations of operations are also small. If we take a proper strategy for choosing T, and errors in initial states and observables are small, the estimation of an operation $\hat{\mathcal{O}}$ is close to the operation without error $\mathcal{O}^{(0)}$ when the actual operation $\bar{\mathcal{O}}$ is close to $\mathcal{O}^{(0)}$.

Please refer to Appendix A for the discussion of the stability of gate set tomog-

raphy.

8.10 Estimation of the cost

In general, when the error in an operation is more significant, there is a higher cost for mitigating the error (to a given level of suppression). We take $\epsilon_{\mathcal{O}} = \|\mathcal{O}-\mathcal{O}^{(0)}\|_{\max}$ as the measure of the error severity in the operation, where $\mathcal{O}$ ($\mathcal{O}^{(0)}$) is the n-qubit operation with (without) error. An upper bound of the cost for correcting error in $\mathcal{O}$ is

$$C_{\mathcal{O}} - 1 \leq \frac{16^{2n}\epsilon_{\mathcal{O}}}{[s_{\min}(A^{(0)}) - 16\epsilon_{\max}]^n}. \tag{8.24}$$

Here, $\epsilon_{\max}$ is the maximum error in all basis operations for all n qubits, and $s_{\min}(A^{(0)}) = \frac{1}{2}(13 - 3\sqrt{17}) \approx 0.315$. Similar upper bounds can be obtained for correcting errors in initial states and observables. Now, we discuss the derivation of inequality (8.24) in detail.

We consider compensation method and take $\lambda = 1$, i.e. the n-qubit operation without error is realised as $\mathcal{O}^{(0)} = \mathcal{O} + \mathcal{E}$, where $\mathcal{E}$ is decomposed using basis operations as shown in Eq. 8.14. Then the cost for correcting the error in $\mathcal{O}$ is determined by

$$C_{\mathcal{O}} = 1 + \sum_{i_1,\ldots,i_n} |q_{i_1,\ldots,i_n}|. \tag{8.25}$$

Decomposition coefficients are determined by $q = (A_1 \otimes \cdots \otimes A_n)^{-1}E$, where q and E are defined in Eqs. (8.15,8.16). Here, q and E are 16^n-dimensional vectors, and $A_1 \otimes \cdots \otimes A_n$ is a 16^n-dimensional matrix. Therefore, for each element of q,

$$|q_{i_1,\ldots,i_n}| \leq 16^n \|(A_1 \otimes \cdots \otimes A_n)^{-1}\|_{\max}\|\mathcal{E}\|_{\max}. \tag{8.26}$$

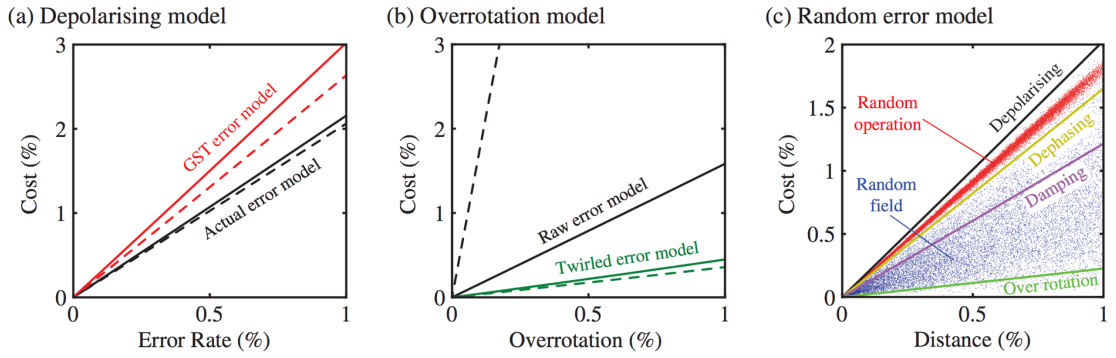


Figure 8.3: Cost $(C-1)$ for correcting errors. We consider a universal set of operations, including the initialisation, measurement, single-qubit Clifford gates, a single-qubit non-Clifford gate, and a two-qubit entangling Clifford gate. Sixteen basis operations of each qubit can be generated using these operations. Every operation in the set has error, and the memory error is also included. We assume that qualities of the initialisation and single-qubit gates are 10 times better than the measurement and two-qubit gates; also that the quality of the memory operation is 100 times better. Details of the model are given in Appendix B. The cost for correcting error in each operation in the universal set is calculated, and the maximum cost over all operations is plotted in the figure. (a) For the depolarising error model, the cost is lower if we directly use actual operations to correct errors, and the cost is higher if we use gate-set-tomography (GST) estimations to correct errors. (b) For the over-rotation model, the cost is higher without using the Pauli twirling, and the cost is lower when the Pauli twirling is used. In (a) and (b), solid curves correspond to the compensation method (with the optimised λ), and dashed curves correspond to the inverse method. (c) The cost as a function of the distance between operations with errors and operations without error. For the operation with error $\mathcal{O}$ and the operation without error $\mathcal{O}^{(0)}$, the distance is $\epsilon_{\mathcal{O}} = \|\mathcal{O} - \mathcal{O}^{(0)}\|_{\max}$. The x-axis illustrates the maximum distance over all operations in the universal set. In (b) and (c), we always use GST estimations. In (c), Pauli twirling and the inverse method are used for all the data. Pauli twirling is applied to the measurement and two-qubit gate, and the inverse method is only applied to the two-qubit gate, while errors in other operations are corrected using the compensation method. We remark that usually the maximum distance and the maximum cost are given by the two-qubit gate. These figure were produced by one of coauthors of Ref. [29], Ying Li.

Here, we have used that the maximum absolute value of an element of E is $\|\mathcal{E}\|_{\max}$. Because $\mathcal{E} = \mathcal{O}^{(0)} - \mathcal{O}$, we have $\|\mathcal{E}\|_{\max} = \epsilon_{\mathcal{O}}$. Using the inequality (41) in Appendix A, we have

$$\begin{aligned} & \|(A_1 \otimes \cdots \otimes A_n)^{-1}\|_{\max} \\ = & \prod_{l=1}^{n} \|A_l^{-1}\|_{\max} \leq \prod_{l=1}^{n} \|A_l^{-1}\|_2 \\ \leq & \prod_{l=1}^{n} \frac{1}{s_{\min}(A^{(0)}) - \|A_l - A^{(0)}\|_2} \\ \leq & \prod_{l=1}^{n} \frac{1}{s_{\min}(A^{(0)}) - 16\|A_l - A^{(0)}\|_{\max}} \\ \leq & \frac{1}{[s_{\min}(A^{(0)}) - 16\epsilon_{\max}]^n}. \end{aligned} \tag{8.27}$$

Here, $\epsilon_{\max} = \max\{\|A_l - A^{(0)}\|_{\max} | l = 1, \ldots, n\}$. There are total 16^n decomposition coefficients, therefore

$$C_{\mathcal{O}} \leq 1 + \frac{16^{2n}\epsilon_{\mathcal{O}}}{(s_{\min}(A^{(0)}) - 16\epsilon_{\max})^n}. \tag{8.28}$$

Here, we have assumed that $\epsilon_{\max} < s_{\min}(A^{(0)})/16$.

We consider using the set of initial states with errors $\{|\rho_k\rangle\rangle\}$ to realise the initial state $|\rho_{k_0}^{(0)}\rangle\rangle$, which is in the set of initial states without error $\{|\rho_k^{(0)}\rangle\rangle\}$. The initial state can be decomposed as $|\rho_{k_0}^{(0)}\rangle\rangle = |\rho_{k_0}\rangle\rangle + \sum_k (q_k - \delta_{k,k_0})|\rho_k\rangle\rangle$ (see section 8.8). We use $\epsilon_{\text{in}} = \|M^{\text{in}} - M^{\text{in}(0)}\|_{\max}$ as the measure of the error severity in initial states. Using $\||\rho_{k_0}\rangle\rangle - |\rho_{k_0}^{(0)}\rangle\rangle\|_{\max} \leq \epsilon_{\text{in}}$ and $\|M^{\text{in}-1}\|_{\max} \leq 4[s_{\min}(M^{\text{in}(0)}) - \epsilon_{\text{in}}]^{-1}$, we have the cost for correcting errors in initial states

$$\begin{aligned} C_{\text{in}} &= \sum_k |q_k| \leq 1 + \sum_k |q_k - \delta_{k,k_0}| \\ &\leq 1 + \frac{4^2\epsilon_{\text{in}}}{s_{\min}(M^{\text{in}(0)}) - \epsilon_{\text{in}}}. \end{aligned} \tag{8.29}$$

A similar argument can be applied to observables. We consider using the set of

observables with errors $\{\langle\langle Q_j|\}$ to realise the observable $\langle\langle Q_{j_0}^{(0)}|$, which is in the set of observables without error $\{\langle\langle Q_j^{(0)}|\}$. We use $\epsilon_{\text{out}} = \|M^{\text{out}} - M^{\text{out}(0)}\|_{\max}$ as the measure of the error severity in observables. Then, the cost for correcting errors in measured observables is

$$C_{\text{out}} \leq 1 + \frac{4^2\epsilon_{\text{out}}}{s_{\min}(M^{\text{out}(0)}) - \epsilon_{\text{out}}}. \tag{8.30}$$

There are also several ways to reduce the cost. The upper bound of the cost can be obtained using the compensation method and taking $\lambda = 1$. In general, we can optimise the value of λ or use the inverse method to minimise the cost. For example, for the depolarising error model (see Appendix B), the cost of using the inverse method is lower than using the compensation method [see Fig. 8.3(a)]. We remark that, to obtain data for the compensation method in Fig. 8.3, we have optimised the value of λ. If we use estimations obtained from GST to correct errors, we can optimise the T matrix to minimise the cost. In Fig. 8.3(a), we can find that, without optimising T matrices, the cost using estimations obtained from GST is higher than using actual operations. If we choose the matrix in the form $T = \otimes_{m=1}^{n} T_m$, where T_m is a 4-dimensional real matrix corresponding to the m^{th} qubit, there are $16n$ parameters to be optimised for a n-qubit quantum computer in total, which is a non-trivial task. Under some reasonable conditions, we can also use the Pauli twirling [150, 151] to reduce the cost.

Pauli twirling

In many quantum computing systems, e.g. superconducting qubits [14] and ion traps [15, 16], the fidelity of single-qubit gates is much better than the fidelity of two-qubit gates, and usually a state can be initialised with a high fidelity while the fidelity of measurement is worse. In this section, we consider the case that error rates of initialisation and single-qubit gates are much lower than error rates

of two-qubit gates and measurement.

If the error rate of initialisation is low (much lower than the error rate of measurement), we know how to choose T so that the estimation of an operation obtained from GST is close to the actual operation. We cannot exactly estimate operations using GST, as it is impossible to distinguish initialisation and measurement errors. If we treat all errors in the initialisation and measurement as measurement error, the difference between the estimation and the actual operation is only determined by the initialisation error. Therefore, if the initialisation is high-fidelity, the estimation obtained in this way and the actual operation are close.

Because the set of basis operations includes Pauli gates, it is easy to use basis operations to correct Pauli errors. By using the Pauli twirling, we can convert the error in a two-qubit entangling Clifford gate to Pauli error [150, 151], which is achieved by applying Pauli gates before and after the two-qubit gate. This treatment of the error is feasible only if the fidelity of Pauli gates is much better than the two-qubit gate, otherwise Pauli gates cause significant new errors, which may not be Pauli error, on the two-qubit gate. In Fig. 8.3(b), we can find that the cost can be significantly reduce by using the Pauli twirling for the over-rotation error model (see Appendix B).

In Fig. 8.3(c), costs of different error models are compared, including the depolarising model, pure-dephasing model, amplitude-damping model and the over-rotation model. We also randomly generated many other error models, and see Appendix B for details of these error models. For a random-operation model, we randomly generate an operation close to the ideal error-free operation, and we find that the cost is approximately the cost of the depolarising model. For a random-field model, we randomly generate a Hamiltonian that drives the erroneous evolution, and the cost is between the depolarising model and over-rotation model.

From Fig. 8.3(c) we see that the cost of quantum error mitigation varies depending on the error model but is generally upper-bounded by the case of depolarising

noise, over the range of noise levels shown here. (Note that other models can exceed the cost of the the depolarising model if we use even lower fidelity gates). For the depolarising model, the cost for mitigating error in a two-qubit entangling gate is $C - 1 \simeq a\epsilon$, where ϵ is the error rate and the factor a is between 2 and 3 [see Fig. 8.3(a)]. If errors in initialisation and single-qubit gates are negligible, or if the matrix T is optimised to minimise the cost, the factor a can approach 2. If we accept the depolarising model as an approximate upper bound, we can estimate the overall cost in a quantum algorithm. Suppose the total number of gates in a quantum algorithm is N, the overall amplification of the standard deviation (uncertainty of the computing result) is $(1 + 2\epsilon)^N$, indicating that $(1 + 2\epsilon)^{2N}$ times more repetitions of the experiment are required in order to achieve the same standard deviation to the case without error mitigation. We are interested in the case that N is large but ϵ is small, therefore, $(1 + 2\epsilon)^{2N} \sim e^{4N\epsilon}$. As a rule of thumb we might take $N\epsilon = 2$ as a limit for acceptable scenarios, since then $e^{4N\epsilon} \approx 3,000$. However, larger overhead factors may be acceptable depending on the speed of the quantum computer.

8.11 Numerical simulation

In our numerical simulation, we apply QEM to the SWAP-test circuit [94] shown in Fig. 8.4, in which we realise each controlled-SWAP gate using Toffoli gates and realise each Toffoli gate using T gates, $T^\dagger$ gates, Hadamard gates and controlled-NOT gates [31]. We note the implementation of a SWAP test using shallow circuit is possible [91, 92], as we mentioned in Chapter 6. However, for present purposes it is not essential to use an optimised realisation of the SWAP circuit; its role is simply to test whether our error mitigation method works properly and indeed the considerable depth of our non-optimal circuit is helpful here. The number of gates scales as $23N_{\mathrm{q}} - 21$, where N_{q} is the number of qubits (e.g. $N_{\mathrm{q}} = 7$ in Fig. 8.4).

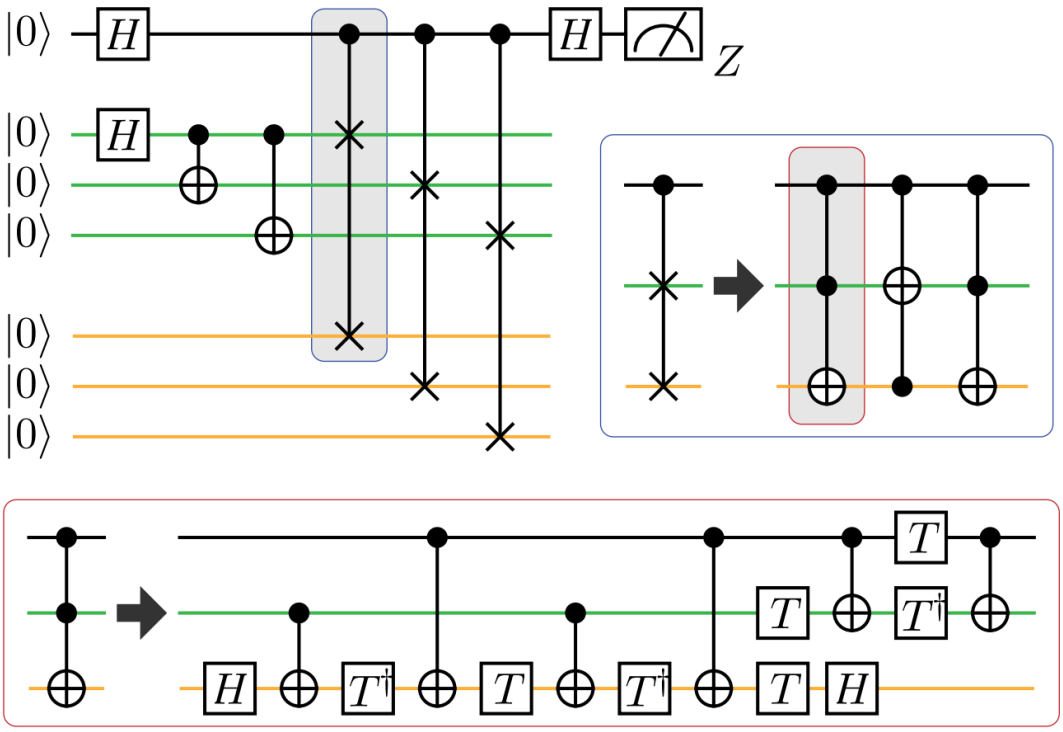


Figure 8.4: Swap-test circuit. The first qubit (denoted black) is a probe qubit, and the expected value of Z gives the overlap between states of two qubit groups (denoted green and orange, respectively). Green qubits are prepared in the GHZ state $(|00\cdots\rangle + |11\cdots\rangle)/\sqrt{2}$, and orange qubits are prepared in $|00\cdots\rangle$. Therefore, the ideal expected value of Z is 0.5.

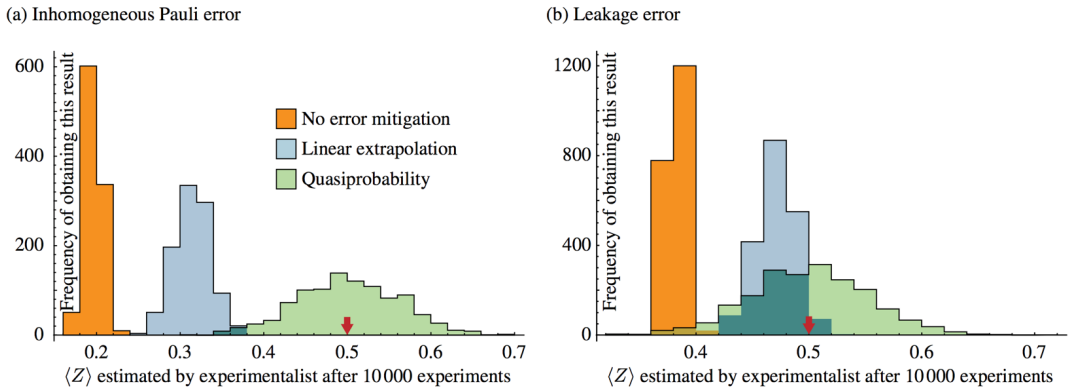


Figure 8.5: Histograms of the estimation of $\langle Z \rangle$ using quantum computers with inhomogeneous Pauli error and leakage error. For the inhomogeneous Pauli error model, the SWAP-test circuit involving 19 qubits is simulated: one qubit is the probe qubit, and each group has 9 qubits. For the leakage error model, the SWAP-test circuit involving fewer qubits (15 qubits) is simulated, because in the numerical simulation we need to use an additional qubit to introduce the leakage process. The ideal value of $\langle Z \rangle$ is marked by the red arrow.

Without error, the expected value of the observable Z (σ^{z} of the probe qubit) in the SWAP-test circuit in Fig. 8.4 is 0.5, because it evaluates the overlap of the GHZ state $(|00\cdots\rangle + |11\cdots\rangle)/\sqrt{2}$ and the state $|00\cdots\rangle$.

We consider error models according to which the same noise $\mathcal{E}$ is applied after the initialisation to the state $|0\rangle$, before the measurement, and before and after each gate. For the controlled-NOT gate, the noise applied is $\mathcal{E} \otimes \mathcal{E}$ on two qubits. We remark that basis operations are also affected by noise in the same way. We consider two types of noise: inhomogeneous Pauli error and leakage error, which can be respectively described as

$$\mathcal{E}_{\mathrm{inh}} = (1 - p_{\mathrm{x}} - p_{\mathrm{y}} - p_{\mathrm{z}})[I] + p_{\mathrm{x}}[\sigma^{\mathrm{x}}] + p_{\mathrm{y}}[\sigma^{\mathrm{y}}] + p_{\mathrm{z}}[\sigma^{\mathrm{z}}]$$

and

$$\mathcal{E}_{\mathrm{leak}} = [|0\rangle\langle 0| + \sqrt{1-p}|1\rangle\langle 1|],$$

where p_α is the probability of the error $[\sigma^\alpha]$, and p is the probability of the leakage error from the state $|1\rangle$. It is worth mentioning that the leakage error is a non-trace-

preserving error. In our simulations, we set $p_{\mathrm{x}} = p_{\mathrm{y}} = 0.0001$, $p_{\mathrm{z}} = 0.0006$, and $p = 0.0008$. Thus in both models the total error rate is 0.08% for initialisation and measurement, 0.16% for single-qubit gates and 0.32% for two-qubit gates, which can be achieved with two-qubit gates in ion traps [16] and can be far surpassed for one-qubit gates [15]. Moreover, with these numbers the expected total number of error events in circuits of the depth and breadth that we consider here is approximately unity for the case the number of simulated qubits equals to 19; this is a challenging domain for error mitigation.

In addition to quasi-probability decomposition (see section 8.14 for an instruction of the implementation), we also study the Richardson extrapolation technique introduced in Ref. [24]. The expected value of Z obtained by running the SWAP-test circuit in a quantum computer with noise depends on the error rate, i.e. it is a function that can be denoted as $\langle Z\rangle(\epsilon)$, where ϵ is the overall error rate. For our first set of numerical experiments we consider linear extrapolation to the error-free value $\langle Z\rangle(0)$ as follows: We obtain the expected value $\langle Z\rangle(\epsilon_0)$ with the lowest attainable error rate ϵ_0, and by increasing error rate to $r\epsilon_0$ with $r > 1$, we obtain another expected value $\langle Z\rangle(r\epsilon_0)$. Using these two values, we can infer $\langle Z\rangle(0) = (r\langle Z\rangle(\epsilon_0) - \langle Z\rangle(r\epsilon_0))/(r-1)$ as shown in Fig. 8.2(b), which is the final estimation of $\langle Z\rangle$. Here, we set $r = 2$.

The first set of numerical results are shown in Fig. 8.5. We assume that the experimentalist makes her overall estimate of the $\langle Z\rangle$ after she performs 10^4 individual experiments. We take this number of runs as a fixed constraint (effectively, we are constraining her overall time resource), and she may choose to employ those runs using one of three alternative approaches: no error correction, linear extrapolation, and quasi-probability decomposition (using basis operations and incorporating GST). In each experiment the SWAP-test circuit or its variant for the purpose of QEM is implemented. Because of the finite number of samples, the estimation is stochastic. Therefore, in our numerical simulation we perform the appropriate se-

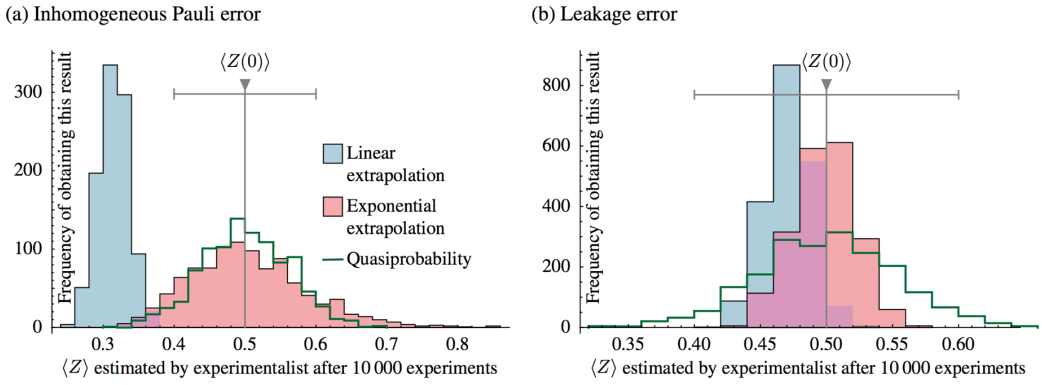


Figure 8.6: Comparison of optimised quantum error mitigation techniques: The green outlines correspond to the quasi-probability technique while solid histograms correspond to the extrapolation technique using a presumption of an underlying linear (blue) or exponential (red) dependence. For the inhomogeneous Pauli error model, the SWAP-test circuit involving 19 qubits is simulated. For the leakage error model, the SWAP-test circuit involving fewer qubits (15 qubits) is simulated. The horizontal axis is the estimate of $\langle Z \rangle$ that an experimentalist who performs 10^4 experiments will obtain. Ideally the circuit produces $\langle Z \rangle = 0.5$. (a) The left panel corresponds to physical errors of the inhomogeneous Pauli type, while the right panel (b) corresponds to physical leakage errors. Note that the horizontal scale differs between the two panels; a grey bar showing the scale from 0.4 to 0.6 appears in both figures to facilitate comparison. For either type of noise, it is clear that exponential extrapolation mitigates noise more than linear extrapolation.

ries of 10^4 experiments, mirroring the actions of the experimentalist, and then we repeat $\geq 1,000$ times in order to determine the distribution of final estimations that may be obtained. The distribution for each case is plotted in Fig. 8.5.

Now, it is clear that QEM approaches can improve the result, i.e. the corresponding distributions are shifted closer to the ideal value 0.5 compared to the approach without QEM. For the inhomogeneous Pauli error model, the means of distributions are at 0.1961, 0.3415, and 0.5011 for the no error correction, linear extrapolation, and quasi-probability decomposition, respectively. The distribution of the quasi-probability approach is centered at the ideal value, which clearly shows its desirable property of completely removing any systematic bias. However, the variance of the distribution is greater (as we expected) compared to other two approaches. A more fair metric would be the expected absolute error versus ideal value (i.e. $\overline{|\langle Z \rangle - 0.5|}$). Given an ideal error-free computer and 10^4 trials, this met-

ric would evaluate to 0.006910. Using the inhomegeneous Pauli error, with our three protocols the three corresponding values are 0.3039, 0.1853 and 0.0491. Similarly, for the leakage error model, the means for three approaches are 0.3819, 0.4710, 0.5007, while the expected absolute error evaluates to 0.1181, 0.0294, and 0.0434.

From these results it may appear that (given a large but reasonable number of samples) the quasi-probability technique outperforms the extrapolation method, with the latter unable to approach the mean of the error-free circuit. However, here the extrapolation method was limited to linear interpolation whereas the physical error rates are high enough that the linear assumption is not appropriate. One could fit a higher order polynomial using more data points (here, we have only used two: one derived from the actual lowest possible error rate and one boosted to twice the error rate.) However, the sampling cost generally scales exponentially to the number of data points [78]. Also, as we are limiting the total number of experimental runs to 10^4, this would lead to greater noise in each data point. Therefore we should not take many data points. Moreover, as we now argue, $\langle Z \rangle (\epsilon)$ is likely to be well-approximated by an exponential decay rather than a polynomial one (i.e. the expected value of the observable falls exponentially with the physical error rate) and two data points are sufficient to estimate the zero-error observable under that assumption.

In Fig. 8.6 we show the results when the experimentalist indeed assumes that the expected value $\langle Z \rangle (\epsilon)$ changes exponentially with respect to the error rate ϵ and converges to 0 in the limit of $\epsilon \to \infty$. Here, we assume

$$\langle Z \rangle (\epsilon) = \langle Z \rangle (0) e^{-a\epsilon}. \tag{8.31}$$

Therefore, we have

$$\begin{aligned} \langle Z \rangle (\epsilon_0) &= \langle Z \rangle (0) e^{-a\epsilon_0} \\ \langle Z \rangle (r\epsilon_0) &= \langle Z \rangle (0) e^{-ar\epsilon_0}. \end{aligned} \tag{8.32}$$

We now have $\langle Z\rangle (\epsilon_0)^r \langle Z\rangle (r\epsilon_0)^{-1} = \langle Z\rangle (0)^{r-1}$, therefore the experimentalist will infer the error-free value as

$$\langle Z\rangle(0) = \langle Z\rangle(\epsilon_0)^{\frac{r}{r-1}} \langle Z\rangle(r\epsilon_0)^{\frac{1}{1-r}}. \tag{8.33}$$

Here we take $r = 2$.

As shown in Fig. 8.6, the distribution of the final result using the exponential extrapolation approaches the ideal value of $\langle Z\rangle$ (which is 0.5 for the SWAP-test circuit) much better than the linear extrapolation. Given the same 10^4 experimental runs, the mean of the experimentalist's estimate is now 0.5111 for the inhomogeneous Pauli error model and 0.4986 for the leakage error model. These numbers are significantly close to those of quasi-probability technique but the variance is smaller. The expected absolute error for inhomogeneous Pauli error and leakage error are 0.06501 and 0.01882, respectively. For the latter, the expected absolute error comes within a factor of three of the shot-noise limit that would be achieved by error-free ideal hardware (0.00691). This is despite the fact that our error-burdened circuits have error rates corresponding to at least one error event per circuit. We emphasise that this suppression results purely from the QEM protocol i.e. it is achieved at no cost in terms of the qubit count or the total number of runs (constrained to 10^4).

Due to the limited power of classical computer we utilised, our exact numerical simulations did not involve more than 19 qubits. However, it is of course very interesting to investigate the relevance of our techniques to quantum computing using over 50 qubits, which is in the so called 'quantum supremacy' regime. Therefore, we estimate the cost of quantum error mitigation in the SWAP-test circuit, using the same error models in our numerical simulation and error rates achievable in ion trap experiments [15, 16], i.e. the error rate of two-qubit gate is 0.1% and error rates of single-qubit operations are 0.01%. Take for example the SWAP test with $N_{\mathrm{q}} = 51$ qubits (the number of gates is $1,152$). For the inhomogeneous Pauli error

model, the overall cost is $C = 2.956$, which implies that we can attain the same computing precision as the ideal case if we have $C^2 = 8.738$ times more repetitions of the experiment, which is experimentally feasible. For the leakage error model, the cost for the 51-qubit SWAP test is $C = 4.338$, which means $C^2 = 18.818$ times more repetitions are required. A plot showing how the cost scales depending on qubit count is shown in Fig. 8.7.

We also evaluate C^2 for a fully paralleled circuit, whose circuit depth is N_{q}, and each layer has $N_{\mathrm{q}}/2$ single qubit gates and $N_{\mathrm{q}}/4$ controlled-NOT gates, which means the quantum circuit has $N_{\mathrm{q}}^2/2$ single qubit gates and $N_{\mathrm{q}}^2/4$ controlled-NOT gates. As single qubit gates, we use T gate, S gate and Hadamard gate, because these gates plus controlled-NOT gate constitute a universal gate set, and we equally assign the number of qubits to these three types of single qubit gates. We plot C^2 versus the number of qubits for the SWAP-test circuit and the fully paralleled circuit in Fig. 8.7. We observe that for the SWAP test circuit, it is feasible to venture into the 'supremacy' regime with the current achievable fidelities; for the more demanding case of a fully paralleled circuit (so that the gate count scales as N_{q}^2) we see that today's error rates would not be sufficient much beyond 50 qubits, but that error rates ten times lower would easily suffice for 80 qubits and beyond.

8.12 Intuition for Exponential extrapolation

Intuitively, the explanation for the success of the exponential extrapolation is as follows. We express the i^{th} noise event occurring in the quantum circuit as

$$\mathcal{E}_i(\epsilon) = (1-\epsilon)[I] + \epsilon\mathcal{E}'(\epsilon)_i, \tag{8.34}$$

where $\mathcal{E}'(\epsilon)$ is the error component. The only assumption is that the error component only weakly depends on the error rate ϵ (see Appendix C). Now, for simplification we ignore the computing operations, which do not affect our general argument.

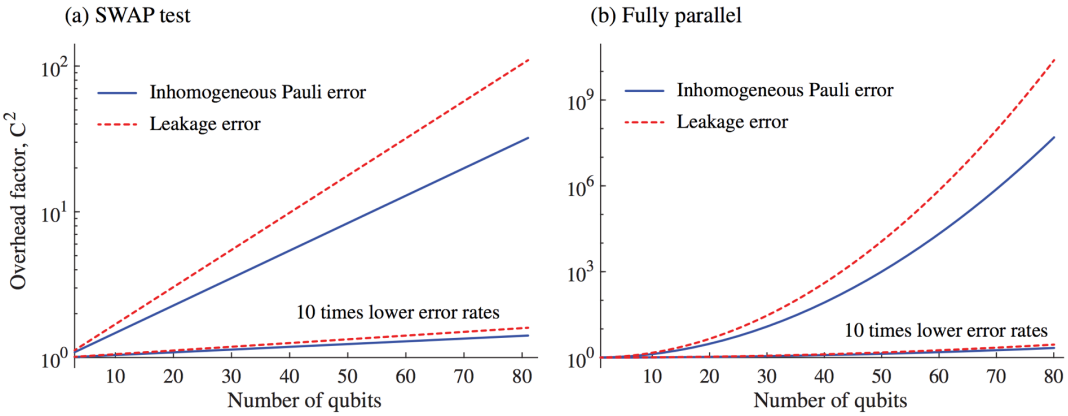


Figure 8.7: The graphs show the cost of matching the performance of an ideal noiseless circuit with a noisy circuit, using the quasi-probability method. The vertical axis (C^2) is a multiplicative factor indicating how many more repetitions of the circuit execution are requited. In each graph the upper pair of lines correspond to error rates achievable in ion trap experiments [15, 16], i.e. the error rate of two-qubit gate is 0.1% and error rates of single-qubit operations are 0.01%. The lower pair of lines indicate the result of reducing these error rates by a factor of ten. The left panel corresponds to the SWAP-test circuit. The right panel corresponds to a circuit where every qubit is actively gated in every time step, and the number of steps equals to the number of qubits.

The total noise that the entire quantum circuit experiences is

$$\prod_{i=1}^{N} \mathcal{E}_i = \prod_{i=1}^{N} [(1-\epsilon)[I] + \epsilon \mathcal{E}_i']. \tag{8.35}$$

Here, N is the total number of the noisy operations. Expanding the overall noise, we get

$$\prod_{i=1}^{N} \mathcal{E}_i = \sum_{n=1}^{N} \binom{N}{n} (1-\epsilon)^{N-n} \epsilon^n \mathcal{X}_n, \tag{8.36}$$

where

$$\mathcal{X}_n = \binom{N}{n}^{-1} \times \begin{pmatrix} \text{the sum of terms where} \\ \mathcal{E}' \text{ appears for } n \text{ times} \end{pmatrix}. \tag{8.37}$$

Note that the coefficient of $\mathcal{X}_n$ in the overall noise corresponds to a binomial distribution, which can be approximated by the Poisson distribution. We have

$$\prod_{i=1}^{N} \mathcal{E}_i = e^{-N\epsilon} \sum_{n=0}^{N} \frac{(N\epsilon)^n}{n!} \mathcal{X}_n, \tag{8.38}$$

We can find that the impact of the overall noise on the expected value of some observable is proportional to $e^{-N\epsilon}$, which implies that exponential extrapolation works better than linear extrapolation.

8.13 Cost for exponential extrapolation

Now, we discuss the cost for exponential extrapolation. We assume that $\langle Z \rangle (\epsilon) \propto e^{-N\epsilon}$, and $\mathrm{Var}(\langle Z \rangle (\epsilon)) = \mathrm{Var}(\langle Z \rangle (r\epsilon))$, and the same number of samples are assigned to two points used for extrapolation. Now, for a function $f(x, y)$, where x and y are random variables, the variance of f can be evaluated by using the formula for propagation of uncertainty

$$\mathrm{Var}[f] = \left|\frac{\partial f}{\partial x}\right|^2 \mathrm{Var}[x] + \left|\frac{\partial f}{\partial y}\right|^2 \mathrm{Var}[y]. \tag{8.39}$$

Assuming $f(x, y) = x^a y^b$, we have

$$\mathrm{Var}[f] = (a x^{a-1} y^b)^2 \mathrm{Var}[x] + (b x^a y^{b-1})^2 \mathrm{Var}[y]. \tag{8.40}$$

Now, by substituting $a = r/r - 1$, $b = 1/(1 - r)$, $x = \langle Z \rangle (\epsilon_0)$ and $y = \langle Z \rangle (r\epsilon_0)$, we have $f = \langle Z \rangle^* (0)$, where $\langle Z \rangle^* (0)$ is the estimated error free expectation value of Z. Then we obtain

$$\begin{aligned} \mathrm{Var}[\langle Z \rangle^* (0)] = &\left(\frac{r}{r-1} \langle Z \rangle (\epsilon_0)^{\frac{1}{r-1}} \langle Z \rangle (r\epsilon_0)^{\frac{1}{1-r}}\right)^2 \mathrm{Var}[\langle Z \rangle (\epsilon_0)] \\ &+ \left(\frac{1}{1-r} \langle Z \rangle (\epsilon_0)^{\frac{r}{r-1}} \langle Z \rangle (r\epsilon_0)^{\frac{r}{1-r}}\right)^2 \mathrm{Var}[\langle Z \rangle (r\epsilon_0)]. \end{aligned} \tag{8.41}$$

Then, by using $\langle Z\rangle(\epsilon) \propto e^{-N\epsilon}$, we have

$$\begin{aligned} \langle Z\rangle(\epsilon_0)^{\frac{1}{r-1}} \langle Z\rangle(r\epsilon_0)^{\frac{1}{1-r}} &= e^{N\epsilon_0} \\ \langle Z\rangle(\epsilon_0)^{\frac{r}{r-1}} \langle Z\rangle(r\epsilon_0)^{\frac{r}{1-r}} &= e^{Nr\epsilon_0}, \end{aligned} \tag{8.42}$$

and assuming $\mathrm{Var}[\langle Z\rangle(\epsilon_0)] = \mathrm{Var}[\langle Z\rangle(r\epsilon_0)]$, we can get the variance of the estimation

$$\mathrm{Var}[\langle Z\rangle^*(0)] = \frac{(r^2 e^{2N\epsilon_0} + e^{2Nr\epsilon_0})}{(r-1)^2}\mathrm{Var}[\langle Z\rangle(\epsilon_0)]. \tag{8.43}$$

Thus the cost for the exponential extrapolation is

$$C^2_{\mathrm{exp}} = \frac{(r^2 e^{2N\epsilon_0} + e^{2Nr\epsilon_0})}{(r-1)^2}, \tag{8.44}$$

which implies we have to perform C^2_{exp} times more measurements. Due to the exponential dependence of r and $N\epsilon_0$, the sampling cost of exponential extrapolation can be much larger than the one of linear extrapolation for large r and $N\epsilon_0$. However, error mitigation generally works in the regime that $N\epsilon_0 \leq 1$, and therefore the cost of linear and exponential extrapolations are similar when $r \approx 1$. It is also possible to have optimal r to minimise C^2_{exp}. For example, for $N\epsilon_0 = 1$, $r \approx 2.2$. However, it is worth mentioning that the optimisation of r does not necessarily indicate $\langle Z\rangle^*(0)$ for that r is the best approximation of the ideal expectation value, as it is merely the optimisation of C^2_{exp}.

8.14 Instruction of the implementation of the quasi-probability method

This section is a self-contained description of how to implement QEM using the quasi-probability decomposition. There are three steps: first, implement GST;

second, compute the quasi-probability decomposition; third, implement the quasi-probability decomposition using the Monte Carlo approach.

8.14.1 Implementation of gate set tomography

General discussions of GST are given in section 8.9, therefore, here we describe GST in a more concrete way so that experimentalists can refer to it to implement actual experiments. GST is implemented to measure all gates used in the quantum computation. We discuss how to measure single qubit gates first and two-qubit gates afterwards.

To measure a single-qubit gate by using GST, we prepare initial states $|0\rangle$, $|1\rangle$, $|+\rangle$, and $|y+\rangle$, where $|+\rangle$, and $|y+\rangle$ are the eigenstates of Pauli operators σ^{x} and σ^{y} with the eigenvalue $+1$, respectively. We denote these states $\bar{\rho}_1$, $\bar{\rho}_2$, $\bar{\rho}_3$ and $\bar{\rho}_4$, respectively. These initial states can be noisy states, which is the essential advantage of GST, i.e. GST can tolerate state preparation and measurement errors. Then, we apply the gate that we want to measure, e.g., Hadamard gate, T gate and $T^\dagger$ gate in the SWAP-test circuit. Here we will use $\bar{\mathcal{O}}$ (superoperator acting on a reduced density matrix) to denote the gate to be measured, which is generally noisy. Subsequently, we measure expectation values for four observables, I, σ^{x}, σ^{y} and σ^{z}, respectively. Here I is a trivial observable whose measurement outcome is always $+1$. We denote these observables as $\bar{Q}_1$, $\bar{Q}_2$, $\bar{Q}_3$, $\bar{Q}_4$, and measurements of these observables can also be noisy. Again, we stress this is the advantage of GST. Then, by repeating the experiment to compute the mean value of observables, we can construct the 4×4 matrix $\tilde{\mathcal{O}}$, and matrix elements are

$$\tilde{\mathcal{O}}_{j,k} = \mathrm{Tr}\left(\bar{Q}_j \bar{\mathcal{O}} \bar{\rho}_k \bar{\mathcal{O}}^\dagger\right). \tag{8.45}$$

Similarly, we can obtain the 4×4 matrix g by choosing not to apply any gate (i.e.

$\bar{\mathcal{O}} = I$) to initial state. Then the matrix elements are

$$g_{j,k} = \mathrm{Tr}\left(\bar{Q}_j \bar{\rho}_k\right). \tag{8.46}$$

This process is implemented for each qubit and each type of single-qubit gate (including all basis operations).

One will find that there is a freedom in the specification of the gate $\bar{\mathcal{O}}$. Legitimate variants can be obtained as

$$\hat{\mathcal{O}} = T g^{-1} \tilde{\mathcal{O}} T^{-1}, \tag{8.47}$$

where T is an invertible 4×4 matrix. The matrix T can be different for different qubits but must be the same for all gates on the same qubit. We can choose T to minimise the cost in QEM. In the case that the error rate of preparing initial states is low, we can take

$$T = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 \end{pmatrix}, \tag{8.48}$$

which will approximately minimise the cost according to our experience. This corresponds to $\bar{M}^{\mathrm{in}}$ matrix discussed in section 8.9 in the absence of noise, i.e. $\bar{M}^{\mathrm{in}(0)}$.

Estimations of the initial state $\bar{\rho}_k$ and the observable $\bar{Q}_j$ are respectively

$$|\hat{\rho}_k\rangle\rangle = T_{\bullet,k}, \tag{8.49}$$

$$\langle\langle \hat{Q}_j| = (g T^{-1})_{j,\bullet}. \tag{8.50}$$

Here, $M_{\bullet,k}$ ($M_{j,\bullet}$) denotes the k^{th} column (j^{th} row) of the matrix M.

To measure a two-qubit gate by using GST, the procedure is basically the same.

The only difference is that to have GST we should prepare 16 initial states and measure 16 observables. Initial states are the tensor products of single-qubit initial states, i.e. $\bar{\rho}_{k_1}^{(1)} \otimes \bar{\rho}_{k_2}^{(2)}$, and observables are tensor products of single-qubit observables, i.e. $\bar{Q}_{j_1}^{(1)} \otimes \bar{Q}_{j_2}^{(2)}$. Here, the superscript is the label of the qubit. Accordingly, the matrix $g = g^{(1)} \otimes g^{(2)}$, which is the tensor product of g matrices of two qubits, and similarly the matrix $T = T^{(1)} \otimes T^{(2)}$, which is the tensor product of T matrices of two qubits.

8.14.2 Quasi-probability decomposition

Using results obtained from GST, we can compute the quasi-probability decomposition. From GST we obtain estimations of initial states, observables to be measured, gates (including basis operations), and they are

$$
\begin{aligned}
|\hat{\rho}_k\rangle\rangle &\quad : \quad \text{Initial state} \\
\langle\langle \hat{Q}_j| &\quad : \quad \text{Observable} \\
\hat{\mathcal{O}} &\quad : \quad \text{Gate} \\
\hat{\mathcal{B}}_i &\quad : \quad \text{Basis operation}
\end{aligned}
$$

These estimations are utilised to compute the quasi-probability decomposition.

Now, we focus on the inverse method , because the cost tends to be smaller than compensation method. We use $\mathcal{O}^{(0)}$ to denote the Pauli transfer matrix of the ideal gate (without error). The estimation of the Pauli transfer matrix of the actual gate with noise (i.e. $\bar{\mathcal{O}}$) is $\hat{\mathcal{O}}$, which is obtained via GST. To compute the decomposition, first, we compute the ideal matrix $\mathcal{O}^{(0)}$; second, we compute the inverse of the noise

$$
\mathcal{N}^{-1} = \mathcal{O}^{(0)} \hat{\mathcal{O}}^{-1}; \tag{8.51}
$$

and finally, we solve the equation (for single-qubit gate)

$$\mathcal{N}^{-1} = \sum_i q_{\mathcal{O},i} \hat{\mathcal{B}}_i \tag{8.52}$$

to determine quasi-probabilities $q_{\mathcal{O},i}$ of the gate $\mathcal{O}$. We need to compute quasi-probabilities for each qubit and each gate. For example, for the SWAP-test circuit, we need to compute the decomposition for Hadamard gate, T gate, $T^\dagger$ gate of each qubit and controlled-NOT gate of each pair of qubits that the controlled-NOT gate may be performed on.

For two-qubit gates, the procedure is the same but tensor products of single-qubit basis operations, i.e. $\hat{\mathcal{B}}^{(1)}_{i_1} \otimes \hat{\mathcal{B}}^{(2)}_{i_2}$ (where the superscribe is the label of the qubit), are used to decompose the inverse of the noise $\mathcal{N}^{-1}$.

In order to mitigate errors in initial states and measurements of observables, we should solve the following equations for the quantities $q^{(m)}_{\rho,k}$ and $q^{(m)}_{Q,j}$:

$$|\rho^{(0)}\rangle\rangle = \sum_k q^{(m)}_{\rho,k} |\hat{\rho}_k\rangle\rangle, \tag{8.53}$$

$$\langle\langle Q^{(0)}| = \sum_k q^{(m)}_{Q,j} \langle\langle \hat{Q}_j| \tag{8.54}$$

for each qubit. Here, m is the label of the qubit, $|\rho^{(0)}\rangle\rangle$ is the column vector representing the ideal initial state $|0\rangle\langle 0|$, and $\langle\langle Q^{(0)}|$ is the row vector representing the ideal observable σ^{z}.

Before implementing the quasi-probability decomposition on a quantum computer, we compute

$$C^{(m)}_\rho = \sum_k |q^{(m)}_{\rho,k}|, \tag{8.55}$$

$$C^{(m)}_Q = \sum_k |q^{(m)}_{Q,j}| \tag{8.56}$$

for each qubit and

$$C_{\mathcal{O}} = \sum_i |q_{\mathcal{O},i}| \tag{8.57}$$

for each gate.

8.14.3 Monte Carlo implementation of the quasi-probability decomposition

It is vital to note that we use estimations $\hat{\mathcal{B}}_i$ (the result from GST) to decompose the inverse of the noise, but usually there is a difference between $\hat{\mathcal{B}}_i$ and the actual basis operation $\bar{\mathcal{B}}_i$. This difference does not cause any error in the final computing result, because the computing result is invariant under a similarity transformation as we have explained in section 8.9.

Now, we describe the procedure to implement the quasi-probability decomposition on a quantum computer. We suppose the circuit is sequentially performing gates $\mathcal{O}_1, \mathcal{O}_2, \ldots, \mathcal{O}_N$ on the initial state $|00\ldots0\rangle$, and the first qubit is measured in the σ^{z} basis to read the computing result. The procedure can be generalised to the case of measuring multiple qubits.

First, we generate a set of random integers: for each qubit m, we randomly select an integer k_m such that each integer would be selected with corresponding probability $|q^{(m)}_{\rho,k_m}|/C^{(m)}_{\rho}$; similarly for each gate l, we generate random integer i_l with corresponding probability $|q_{\mathcal{O}_l,i_l}|/C_{\mathcal{O}_l}$; and finally we generate random integer j_1 with the probability $|q^{(1)}_{Q,j_1}|/C^{(1)}_{Q}$.

Second, on the quantum computer, we implement the following quantum computation: we initialise the qubit m in the state $\bar{\rho}^{(m)}_{k_m}$; then we sequentially perform gates $\bar{\mathcal{O}}_1$, $\bar{B}_{i_1}$, $\bar{\mathcal{O}}_2$, $\bar{B}_{i_2}$, $\ldots$, $\bar{\mathcal{O}}_N$, $\bar{B}_{i_N}$; finally we measure the observable $\bar{Q}_{j_1}$. The measurement outcome is μ.

Third, we compute the effective measurement outcome, the product of the mea-

surement result and the parity corresponding to the applied operation, as

$$\mu_{\text{eff}} = \text{sgn}\left(\prod_m q^{(m)}_{\rho,k_m} \prod_l q_{\mathcal{O}_l,i_l} q^{(1)}_{Q,j_1}\right)\mu. \tag{8.58}$$

By repeating these three steps, we can obtain the mean of effective outcomes $\text{E}[\mu_{\text{eff}}]$. The final computing result is $C\text{E}[\mu_{\text{eff}}]$, where

$$C = \prod_m C^{(m)}_\rho \prod_l C_{\mathcal{O}_l} C^{(1)}_Q. \tag{8.59}$$

8.15 Discussion

We have demonstrated that, following our protocol step by step, an experimentalist can derive an algorithm to run on a noisy quantum computer so as to estimate the ideal output observable without any bias. The experimentalist does not require any prior knowledge of the physical property of the noise, and the only condition is that the noise is localised and Markovian. For this purpose, we have shown that quantum gate set tomography is a perfect tool for measuring the noise in a quantum computer, if the aim is only to compensate the effect of the noise in quantum computing, where only the expected value of observable is of interest; and we also have shown that single-qubit Clifford gates and measurement can derive a complete set of operations that can compensate any noise in quantum computing.

The price of using such a systematic method to cancel computing errors is that the quantum computation needs to run for a longer time than an error-free system. We verify the protocol with numerical simulations of up to 19 qubits, in which an alternative method, i.e. exponential error extrapolation, is introduced and studied. We find that the estimation using exponential error extrapolation is also very accurate, while the computing time might be shorter. An approach combining two methods may optimise both accuracy and efficiency.

Our general conclusion is that these quantum error mitigation techniques can

dramatically enhance the performance of quantum computers, especially at the small-to-medium scale where full fault tolerant quantum error correction is impossible. Our simulations have considered circuits up to 19 qubits, but with error rates considerably *worse* than the current achievable error rates. Extrapolating from the trends that we observe in these smaller systems, we anticipate that hybrid algorithms involving 50+ qubits, which is beyond the reach of classical emulation, will benefit from QEM techniques if the hardware fidelity matches today's state-of-the-art error or modestly improves upon it.

Chapter 9

Mitigating algorithmic errors

9.1 Introduction

Realising the time evolution operator $U(t) = e^{-iHt}$ for a given Hamiltonian H is a vital step for quantum simulation, which can be used for studying both its dynamic [2] and static [3] properties. Several methods have been proposed to efficiently approximate the time evolution operator $U(t)$ [67, 68, 152, 69, 153]. Although the latest methods [152, 69] have been significantly improved, the simulation accuracy is still limited by finite resources, such as short circuit depth, finite system runtime, and large physical errors in the system. To study such a limitation, we focus on the the Trotterisation method [154], introduced for quantum simulation by Lloyd [2].

In Ref. [2], decomposing the time evolution unitary operator into a product form has been shown to be an efficient method for approximating $U(t)$. Suppose a Hamiltonian H for the simulated system can be decomposed into a sum of local Hamiltonians H_k as $H = \sum_k H_k$. By using the Trotter-Suzuki formula, we can approximate the time evolution unitary operators as

$$U(t) = \big(\prod_k \exp(-iH_k t/N)\big)^N + O(t^2/N). \tag{9.1}$$

Note that there are higher order product formulas [155] and protocols involving

randomisation [68], but for our purposes this simple expression suffices.

Here, $O(t^2/N)$ is the algorithmic error from the insufficient discretisation. As the number of the Trotter steps N increases, the algorithmic error can be suppressed in the ideal situation. However, for NISQ era devices [156], noise such as gate errors and decoherence inevitably afflicts the quantum systems, and the errors accumulate by increasing the number of the Trotter steps. Consequently, we can only have restricted Trotter step number and cannot necessarily obtain the desired accuracy, which was shown in Ref. [157].

Recently, as we discussed in previous chapters, quantum error mitigation (QEM) was proposed for compensating for physical errors [24, 28, 80, 29, 158, 81, 82]. One of such methods is the extrapolation technique [24, 28, 29] as explained in detail in chapter 5. The strategy is, by deliberately boosting the error rate and using the several points with higher error rate, we can infer the ideal error free expectation value of the observable.

In this chapter, we apply the extrapolation technique to the Trotter decomposition. For example, suppose that we consider two cases where we either employ N_1 Trotter steps, or instead N_2 steps where $N_1 > N_2$. By extrapolating the expectation values obtained from Trotter step N_1 and N_2, we can estimate the more accurate expectation value corresponding to the Trotter number $N_{\mathrm{id}} > N_1$. Moreover, by combining this algorithmic error mitigation method with the originally proposed extrapolation method for mitigating physical errors, we will show that further improvement of the accuracy is achieved.

9.2 The Optimal number of Trotter steps for noisy quantum simulation

As mentioned in section 9.1, the number of Trotter step cannot be infinitely increased, and is restricted due to physical noise. In this section, we first review the

theoretical analysis [157] about the optimal number of Trotter steps in Hamiltonian simulation with physical noise. Denote the channel of the Trotter decomposition $e^{-iH_k t/N}$ as $\mathcal{V}_k$ and the physical noise as an extra channel $\mathcal{E}_k$, when we only consider Markovian errors, the noisy stroboscopic channel for the i^{th} Trotter step is

$$\mathcal{E}_i^{\text{strobo}} = \mathcal{E}_L \circ \mathcal{V}_L \circ \mathcal{E}_{L-1} \circ \mathcal{V}_{L-1} ... \circ \mathcal{E}_1 \circ \mathcal{V}_1, \tag{9.2}$$

where L is the number of the local Hamiltonians. When we use the number of Trotter steps N, the entire noisy channel of the Trotter decomposition is

$$\mathcal{E}^{\text{noisyTrotter}} = \mathcal{E}_N^{\text{strobo}} \circ \mathcal{E}_{N-1}^{\text{strobo}} \circ \cdots \circ \mathcal{E}_1^{\text{strobo}}. \tag{9.3}$$

The distance between two channels $\mathcal{E}_1$ and $\mathcal{E}_2$ is defined by the trace distance or distinguishability between the output states of the two channels [159, 160],

$$D(\mathcal{E}_1, \mathcal{E}_2) = \max_{\rho} \|\mathcal{E}_1(\rho) - \mathcal{E}_2(\rho)\|, \tag{9.4}$$

where the maximisation is over a properly chosen state set and $\|M\| = \text{Tr}[\sqrt{M^\dagger M}]$ for matrix M. The distance between the ideal channel $\mathcal{U}^{\text{ideal}}$ for the evolution e^{-iHt} and the noisy implementation $\mathcal{E}^{\text{noisyTrotter}}$ is

$$\begin{aligned}
&D\left(\mathcal{U}^{\text{ideal}}, \mathcal{E}^{\text{noisyTrotter}}\right) \\
&\leq \sum_{i=1}^{N} D\left(\sqrt[N]{\mathcal{U}^{\text{ideal}}}, \mathcal{E}_i^{\text{strobo}}\right), \\
&\leq \sum_{i=1}^{N} D\left(\sqrt[N]{\mathcal{U}^{\text{ideal}}}, \sqrt[N]{\mathcal{V}^{\text{nonoise}}}\right) + D\left(\mathcal{E}_i^{\text{strobo}}, \sqrt[N]{\mathcal{V}^{\text{nonoise}}}\right),
\end{aligned} \tag{9.5}$$

where the second line follows from the chaining property

$$D\left(\mathcal{E}_1 \circ \mathcal{E}_2, \mathcal{E}_1' \circ \mathcal{E}_2'\right) \leq D\left(\mathcal{E}_1, \mathcal{E}_1'\right) + D\left(\mathcal{E}_2, \mathcal{E}_2'\right), \tag{9.6}$$

the third line follows from the triangle inequality

$$D\left(\mathcal{E}_1, \mathcal{E}_2\right) \leq D(\mathcal{E}_1, \mathcal{E}_3) + D\left(\mathcal{E}_2, \mathcal{E}_3\right), \tag{9.7}$$

and $\sqrt[N]{\mathcal{V}^{\text{nonoise}}}$ is the channel of the noise free stroboscopic evolution $\prod_k e^{-iH_k t/N}$ of each Trotter decomposition. Now, we define the algorithmic and physical errors ε_{alg} and ε_{phys} as

$$\begin{aligned} \varepsilon_{alg} &= \sum_{i=1}^{N} D\left(\sqrt[N]{\mathcal{U}^{\text{ideal}}}, \sqrt[N]{\mathcal{V}^{\text{nonoise}}}\right) = \frac{\alpha}{N}, \\ \varepsilon_{phys} &= \sum_{i=1}^{N} D\left(\mathcal{E}_i^{\text{strobo}}, \sqrt[N]{\mathcal{V}^{\text{nonoise}}}\right) = \beta N, \end{aligned} \tag{9.8}$$

where $\alpha = D\left(\sqrt[N]{\mathcal{U}^{\text{ideal}}}, \sqrt[N]{\mathcal{V}^{\text{nonoise}}}\right) N^2$ and $\beta = D\left(\mathcal{E}^{\text{strobo}}, \sqrt[N]{\mathcal{V}^{\text{nonoise}}}\right)$. Here, for simplicity, we assume that the noise model is the same for each stroboscopic sequence, that is $\mathcal{E}_i^{\text{strobo}}$ is the same for different i. Note that $\varepsilon_{alg} \propto 1/N$ reflects that the algorithmic error can be linearly mitigated with an increasing number of Trotter steps; while $\varepsilon_{phys} \propto N$ is because errors linearly accumulate as the number of Trotter steps increases.

Now, we optimise the distance

$$D\left(\mathcal{U}^{\text{ideal}}, \mathcal{E}^{\text{noisyTrotter}}\right) = \frac{\alpha}{N} + \beta N, \tag{9.9}$$

so that we can get the optimised number of Trotter step as

$$N_{\text{opt}} = \sqrt{\alpha/\beta}, \tag{9.10}$$

and we have trace distance $D = 2\sqrt{\alpha\beta}$ corresponding to N_{opt}. Therefore, due to the existence of physical errors, we cannot choose an infinitely large number of Trotter steps to compensate for the algorithmic error. Although it is hard to analytically calculate α and β for a general physical Hamiltonian and noise models, we will numerically investigate the optimal number of Trotter steps in Sec. 9.4.

9.3 Error mitigation for algorithmic errors

In this section, we discuss how the algorithmic error in Trotterisation can be considered to be similar to physical error and how to generalise the extrapolation error mitigation method to suppress the algorithmic error. Given a Hamiltonian H that has decomposition $H = \sum_k H_k$, the first order Trotter formula approximates the time evolution operator $U(t) = e^{-i\sum_k H_k t}$ by decomposing it into a product form,

$$U(t) = \left(\prod_k e^{-iH_k t/N}\right)^N + \sum_{i<j}[H_i, H_j]t^2/2N + \sum_{m=3}^{\infty} E(m), \tag{9.11}$$

where the higher order terms $E(m)$ can be upper bounded by $\|E(m)\| \leq N\|Ht/N\|^m/m!$, and $\|\cdot\|$ denotes the maximal eigenvalue of the matrix. Denote $\varepsilon_N = 1/N$ and $U_{\varepsilon_N}(t)$ as

$$U_{\varepsilon_N}(t) \equiv \left(\prod_k e^{-iH_k t\varepsilon_N}\right)^{1/\varepsilon_N}, \tag{9.12}$$

then we can straightforwardly see how $U_{\varepsilon_N}(t)$ approximates $U(t) = e^{-i\sum_k H_k t}$, i.e.,

$$\lim_{\varepsilon_N \to 0^+} U_{\varepsilon_N}(t) = U(t) = e^{-i\sum_k H_k t}. \tag{9.13}$$

Suppose the time evolution operator $U_{\varepsilon_N}(t)$ is applied to an initial state $|\psi_0\rangle$, then the output state is $U_{\varepsilon_N}(t)|\psi_0\rangle$. When we measure observable $\hat{A}$ of the final output state, the average value is

$$\langle \hat{A}(t)\rangle(\varepsilon_N) = \langle\psi_0| U_{\varepsilon_N}(t)^\dagger \hat{A} U_{\varepsilon_N}(t)|\psi_0\rangle. \tag{9.14}$$

On the other hand, the error-free expectation value for the observable is $\langle \hat{A}(t)\rangle(0) = \langle\psi_0| e^{iHt}\hat{A}e^{-iHt}(t)|\psi_0\rangle$. Regarding $\langle \hat{A}(t)\rangle(\varepsilon_N)$ as a function of ε_N, we can expand $\langle \hat{A}(t)\rangle(\varepsilon_N)$ as a function of ε_N by using the Taylor expansion

$$\langle \hat{A}(t)\rangle(\varepsilon_N) = \langle \hat{A}(t)\rangle(0) + \sum_{j=1}^{n} \varepsilon_N^j \hat{A}(t)_j + O(\varepsilon_N^{j+1}), \tag{9.15}$$

where $\langle \hat{A}(t) \rangle (0)$ and $\hat{A}(t)_j$ are independent of ε_N.

Therefore, the extrapolation error mitigation method can be also applied to suppress the algorithmic error. In the original error mitigation scheme for suppressing physical errors, we may choose to boost the error rate ε to several different error rates $a_j\varepsilon$, which can be realised by intensionally adding more noise to the circuit, as mentioned in section 5.2.1. Here, as the error rate ε_N is $1/N$, we can straightforwardly increase the algorithmic error ε_N by using a smaller number of Trotter steps. For instance, by taking $N' = N/2$, we can effectively double the algorithmic error $\varepsilon_{N'} = 2\varepsilon_N$. With n different numbers of Trotter steps $N_n \leq \cdots \leq N_1 \leq N_0 = N_{\text{opt}}$ and $N_j = N_0/a_j$, we can therefore suppress the algorithmic error by a linear combination of the results

$$\begin{aligned}
&\langle \hat{A}(t) \rangle_{est} (0) \\
&= \sum_{i=0}^{n} \gamma_i' \langle \hat{A}(t) \rangle (\varepsilon_{N_i}) \\
&= \sum_{i=0}^{n} \gamma_i' \langle \hat{A}(t) \rangle (0) + \sum_{j=1}^{n} \hat{A}(t)_j \varepsilon_{N_{\text{opt}}}^j \sum_{i=0}^{n} \gamma_i' a_i^j + O(\varepsilon_{N_{\text{opt}}}^{n+1}) \\
&= \langle \hat{A}(t) \rangle (0) + O(\varepsilon_{N_{\text{opt}}}^{n+1}).
\end{aligned} \tag{9.16}$$

Here, γ_i' is chosen such that $\sum_{i=0}^{n} \gamma_i' = 1$, $\sum_{i=0}^{n} \gamma_i' a_i^j = 0$ for $j = 1, 2, \ldots, n$. Therefore, we can suppress the algorithmic error to an order of $O((\varepsilon_{N_{\text{opt}}})^{n+1}) = O((1/N_{\text{opt}})^{n+1})$.

The variance of the estimation $\langle \hat{A} \rangle_{est} (0)$ is

$$\text{Var}\big(\langle \hat{A} \rangle_{est} (0) \big) = \sum_{i=0}^{n} \gamma_i'^2 \text{Var}\big(\langle \hat{A} \rangle_{est} (\varepsilon_{N_i}) \big). \tag{9.17}$$

Thus, by assuming that the variance is the same for different $\langle \hat{A} \rangle_{est} (\varepsilon_{N_i})$, the variance of the estimation $\text{Var}\big(\langle \hat{A} \rangle_{est} (0) \big)$ is $\sum_{i=0}^{n} \gamma_i'^2$ times larger than the variance of $\langle \hat{A} \rangle_{est} (\varepsilon_{N_i})$. To have the same precision as the case without error mitigation, we need to have $\sum_{i=0}^{n} \gamma_i'^2$ times more samples, which is the cost of this method. We

denote the multiplicative cost of the extrapolation for algorithmic error by

$$\Gamma_{alg} \equiv \sum_{i=0}^{n} \gamma_i'^2. \tag{9.18}$$

Moreover, we can combine the extrapolation for physical errors with the extrapolation for algorithmic errors. The possible scenario is, first, mitigating the physical error with extrapolation for physical error, and second, applying extrapolation for algorithmic error to the data whose physical error is suppressed. Consequently, we can obtain the expected value of observables, in which both physical and algorithmic errors are mitigated. Then, the total cost is

$$\Gamma_{alg+phys} \equiv \Gamma_{alg} \times \Gamma_{phys}, \tag{9.19}$$

where Γ_{phys} is the cost for physical error mitigation.

Note that, the only requirement for the extrapolation method is that $U_{\varepsilon_N}(t)$ can be expressed as an explicit function of ε_N, and $\lim_{\varepsilon_N \to 0^+} U_{\varepsilon_N}(t) = U(t)$. Therefore, the same argument can be applied to the higher order Trotterisation [67]. Furthermore, this method can still be applied even if the expectation value cannot be efficiently expanded as a function of $1/N$. The method also works as long as $\langle \hat{A} \rangle$ can be expanded with ε which is a function $f(N_{\mathrm{t}})$ of the tunable parameter N_{t}.

The details of upper bound for algorithmic error for the linear extrapolation case is as follows. By using the first order Trotter formula, we can approximate the time evolution unitary operator as

$$U(t) = \exp(-iHt) = \left(\prod_i \exp(-iH_i t/N) \right)^N \tag{9.20}$$

$$+ \sum_{i<j} [H_i, H_j] t^2/2N + \sum_{m=3}^{\infty} E(m), \tag{9.21}$$

where $E(m)$ can be upper bounded by $\|E(m)\| \leq N\|Ht/N\|^m/m!$, $\|\cdot\|$ denotes

the maximal eigenvalue. We denote U_N as $\left(\prod_i \exp(-iH_i t/N)\right)^N$, $\varepsilon_N = 1/N$ and $U_{\varepsilon_N}(t) \equiv \left(\prod_k e^{-iH_k t\varepsilon_N}\right)^{1/\varepsilon_N}$ which converges to $U(t)$ with $\varepsilon_N \to 0^+$. The difference between $U(t)$ and $U_{\varepsilon_N}(t)$ can be rewritten as

$$U(t) = U_{\varepsilon_N}(t) + a/N + E_N, \tag{9.22}$$

where $a = \sum_{i<j}[H_i, H_j]t^2/2$ and $E_N = \sum_{m=3}^{\infty} E(m)$ that can be bounded by

$$\begin{aligned} \|E_N\| &\leq N \sum_{m=3}^{\infty} \|Ht/N\|^m/m!, \\ &< \frac{\|H\|^3 t^3}{6N^2} e^{\|H\|t/N}. \end{aligned} \tag{9.23}$$

Suppose the time evolution operator $U_{\varepsilon_N}(t)$ is applied to an initial state $|\psi_0\rangle$, the output state is $U_{\varepsilon_N}(t)|\psi_0\rangle$. When we measure observable A of the final output state, the average value is

$$\langle \hat{A}(t)\rangle(\varepsilon_N) = \langle\psi_0| U_{\varepsilon_N}(t)^\dagger \hat{A} U_{\varepsilon_N}(t) |\psi_0\rangle. \tag{9.24}$$

and the average value for evolution $U(t)$ is $\langle\psi_0| e^{iHt}\hat{A}e^{-iHt}|\psi_0\rangle$. The relationship between them can be expressed as

$$\begin{aligned} &\langle \hat{A}(t)\rangle(\varepsilon_N) \\ &= \langle \hat{A}(t)\rangle(0) - 1/N[\langle\psi_0| U^\dagger \hat{A} a |\psi_0\rangle + \langle\psi_0| a^\dagger \hat{A} U |\psi_0\rangle] \\ &+ \langle\psi_0| (a/N + E_N)^\dagger \hat{A}(a/N + E_N) |\psi_0\rangle \\ &- [\langle\Psi| U^\dagger \hat{A} E_N |\Psi\rangle + \langle\Psi| (E_N)^\dagger \hat{A} U |\Psi\rangle] \\ &= \langle \hat{A}(t)\rangle(0) + b/N + R_N \end{aligned} \tag{9.25}$$

where $b = -[\langle\psi_0| U^\dagger \hat{A} a |\psi_0\rangle + \langle\psi_0| a^\dagger \hat{A} U |\psi_0\rangle]$ is independent of N, and $R_N = \langle\psi_0| (a/N+E_N)^\dagger \hat{A}(a/N+E_N) |\psi_0\rangle - [\langle\psi_0| U^\dagger \hat{A} E_N |\psi_0\rangle + \langle\psi_0| (E_N)^\dagger \hat{A} U |\psi_0\rangle]$ is bound

by

$$R_N \leq \left[\left(\frac{\|a\|}{N} + \frac{\|H\|^3 t^3}{6N^2} e^{\|H\|t/N} \right)^2 + \frac{\|H\|^3 t^3}{3N^2} e^{\|H\|t/N} \right] \|\hat{A}\|. \tag{9.26}$$

It is not hard to see that $R_N = O(1/N^2)$

Now we use the linear combination of $\langle \hat{A}(t) \rangle (\varepsilon_{N_1})$ and $\langle \hat{A}(t) \rangle (\varepsilon_{N_2})$ to estimate $\langle \hat{A}(t) \rangle (0)$ by

$$\langle \hat{A}(t) \rangle_{est} (0) = \beta_1 \langle \hat{A}(t) \rangle (\varepsilon_{N_1}) + \beta_2 \langle \hat{A}(t) \rangle (\varepsilon_{N_2}). \tag{9.27}$$

We choose the parameter β_1, β_2 satisfying $\beta_1/N_1 + \beta_2/N_2 = 0$ and $\beta_1 + \beta_2 = 1$ so that the difference between $\langle \hat{A}(t) \rangle_{est} (0)$ and $\langle \hat{A}(t) \rangle (0)$ is

$$| \langle \hat{A}(t) \rangle_{est} (0) - \langle \hat{A}(t) \rangle (0)| \leq R_{N_1} \beta_1 + R_{N_2} \beta_2. \tag{9.28}$$

Based on the two point extrapolation method, we use the simulation results with Trotter steps N_1 and N_2, which both are $O(N)$ to reduce the algorithmic error from $O(1/N)$ to $O(1/N^2)$.

9.4 Numerical simulation

In this section, we consider a five qubit system and numerically test our new error mitigation method for algorithmic errors when we simulate real time evolution. As shown in Fig. 9.1, the Hamiltonian only has local and near neighborhood interactions,

$$H = J \sum_{i=1}^{5} Z_3 Z_i + B \sum_{i=1}^{5} X_i, \tag{9.29}$$

where X_i (Z_i) denotes the spin half Pauli x (z) operator acting on the ith qubit, $J = 3$, and $B = 2$. In our numerical simulation, we initialise the state to $|\psi_0\rangle = |0, 0, 0, 0, 0\rangle$, evolve it with time t, and measure $\hat{A} = X_1$.

We consider both algorithmic error from the finite number of Trotter steps and

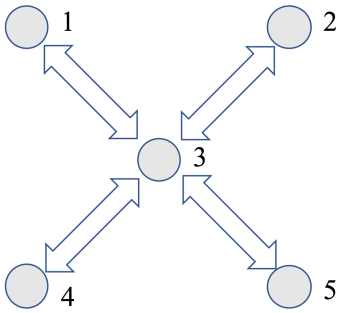


Figure 9.1: Schematic of the five qubit Hamiltonian.

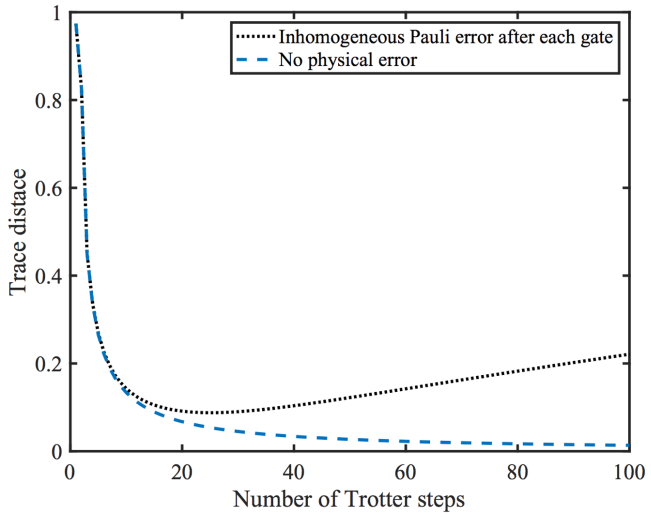
physical error from gate noise. As we will mention later, we apply the extrapolation method for both physical and algorithmic errors, using the procedure discussed in section 9.3. We consider inhomogeneous Pauli error for both single and two qubit gates,

$$\mathcal{E}(\rho) = (1-p)\rho + p_x X\rho X + p_y Y\rho Y + p_z Z\rho Z, \tag{9.30}$$

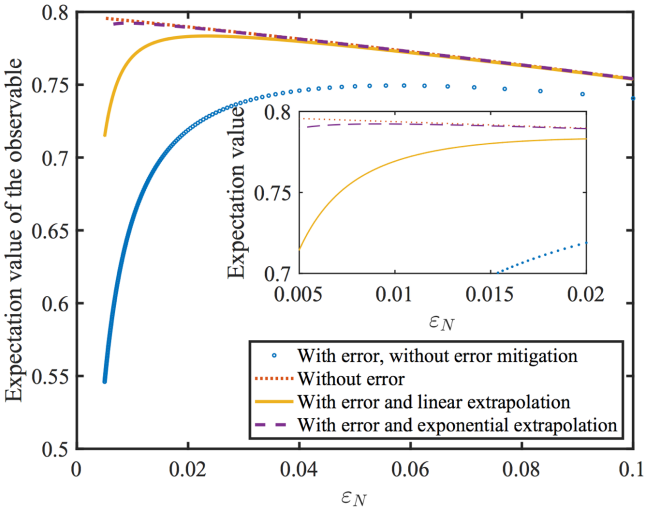
where $p = p_x + p_y + p_z$. For single qubit gates, we set $p_x = p_y = 2.0 \times 10^{-5}$, $p_z = 6.0 \times 10^{-5}$; for two qubit gate, we take error channel $\mathcal{E}_2 = \mathcal{E} \otimes \mathcal{E}$ and set $p_x = p_y = 1.0 \times 10^{-4}$, and $p_z = 3.0 \times 10^{-4}$ for each $\mathcal{E}$. The two qubit gate error rate is 10^2 times higher as can be typical in actual experimental setups (see e.g. Refs [15, 16].) Note that such a noise rate corresponds to the current state-of-the-art experiment system [15, 16].

Without considering gate error and shot noise to the measurement, an infinite number of Trotter steps should be used to increase the simulation accuracy. When gate errors are present, the optimal number of Trotter steps is limited as shown in Fig. 9.2(a). Here, we fix the total evolution time $t = 0.5$ and numerically find the optimal number of Trotter steps to be 25.

To mitigate the algorithmic error, we use two or three different numbers of Trotter steps to estimate the value corresponding to an infinite number of Trotter steps. For a given number of Trotter steps N, the runtime of the circuit is related to the circuit depth, which is proportional to N, and the circuit repetition time m, which is used to get an accurate estimation of the measurement. We denote



(a)



(b)

Figure 9.2: The optimal number of Trotter steps N and the continuity of the measurement as a function of the inverse of the number of Trotter steps $\varepsilon_N = 1/N$. (a) The trace distance between the the ideal state and the simulated state in the presence of noise. The blue curve denotes the trace distance without physical error, which goes to zero with an infinite large number of Trotter steps. The black curve denotes the trace distance with inhomogeneous Pauli error after each gate and the optimal number of Trotter steps is 25. (b) The expectation value of the observable versus $\varepsilon_N = 1/N$. Here N is the number of Trotter steps. The horizontal axis corresponds to the number of Trotter step $10 - 200$. From the simulation result, we can confirm the continuity of the function.

the total runtime resource cost M as $M = mN$. To compare different simulation scenarios, we thus consider the same total cost M to have a fair comparison. For three- and two- point extrapolation for algorithmic error, we divide M equally to different number of Trotter steps. For instance, for three Trotter step extrapolation,

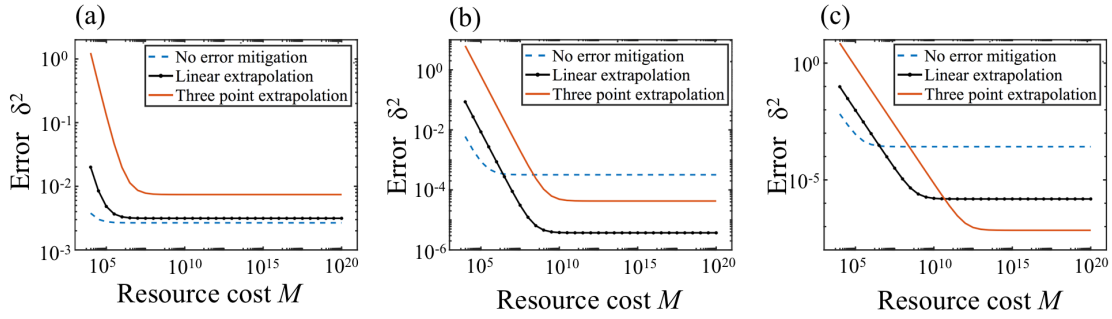


Figure 9.3: The error δ^2 against the total runtime resource M for different cases with no algorithmic error mitigation, linear algorithmic error extrapolation and three-point algorithmic error extrapolation. We consider physical errors to the circuit and set the total evolution time $t = 0.5$. The number of Trotter steps $N = 25, 20, 15$, $N = 25, 15$, $N = 25$ are used for the three-point, linear extrapolation and no error mitigation cases, respectively. Blue line denotes the no error mitigation case, black line denotes the linear algorithmic error extrapolation case, red line denotes the three-point algorithmic error extrapolation case. (a) No physical error mitigation is applied. (b) Linear extrapolation is applied to suppress physical errors. (c) Exponential extrapolation is applied to suppress physical errors. Here, we set the ratio r of the two error rates to be 2 for both linear and exponential extrapolation.

$N_1^{(3)}$, $N_2^{(3)}$, and $N_3^{(3)}$, we have $m_i^{(3)} = M/(3N_i^{(3)})$ for $i = 1, 2, 3$; for two Trotter step extrapolation, $N_1^{(2)}$, $N_2^{(2)}$, $m_i^{(2)} = M/(2N_i^{(2)})$ for $i = 1, 2$; and for one number of Trotter steps $N_1^{(1)}$, $m_1^{(1)} = M/N_1^{(1)}$.

In our simulation, we compare the three cases for suppressing algorithmic errors: no error mitigation, linear extrapolation, and three points extrapolation. In order to quantify the performance of our simulation method, we evaluate the error of the estimation value $\langle \hat{A}(t) \rangle (0)_{est}$ by

$$\delta^2 = (\langle \hat{A}(t) \rangle (0) - \langle \hat{A}(t) \rangle (0)_{est})^2, \tag{9.31}$$

where $\langle \hat{A}(t) \rangle (0)$ is the error-free average value.

Now, we show that physical and algorithmic errors can be suppressed with the error mitigation methods. To begin with, we check the continuity of the $\langle \hat{A}(t) \rangle$ as a function of $\varepsilon_N = 1/N$, to show that $\langle \hat{A}(t) \rangle$ can be Taylor expanded by ε_N. We consider the case with and without physical error mitigation. We measure

the expectation values with original error probability p and twice boosted error probability $2p$. Then by applying the linear and exponential extrapolation methods, we can suppress the physical errors. As shown in Fig. 9.2 (b), we can see that $\langle \hat{A}(t) \rangle$ is indeed a continuous function of ε_N, which confirms the possibility of algorithmic error mitigation. Furthermore, the best accuracy achieved by linear and exponential extrapolation is $\delta^2 = 2 \times 10^{-4}$ and 2.79×10^{-5}, with the number of Trotter steps $N = 43$ and 109, respectively. Therefore it is clear that we cannot increase the number of Trotter steps infinitely even when the extrapolation method is applied.

To show the effect of algorithmic error mitigation, we also consider three cases for physical errors: no error mitigation, linear extrapolation, and exponential extrapolation. First, we consider the case where no error mitigation is employed for the physical error. In such a case, physical errors still dominate and we find that algorithmic error extrapolation cannot improve the simulation accuracy, as shown in Fig. 9.3(a). This is because, although we suppress the algorithmic error by linearly combining the results from different number of Trotter steps, due to the large deviation from physical errors, the estimation becomes worse. As the total resource M increases, the shot noise is suppressed and the accuracy converges. The accuracy δ^2 at the converged point without algorithmic and physical errors mitigations is 2.67×10^{-3}.

Next, we consider the cases where linear extrapolation is applied to suppress physical errors. Subsequently, we apply the algorithmic error extrapolation to suppress algorithmic errors due to Trotterisation. As shown in Fig. 9.3(b), we find that the algorithmic linear extrapolation outperforms the no error mitigation and three point extrapolation cases for large M. The converged accuracy with sufficiently large $M \gtrsim 10^8$ is $\delta^2 = 3.72 \times 10^{-6}$ under linear extrapolation of algorithmic errors. The improvement of the accuracy is 717 times, compared with the case where no error mitigation is applied for both physical and algorithmic errors.

Finally, in Fig. 9.3(c), we plot the result for the case where exponential extrapo-

lation is used for suppressing physical errors. It can be seen that the physical error is successfully reduced, and three point extrapolation works properly, outperforming the performance of linear extrapolation for large $M \gtrsim 10^{13}$. The converged accuracy δ^2 for sufficiently large M is 6.95×10^{-8}. This accuracy correspond to 3.8×10^4 times improvement, compared with the case where no error mitigation is applied for both physical and algorithmic errors.

9.5 Discussion

In this chapter, we described an error mitigation method for suppressing algorithmic errors in Trotterisation of Hamiltonian simulation. We first show that due to physical noise, the number of Trotter steps cannot be increased infinitely, and the optimal number of Trotter steps is finite. Then, we show how the recently proposed physical error mitigation methods can be extended to suppress algorithmic errors in Trotterizsation. We numerically test our algorithmic error mitigation method in a five qubit Hamiltonian and show how it can improve simulation accuracy by combining it with physical error mitigation. Although we only focus on the first-order Trotterisation, our scheme can also be extended to other Trotterisation schemes, such as higher-order Trotterisation [67] and randomisation Trotterisation [68]. This is because the expectation values obtained from these methods can also be written as a function of the number of Trotter steps. Although, in the presence of physical errors, higher-order Trotter decompositions may not reduce the overall error [24], we leave these extensions in future works. Moreover, the other recently proposed Hamiltonian simulation methods, e.g. Taylor series [152] and quantum signal processing [69, 153], offer alternative ways for Hamiltonian simulation instead of Trotterisation. These methods divide the simulation time t into N segments and simulate the evolution for each time t/N. Considering $1/N$ as an error, our extrapolation method can also be applied by modifying N and extrapolate the results

with different N to improve the estimation accuracy.

The extrapolation method has broad applications in diverse fields, including error mitigation, computational chemistry, linear optics simulation, etc. In classical computational chemistry, extrapolation is widely used in solving molecular structure problem and Monto Carlo simulation of dynamics [161, 162]. In linear optics simulation, the extrapolation method is used to simulate single photons with imperfect photon sources such as coherent states from lasers [78]. We may be able to also apply this formalism for mitigating the algorithmic error to other schemes, such as variational quantum eigensolver (VQE) [19, 65, 20, 138, 80, 62, 22]. The result obtained from VQE can be regarded as a function of the depth of the quantum circuit, and the extrapolation method can be applied. For example, considering the hardware efficient ansatz [20] with different circuit depths, one can get simulation results with different accuracy. Therefore, by running experiments with different small depths, the extrapolation method can be applied to infer the results with a much larger depth. However, when the algorithm gets trapped in local minima, the error mitigation method will fail to work. We leave the discussion about the practical performance to future works. Similar arguments can also be applied to other types of variational algorithms, for simulating real or imaginary time evolutions [24, 46, 27].

Also, it may be possible to apply this method to quantum error correction, because the expectation of an observable should be also a function of the size of the code, e.g. when we consider n-qubit bit flip code ($n = 3$ case is mentioned in section 3.2), the computing result is a function of n. Therefore, by using the results from the codes with different size, the extrapolation may be applied to infer the computing result from a larger code size.

Conclusion

In this DPhil thesis, firstly in chapter 2, we introduced the basics of quantum mechanics required to understand the thesis. In chapter 3, we discussed the early quantum algorithms and error correction and explained that they are highly expensive with regard to the number of required qubits. As a solution to it, in chapter 3, the hybrid quantum-classical algorithm was introduced. However, the applications of hybrid algorithms were quite restricted i.e., the ground state search of molecules and simulation of dynamics of quantum systems. In chapter 4, we discussed that the error mitigation is also necessary to make the best of such algorithms, but firstly proposed methods were not practical in that the tolerable error rate was low and the applicable noise model was restricted, and suitable tomography method to construct the error mitigation operations was unclear.

As accomplishments of the author's research, firstly, we have significantly broadened the applicability of hybrid quantum-classical algorithms.

In chapter 6, we introduced a new hybrid algorithm for finding the spectrum of the given Hamiltonian, required for many applications including the design of solar cells and creation of new drugs. We benchmarked our new quantum algorithm and it successfully discovered the spectrum of 18 qubit 3SAT and 10 qubit LiH Hamiltonian.

In chapter 7, we introduced a hybrid algorithm for simulating a broad range of general processes, based on variational real and imaginary time quantum simulation. It may be useful for machine learning tasks for example, and there should be variety

of applications. We applied our algorithm to simulation of open quantum system by using stochastic Schödinger equation, and it successfully worked for 3 qubit one-dimensional Ising model under amplitude damping.

As a second theme of research, we also contributed to quantum error mitigation for hybrid algorithms. In chapter 8, we have rendered the quantum error mitigation practical, and tolerable error rate became much higher and applicable noise models were significantly broadened. Furthermore, we found that the gate set tomography is a suitable tomography method for constructing error mitigation operations. Consequently, we made error mitigation techniques applicable to suppress errors of current quantum devices and it may make quantum devices useful for solving intractable problems for classical computers.

Finally, in chapter 9, we introduced the quantum error mitigation technique for mitigating algorithmic errors, which may be useful for Hamiltonian simulation.

To conclude, we have significantly generalised the hybrid algorithms and developed the quantum error mitigation, leading to the realisation of practical near-term quantum computing and quantum supremacy. It is the author's hope that the work he has performed during his doctorate can contribute to the community's efforts to make these fascinating and exciting machines become a reality.

As a future research, we consider applications of variational quantum algorithms to completely different fields such as quantum metrology. If we optimise the value related to the quality of quantum metrology, e.g., quantum fisher information, we can simultaneously obtain the suitable state for metrology. Also, we may be able to find another cost function whose optimisation contributes to achieving useful tasks and prepares an state corresponding to them. As to error mitigation, the quasi-probability method we introduces requires gate set tomography, which is costly. If we can implement a quasi-probability method which does not rely on tomography, it should be useful. A quasi-probability method based on models (such as Lindblad master equation etc.) might reduce the cost for quasi-probability method.

Appendices

A Stability of the quantum gate set tomography

In this section, we discuss the stability of GST. Now we define

$$\begin{aligned}
\bar{\varepsilon}_{\text{in}} &= \max\{\|\bar{M}_m^{\text{in}} - M^{\text{in}(0)}\|_2 | m = 1, \ldots, n\}, && (32)\\
\bar{\varepsilon}_{\text{out}} &= \max\{\|\bar{M}_m^{\text{out}} - M^{\text{out}(0)}\|_2 | m = 1, \ldots, n\}, && (33)\\
\bar{\varepsilon}_{\mathcal{O}} &= \|\bar{\mathcal{O}} - \mathcal{O}^{(0)}\|_2, && (34)
\end{aligned}$$

which describe severities of the initialisation error, measurement error and operation error, respectively. Here, $\bar{M}_m^{\text{in}}$ and $\bar{M}_m^{\text{out}}$ are matrices corresponding to the m^{th} qubit. The overall matrices of n qubits are $\bar{M}^{\text{in}} = \bigotimes_{m=1}^{n} \bar{M}_m^{\text{in}}$ and $\bar{M}^{\text{out}} = \bigotimes_{m=1}^{n} \bar{M}_m^{\text{out}}$.

Similar to the analyse of the linear independence of basis operations (i.e. the invertibility of the matrix A in section 8.8), we have that $\bar{M}^{\text{in}}$ and $\bar{M}^{\text{out}}$ are always invertible, i.e. $g = \bar{M}^{\text{out}}\bar{M}^{\text{in}}$ is always invertible, if $\bar{\varepsilon}_{\text{in}} < s_{\min}(M^{\text{in}(0)})$ and $\bar{\varepsilon}_{\text{out}} < s_{\min}(M^{\text{out}(0)})$. Choosing $\{\rho_k^{(0)}\}$ and $\langle\langle Q^{(0)}|$ as in section 8.8, we have $s_{\min}(M^{\text{in}(0)}) = \frac{1}{2}\sqrt{\frac{5-\sqrt{17}}{2}} \simeq 0.3311$ and $s_{\min}(M^{\text{out}(0)}) = 1$.

We choose $T = M^{\text{in}(0)\otimes n}$, then $\hat{M}^{\text{in}} = \bigotimes_{m=1}^{n} \hat{M}_m^{\text{in}}$ and $\hat{M}^{\text{out}} = \bigotimes_{m=1}^{n} \hat{M}_m^{\text{out}}$, where $\hat{M}_m^{\text{in}}$ and $\hat{M}_m^{\text{out}}$ are matrices corresponding to the m^{th} qubit. The severity of errors in estimations of initial states is

$$\hat{\varepsilon}_{\text{in}} = \max\{\|\hat{M}_m^{\text{in}} - M^{\text{in}(0)}\|_2 | m = 1, \ldots, n\} = 0, \quad (35)$$

and the severity of errors in estimations of observables is

$$\begin{aligned}
\hat{\varepsilon}_{\text{out}} &= \max\{\|\hat{M}_m^{\text{out}} - M^{\text{out}(0)}\|_2 | m = 1, \ldots, n\}\\
&\leq (\bar{\varepsilon}_{\text{out}}\bar{\varepsilon}_{\text{in}} + \|M^{\text{in}(0)}\|_2\bar{\varepsilon}_{\text{out}} + \|M^{\text{out}(0)}\|_2\bar{\varepsilon}_{\text{in}})\\
&\quad \times \|M^{\text{in}(0)-1}\|_2. && (36)
\end{aligned}$$

Here, we have used that

$$\begin{aligned}
& \hat{M}_m^{\rm out} - M^{{\rm out}(0)} \\
= \; & (\bar{M}_m^{\rm out}\bar{M}_m^{\rm in} - M^{{\rm out}(0)}M^{{\rm in}(0)})M^{{\rm in}(0)-1} \\
= \; & [(\bar{M}_m^{\rm out} - M^{{\rm out}(0)})(\bar{M}_m^{\rm in} - M^{{\rm in}(0)}) \\
& +(\bar{M}_m^{\rm out} - M^{{\rm out}(0)})M^{{\rm in}(0)} \\
& +M^{{\rm out}(0)}(\bar{M}_m^{\rm in} - M^{{\rm in}(0)})]M^{{\rm in}(0)-1}. \qquad (37)
\end{aligned}$$

Choosing $\{\rho_k^{(0)}\}$ and $\langle\langle Q^{(0)}|$ as in section 8.8, we have $\|M^{{\rm in}(0)}\|_2 = \frac{1}{2}\sqrt{\frac{5+\sqrt{17}}{2}} \simeq 1.0679$, $\|M^{{\rm out}(0)}\|_2 = 1$ and $\|M^{{\rm in}(0)-1}\|_2 = s_{\rm min}^{-1}(M^{{\rm in}(0)}) = 2\sqrt{\frac{2}{5-\sqrt{17}}} \simeq 3.0204$.

The severity of the error in the estimation of a n-qubit operation is

$$\begin{aligned}
\hat{\varepsilon}_{\mathcal{O}} &= \|\hat{\mathcal{O}} - \mathcal{O}^{(0)}\|_2 \\
&\leq \|\hat{\mathcal{O}} - \bar{\mathcal{O}}\|_2 + \|\bar{\mathcal{O}} - \mathcal{O}^{(0)}\|_2 \\
&\leq \frac{2\bar{\varepsilon}_{\rm in}^{(n)}}{[s_{\rm min}(M^{{\rm in}(0)})]^n - \bar{\varepsilon}_{\rm in}^{(n)}}(\|\mathcal{O}^{(0)}\|_2 + \bar{\varepsilon}_{\mathcal{O}}) \\
&\quad +\bar{\varepsilon}_{\mathcal{O}}, \qquad (38)
\end{aligned}$$

as we will show next. Here,

$$\bar{\varepsilon}_{\rm in}^{(n)} = (\|M^{{\rm in}(0)}\|_2 + \bar{\varepsilon}_{\rm in})^n - \|M^{{\rm in}(0)}\|_2^n. \qquad (39)$$

For an invertible matrix A, we have

$$\|A^{-1}\|_2 = \sup_{x\neq 0}\frac{\|A^{-1}x\|_2}{\|x\|_2} = \sup_{y\neq 0}\frac{\|y\|_2}{\|Ay\|_2}. \qquad (40)$$

Then, using the inequality (8.13), we have

$$\|A^{-1}\|_2 \leq \frac{1}{s_{\rm min}(A^{(0)}) - \|A - A^{(0)}\|_2}. \qquad (41)$$

We have the expression

$$\begin{aligned}
& \hat{\mathcal{O}} - \bar{\mathcal{O}} \\
= \; & M^{\mathrm{in}(0)\otimes n} \bar{M}^{\mathrm{in}-1} \bar{\mathcal{O}} \bar{M}^{\mathrm{in}} (M^{\mathrm{in}(0)\otimes n})^{-1} - \bar{\mathcal{O}} \\
= \; & (M^{\mathrm{in}(0)\otimes n} - \bar{M}^{\mathrm{in}}) \bar{M}^{\mathrm{in}-1} \bar{\mathcal{O}} \\
& \times (\bar{M}^{\mathrm{in}} - M^{\mathrm{in}(0)\otimes n}) (M^{\mathrm{in}(0)\otimes n})^{-1} \\
& + (M^{\mathrm{in}(0)\otimes n} - \bar{M}^{\mathrm{in}}) \bar{M}^{\mathrm{in}-1} \bar{\mathcal{O}} \\
& + \bar{\mathcal{O}} (\bar{M}^{\mathrm{in}} - M^{\mathrm{in}(0)\otimes n}) (M^{\mathrm{in}(0)\otimes n})^{-1}. \qquad (42)
\end{aligned}$$

First, we have $\|\bar{\mathcal{O}}\|_2 \leq \|\mathcal{O}^{(0)}\|_2 + \bar{\varepsilon}_{\mathcal{O}}$. Second, using

$$\begin{aligned}
& \|A \otimes B - C \otimes D\|_2 \\
= \; & \|A \otimes B - A \otimes D + A \otimes D - C \otimes D\|_2 \\
\leq \; & \|A\|_2 \|B - D\|_2 + \|A - C\|_2 \|D\|_2, \qquad (43)
\end{aligned}$$

we have

$$\begin{aligned}
& \|\bar{M}^{\mathrm{in}} - M^{\mathrm{in}(0)\otimes n}\|_2 \\
\leq \; & \bar{\varepsilon}_{\mathrm{in}} \sum_{h=1}^{n} \|M^{\mathrm{in}(0)}\|_2^{n-h} \prod_{m=1}^{h-1} \|\bar{M}_m^{\mathrm{in}}\|_2 \\
\leq \; & \bar{\varepsilon}_{\mathrm{in}} \sum_{h=1}^{n} \|M^{\mathrm{in}(0)}\|_2^{n-h} (\|M^{\mathrm{in}(0)}\|_2 + \bar{\varepsilon}_{\mathrm{in}})^{h-1} \\
= \; & \bar{\varepsilon}_{\mathrm{in}}^{(n)}. \qquad (44)
\end{aligned}$$

Third, using the inequality (41), we have

$$\|\bar{M}^{\mathrm{in}-1}\|_2 \leq \frac{1}{[s_{\min}(M^{\mathrm{in}(0)})]^n - \bar{\varepsilon}_{\mathrm{in}}^{(n)}}. \qquad (45)$$

We remark that for a d-dimensional matrix M, $\|M\|_{\max} \leq \|M\|_2 \leq d\|M\|_{\max}$.

B Error models

We consider a quantum computer with the following operations. The initialisation $\mathcal{I}^{(0)} = [\pi] + [\pi\sigma^{\mathrm{x}}]$, which prepares the state $|0\rangle$. The projective measurement $[\pi]$. Single-qubit Clifford gates $[R_{\mathrm{x}}]$ and $[R_{\mathrm{z}}]$. The single-qubit non-Clifford gate $[T]$, where $T = I\cos\frac{\pi}{8} + i\sigma^{\mathrm{x}}\sin\frac{\pi}{8}$. Two-qubit maximally entangling gate $[\Lambda]$, where $\Lambda = \frac{1}{\sqrt{2}}(I + i\sigma^{\mathrm{z}} \otimes \sigma^{\mathrm{z}})$, which is equivalent to the controlled-NOT gate and controlled-phase gate up to single-qubit gates. The sixteen basis operations can be realised as shown in Table 8.1. In order to perform GST, we choose initial states and observables as in section 8.8, and we choose the invertible matrix $T = M^{\mathrm{in}(0)\otimes n}$ for n qubits.

For the initialisation, the state prepared is ρ_0 rather than $|0\rangle\langle 0|$. We can always express the initialisation operation with error as $\mathcal{I} = \mathcal{N}_{\mathrm{i}}\mathcal{I}$, where $\mathcal{N}_{\mathrm{i}}(|0\rangle\langle 0|) = \rho_0$.

A POVM is defined by a set of operators $\{E_k\}$ satisfying $\sum_k E_k^\dagger E_k = I$. In a POVM, the state is mapped to $E_k \rho E_k^\dagger$ when the outcome is k. When the measurement has error, we may not be able to obtain all the information k. Usually there are only two outcomes corresponding to $|0\rangle$ and $|1\rangle$, respectively. In this case, maybe several k values correspond to the same outcome $\nu = 0, 1$. Therefore, we model the projective measurement $[\pi]$ with error as $\mathcal{M}\rho = \sum_{k\in K_0} E_k \rho E_k^\dagger$, where K_ν is the set of k corresponding to the measurement outcome ν.

For a gate without error $\mathcal{G}^{(0)}$, the gate with error can be expressed as $\mathcal{G} = \mathcal{N}_{\mathrm{a}}\mathcal{G}^{(0)}\mathcal{N}_{\mathrm{b}}$. Any noisy gate can be expressed in this form: Because $\mathcal{G}^{(0)}$ is invertible, we can always take $\mathcal{N}_{\mathrm{b}} = [I]$ and $\mathcal{N}_{\mathrm{a}} = \mathcal{G}\mathcal{G}^{(0)-1}$.

We suppose that time costs of the measurement $[\pi]$ and the two-qubit gate $[\Lambda]$ are the same, and time costs of single qubit gates are negligible.

We distinguish the identity operation and the memory operation. Without error, both of them are the same operation $[I]$. In any case, the identity operation is $[I]$, which means that the next operation is performed immediately, so it takes no time and there is not any memory error. When the memory operation is performed, the

qubit waits for the next operation, so memory errors may occur on it. We apply the identity operation for measuring the matrix g. In the basis operation set, the operation $[I]$ is replaced by the memory operation.

We set the cycle time of the computing as the time cost of the measurement and the two-qubit gate. In one cycle, only one operation is performed on a qubit. If the operation is a single-qubit gate, the gate is performed at the middle of the cycle, i.e. the overall operation is $\mathcal{N}_{\rm m}\mathcal{G}\mathcal{N}_{\rm m}$, where $\mathcal{N}_{\rm m}$ denotes memory noise. If no gate or measurement is performed in the cycle, the overall operation is $\mathcal{N}_{\rm m}^2$, which is the error version of the operation $[I]$ in the basis operation set.

We suppose that the single-qubit noise is described by the superoperator $\mathcal{E}^{(1)}(\epsilon)$, and the two-qubit noise is described by the superoperator $\mathcal{E}^{(2)}(\epsilon)$. Here, ϵ is a parameter describing the intensity of the noise. Then, the initialisation noise is $\mathcal{N}_{\rm i} = \mathcal{E}^{(1)}(\frac{\epsilon}{10})$, and the measurement with noise is $\tilde{M} = \mathcal{E}^{(1)}(\frac{\epsilon}{2})[\pi]\mathcal{E}^{(1)}(\frac{\epsilon}{2})$. For single-qubit gates, $\mathcal{N}_{\rm a} = \mathcal{N}_{\rm b} = \mathcal{E}^{(1)}(\frac{\epsilon}{20})$. For the two-qubit gate, $\mathcal{N}_{\rm a} = \mathcal{N}_{\rm b} = \mathcal{E}^{(2)}(\frac{\epsilon}{2})$. For the memory operation, $\mathcal{N}_{\rm m} = \mathcal{E}^{(1)}(\frac{\epsilon}{200})$.

B.1 Depolarising Error

The single-qubit depolarising noise is

$$\mathcal{E}^{(1)}(\epsilon) = (1 - \frac{4\epsilon}{3})[I] + \frac{\epsilon}{3}\sum_{\alpha=0}^{3}[\sigma^{\alpha}], \tag{46}$$

where σ^0, σ^1, σ^2 and σ^3 correspond to I, $\sigma^{\rm x}$, $\sigma^{\rm y}$ and $\sigma^{\rm z}$, respectively. The two-qubit depolarising noise is

$$\mathcal{E}^{(2)}(\epsilon) = (1 - \frac{16\epsilon}{15})[I \otimes I] + \frac{\epsilon}{15}\sum_{\alpha,\beta=0}^{3}[\sigma^{\alpha} \otimes \sigma^{\beta}]. \tag{47}$$

The x-axis (error rate) in Fig. 8.3(a) is ϵ of the two-qubit gate.

B.2 Dephasing Error

The single-qubit dephasing noise is

$$\mathcal{E}^{(1)}(\epsilon) = (1-\epsilon)[I] + \epsilon[\sigma^{\mathrm{z}}]. \tag{48}$$

The two-qubit dephasing noise is

$$\begin{aligned}\mathcal{E}^{(2)}(\epsilon) &= (1-\epsilon)[I\otimes I] + \frac{\epsilon}{3}([I\otimes\sigma^{\mathrm{z}}] \\ &\quad +[\sigma^{\mathrm{z}}\otimes I] + [\sigma^{\mathrm{z}}\otimes\sigma^{\mathrm{z}}]).\end{aligned} \tag{49}$$

B.3 Damping Error

The single-qubit damping noise is

$$\begin{aligned}\mathcal{E}^{(1)}(\epsilon) &= [\frac{I+\sigma^{\mathrm{z}}}{2} + \sqrt{1-\epsilon}\frac{I-\sigma^{\mathrm{z}}}{2}] \\ &\quad +[\sqrt{\epsilon}\frac{\sigma^{\mathrm{x}}+i\sigma^{\mathrm{y}}}{2}].\end{aligned} \tag{50}$$

The two-qubit damping noise is

$$\mathcal{E}^{(2)}(\epsilon) = \mathcal{E}^{(1)}(\frac{\epsilon}{2})\otimes\mathcal{E}^{(1)}(\frac{\epsilon}{2}). \tag{51}$$

B.4 Over-rotation error

Noise is gate dependent. Initialisation, measurement and memory operation are perfect, i.e. $\mathcal{E}^{(1)} = [I]$ for these operations. Only gates have noise. For gate R_{x}, $\mathcal{E}^{(1)}(\epsilon) = [I\cos\frac{\epsilon\pi}{4} + i\sigma^{\mathrm{x}}\sin\frac{\epsilon\pi}{4}]$. For gate R_{x}, $\mathcal{E}^{(1)}(\epsilon) = [I\cos\frac{\epsilon\pi}{4} + i\sigma^{\mathrm{z}}\sin\frac{\epsilon\pi}{4}]$. For gate T, $\mathcal{E}^{(1)}(\epsilon) = [I\cos\frac{\epsilon\pi}{8} + i\sigma^{\mathrm{z}}\sin\frac{\epsilon\pi}{8}]$. For gate Λ, $\mathcal{E}^{(2)}(\epsilon) = [I\otimes I\cos\frac{\epsilon\pi}{4} + i\sigma^{\mathrm{z}}\otimes\sigma^{\mathrm{z}}\sin\frac{\epsilon\pi}{4}]$. The x-axis (over rotation) in Fig. 8.3(a) is ϵ of the two-qubit gate.

B.5 Random-field error

Noise is gate dependent. For each operation, the noise $\mathcal{E}^{(1,2)}(\epsilon) = [e^{-i\epsilon\pi H}]$ is determined by a Hamiltonian. Here, $H = (h + h^\dagger)/2$, and each element of h is randomly generated with a uniform distribution in the unit circle. We remark that the noise is time independent, i.e. the noise is the same for the same gate implemented at different times.

B.6 Random-operation error

The operation without noise is $\mathcal{G}^{(0)}$. The operation with noise is $\mathcal{G}(\epsilon)$, which depends on the error parameter. As the same as other models, the error parameter ϵ is operation dependent. Each operation can be expressed using a χ-matrix [31]. We suppose the χ-matrix corresponding to $\mathcal{G}(\epsilon)$ is χ, and the χ-matrix corresponding to $\mathcal{G}^{(0)}$ is $\chi^{(0)}$. To generate χ, firstly we generate a Hermitian matrix around $\chi^{(0)}$, which is $\chi' = \chi^{(0)} + \epsilon H$, where H is generate as the same as the random Hamiltonian. Second, if $\mathcal{G}^{(0)}$ is not measurement, $\mathcal{G}(\epsilon)$ should be trace preserving. However, χ' may correspond to a non-trace preserving operation. In this case, we project χ' to the subspace in the matrix space that corresponds to trace preserving operations, i.e. χ'' is the matrix closest to χ' and corresponds to a trace preserving operation. If $\mathcal{G}^{(0)}$ is measurement, $\chi'' = \chi'$. Third, χ'' may not be positive semi-definite. Therefore, we take $\chi''' = \chi'' + \lambda_{\min} I$ if the minimum eigenvalue $\lambda_{\min}$ of χ'' is negative; otherwise $\chi''' = \chi''$. Finally, $\chi = f\chi'''$, where the factor f makes sure that the operation is still trace preserving and the maximum eigenvalue of χ is smaller than 1.

C Error component of Pauli error and leakage error

Taking $\epsilon = p_{\rm x} + p_{\rm y} + p_{\rm z}$, it is obvious that in the inhomogeneous Pauli error model, the error component does not depend on the error rate, i.e. $\mathcal{E}' = \epsilon^{-1}(p_{\rm x}[\sigma^{\rm x}] + p_{\rm y}[\sigma^{\rm y}] + p_{\rm z}[\sigma^{\rm z}])$. We remark that ratios p_α/ϵ do not change with ϵ.

For the leakage error model, we take $\epsilon = p$. Then the error component is

$$\begin{aligned} \mathcal{E}'(\rho) &= \pi_0\rho\pi_0 + \frac{\sqrt{1-p}-(1-p)}{p}(\pi_0\rho\pi_1 + \pi_1\rho\pi_0) \\ &= \pi_0\rho\pi_0 + [1/2 + \mathcal{O}(p)](\pi_0\rho\pi_1 + \pi_1\rho\pi_0), \end{aligned} \tag{52}$$

where $\pi_0 = |0\rangle\langle 0|$ and $\pi_1 = |1\rangle\langle 1|$. Therefore, $\mathcal{E}'$ varies slowly with p.

Bibliography

[1] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2):303–332, 1999.

[2] Seth Lloyd. Universal quantum simulators. *Science*, 273:1073–1078, Aug 1996.

[3] Alán Aspuru-Guzik, Anthony D Dutoi, Peter J Love, and Martin Head-Gordon. Simulated quantum computation of molecular energies. *Science*, 309(5741):1704–1707, 2005.

[4] Benjamin P Lanyon, James D Whitfield, Geoff G Gillett, Michael E Goggin, Marcelo P Almeida, Ivan Kassal, Jacob D Biamonte, Masoud Mohseni, Ben J Powell, Marco Barbieri, et al. Towards quantum chemistry on a quantum computer. *Nat. Chem.*, 2(2):106, 2010.

[5] James D Whitfield, Jacob Biamonte, and Alán Aspuru-Guzik. Simulation of electronic structure hamiltonians using quantum computers. *Molecular Physics*, 109(5):735–750, 2011.

[6] William G Unruh. Maintaining coherence in quantum computers. *Phys. Rev. A*, 51(2):992, 1995.

[7] Peter W Shor. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A*, 52(4):R2493, 1995.

[8] Andrew M Steane. Error correcting codes in quantum theory. *Phys. Rev. Lett.*, 77(5):793, 1996.

[9] A Robert Calderbank and Peter W Shor. Good quantum error-correcting codes exist. *Phys. Rev. A*, 54(2):1098, 1996.

[10] Raymond Laflamme, Cesar Miquel, Juan Pablo Paz, and Wojciech Hubert Zurek. Perfect quantum error correcting code. *Phys. Rev. Lett*, 77(1):198, 1996.

[11] Charles H Bennett, David P DiVincenzo, John A Smolin, and William K Wootters. Mixed-state entanglement and quantum error correction. *Phys. Rev. A*, 54(5):3824, 1996.

[12] David P DiVincenzo and Peter W Shor. Fault-tolerant error correction with efficient quantum codes. *Phys. Rev. Lett*, 77(15):3260, 1996.

[13] Daniel Gottesman. Theory of fault-tolerant quantum computation. *Phys. Rev. A*, 57(1):127, 1998.

[14] Rami Barends, Julian Kelly, Anthony Megrant, Andrzej Veitia, Daniel Sank, Evan Jeffrey, Ted C White, Josh Mutus, Austin G Fowler, Brooks Campbell, et al. Superconducting quantum circuits at the surface code threshold for fault tolerance. *Nature*, 508(7497):500, 2014.

[15] T. P. Harty, D. T. C. Allcock, C. J. Ballance, L. Guidoni, H. A. Janacek, N. M. Linke, D. N. Stacey, and D. M. Lucas. High-fidelity preparation, gates, memory, and readout of a trapped-ion quantum bit. *Phys. Rev. Lett.*, 113:220501, Nov 2014.

[16] 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.*, 117:060504, Aug 2016.

[17] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. Surface codes: Towards practical large-scale quantum computation. *Phys. Rev. A*, 86:032324, Sep 2012.

[18] Joe O'Gorman and Earl T Campbell. Quantum computation with realistic magic-state factories. *Phys. Rev. A*, 95(3):032338, 2017.

[19] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J Love, Alán Aspuru-Guzik, and Jeremy L O'brien. A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.*, 5:4213, 2014.

[20] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M Chow, and Jay M Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242, 2017.

[21] Nikolaj Moll, Panagiotis Barkoutsos, Lev S Bishop, Jerry M Chow, Andrew Cross, Daniel J Egger, Stefan Filipp, Andreas Fuhrer, Jay M Gambetta, Marc Ganzhorn, et al. Quantum optimization using variational algorithms on near-term quantum devices. *Quantum Sci. Technol.*, 3(3):030503, 2018.

[22] Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. The theory of variational hybrid quantum-classical algorithms. *New J. Phys.*, 18(2):023023, 2016.

[23] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

[24] Ying Li and Simon C Benjamin. Efficient variational quantum simulator incorporating active error minimization. *Phys. Rev. X*, 7(2):021050, 2017.

[25] Tyson Jones, Suguru Endo, Sam McArdle, Xiao Yuan, and Simon C Benjamin. Variational quantum algorithms for discovering hamiltonian spectra. *Phys. Rev. A*, 99(6):062304, 2019.

[26] Oscar Higgott, Daochen Wang, and Stephen Brierley. Variational quantum computation of excited states. *arXiv preprint arXiv:1805.08138*, 2018.

[27] Suguru Endo, Ying Li, Simon Benjamin, and Xiao Yuan. Variational quantum simulation of general processes. *arXiv preprint arXiv:1812.08778*, 2018.

[28] Kristan Temme, Sergey Bravyi, and Jay M. Gambetta. Error mitigation for short-depth quantum circuits. *Phys. Rev. Lett.*, 119:180509, Nov 2017.

[29] Suguru Endo, Simon C Benjamin, and Ying Li. Practical quantum error mitigation for near-future applications. *Phys. Rev. X*, 8(3):031027, 2018.

[30] Suguru Endo, Qi Zhao, Ying Li, Simon Benjamin, and Xiao Yuan. Mitigating algorithmic errors in a hamiltonian simulation. *Phys. Rev. A*, 99(1):012334, 2019.

[31] Michael A Nielsen and Isaac Chuang. Quantum computation and quantum information, 2002.

[32] Albert Einstein, Boris Podolsky, and Nathan Rosen. Can quantum-mechanical description of physical reality be considered complete? *Phys. Rev.*, 47(10):777, 1935.

[33] Goran Lindblad. On the generators of quantum dynamical semigroups. *Commun. Math. Phys.*, 48(2):119–130, 1976.

[34] Crispin Gardiner, Peter Zoller, and Peter Zoller. *Quantum noise: a handbook of Markovian and non-Markovian quantum stochastic methods with applications to quantum optics*, volume 56. Springer Science & Business Media, 2004.

[35] Dorit Aharonov, Wim Van Dam, Julia Kempe, Zeph Landau, Seth Lloyd, and Oded Regev. Adiabatic quantum computation is equivalent to standard quantum computation. *SIAM Rev.*, 50(4):755–787, 2008.

[36] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution. *arXiv preprint quant-ph/0001106*, 2000.

[37] Robert Raussendorf, Daniel E Browne, and Hans J Briegel. Measurement-based quantum computation on cluster states. *Phys. Rev. A*, 68(2):022312, 2003.

[38] Richard Jozsa. An introduction to measurement based quantum computation. *NATO Sc. S Ss. III C. S.*, 199:137–158, 2006.

[39] Robert Raussendorf and Hans J Briegel. A one-way quantum computer. *Physical Review Letters*, 86(22):5188, 2001.

[40] Samuel L. Braunstein and Peter van Loock. Quantum information with continuous variables. *Rev. Mod. Phys.*, 77:513–577, Jun 2005.

[41] Seth Lloyd and Samuel L. Braunstein. Quantum computation over continuous variables. *Phys. Rev. Lett.*, 82:1784–1787, Feb 1999.

[42] David P. DiVincenzo. Two-bit gates are universal for quantum computation. *Phys. Rev. A*, 51:1015–1022, Feb 1995.

[43] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM J. Comput*, 26(5):1411–1473, 1997.

[44] James W Cooley and John W Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301, 1965.

[45] A Yu Kitaev. Quantum measurements and the abelian stabilizer problem. *arXiv preprint quant-ph/9511026*, 1995.

[46] Sam McArdle, Suguru Endo, Tyson Jones, Ying Li, Simon Benjamin, and Xiao Yuan. Variational quantum simulation of imaginary time evolution with applications in chemistry and beyond. *arXiv preprint arXiv:1804.03023*, 2018.

[47] Richard P Feynman. Simulating physics with computers. *Int. J. Theor. Phys.*, 21(6):467–488, 1982.

[48] Emanuel Knill and Raymond Laflamme. Concatenated quantum codes. *arXiv preprint quant-ph/9608012*, 1996.

[49] Daniel Gottesman. Stabilizer codes and quantum error correction. *arXiv preprint quant-ph/9705052*, 1997.

[50] A Yu Kitaev. Quantum computations: algorithms and error correction. *Russ. Math. Surv+*, 52(6):1191–1249, 1997.

[51] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Int. J. Theor. Phys.*, 43(9):4452–4505, 2002.

[52] Hector Bombin and Miguel Angel Martin-Delgado. Topological quantum distillation. *Phys. Rev. Lett.*, 97(18):180501, 2006.

[53] Matthew McKague. Interactive proofs for bqp via self-tested graph states. *arXiv preprint arXiv:1309.5675*, 2013.

[54] Tomoyuki Morimae, Keisuke Fujii, and Harumichi Nishimura. Quantum merlin-arthur with noisy channel. *arXiv preprint arXiv:1608.04829*, 2016.

[55] M-H Yung, Jorge Casanova, Antonio Mezzacapo, Jarrod Mcclean, Lucas Lamata, Alan Aspuru-Guzik, and Enrique Solano. From transistor to trapped-ion computers for quantum chemistry. *Sci. Rep-UK*, 4:3589, 2014.

[56] Jonathan Romero, Ryan Babbush, Jarrod R McClean, Cornelius Hempel, Peter J Love, and Alán Aspuru-Guzik. Strategies for quantum computing

molecular energies using the unitary coupled cluster ansatz. *Quantum Sci. Technol.*, 4(1):014008, 2018.

[57] Sam McArdle, Suguru Endo, Alan Aspuru-Guzik, Simon Benjamin, and Xiao Yuan. Quantum computational chemistry. *arXiv preprint arXiv:1808.10402*, 2018.

[58] Trygve Helgaker, Poul Jorgensen, and Jeppe Olsen. *Molecular electronic-structure theory*. John Wiley & Sons, 2014.

[59] Attila Szabo and Neil S Ostlund. *Modern quantum chemistry: introduction to advanced electronic structure theory*. Courier Corporation, 2012.

[60] Jacob T Seeley, Martin J Richard, and Peter J Love. The bravyi-kitaev transformation for quantum computation of electronic structure. *J. Chem. Phys.*, 137(22):224109, 2012.

[61] Sergey B Bravyi and Alexei Yu Kitaev. Fermionic quantum computation. *Ann. Phys-New York*, 298(1):210–226, 2002.

[62] J. I. Colless, V. V. Ramasesh, D. Dahlen, M. S. Blok, M. E. Kimchi-Schwartz, J. R. McClean, J. Carter, W. A. de Jong, and I. Siddiqi. Computation of molecular spectra on a quantum processor with an error-resilient algorithm. *Phys. Rev. X*, 8:011021, Feb 2018.

[63] Marc Ganzhorn, Daniel J Egger, P Barkoutsos, Pauline Ollitrault, Gian Salis, Nikolaj Moll, M Roth, A Fuhrer, P Mueller, S Woerner, et al. Gate-efficient simulation of molecular eigenstates on a quantum computer. *Phys. Rev. Appl.*, 11(4):044092, 2019.

[64] Cornelius Hempel, Christine Maier, Jonathan Romero, Jarrod McClean, Thomas Monz, Heng Shen, Petar Jurcevic, Ben P Lanyon, Peter Love, Ryan

Babbush, et al. Quantum chemistry calculations on a trapped-ion quantum simulator. *Phys. Rev. X*, 8(3):031022, 2018.

[65] P. J. J. O'Malley, R. Babbush, I. D. Kivlichan, J. Romero, J. R. McClean, R. Barends, J. Kelly, P. Roushan, A. Tranter, N. Ding, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, A. G. Fowler, E. Jeffrey, E. Lucero, A. Megrant, J. Y. Mutus, M. Neeley, C. Neill, C. Quintana, D. Sank, A. Vainsencher, J. Wenner, T. C. White, P. V. Coveney, P. J. Love, H. Neven, A. Aspuru-Guzik, and J. M. Martinis. Scalable quantum simulation of molecular energies. *Phys. Rev. X*, 6:031007, Jul 2016.

[66] Yangchao Shen, Xiang Zhang, Shuaining Zhang, Jing-Ning Zhang, Man-Hong Yung, and Kihwan Kim. Quantum implementation of the unitary coupled cluster for simulating molecular electronic structure. *Phys. Rev. A*, 95:020501, Feb 2017.

[67] Masuo Suzuki. General theory of fractal path integrals with applications to many-body theories and statistical physics. *Jo. Mat. Phys.*, 32(2):400–407, 1991.

[68] Andrew M Childs, Aaron Ostrander, and Yuan Su. Faster quantum simulation by randomization. *arXiv preprint arXiv:1805.08385*, 2018.

[69] Guang Hao Low and Isaac L Chuang. Hamiltonian simulation by qubitization. *arXiv preprint arXiv:1610.06546*, 2016.

[70] P. A. M. Dirac. Note on exchange phenomena in the thomas atom. *Mathematical Proceedings of the Cambridge Philosophical Society*, 26(3):376–385, 1930.

[71] Âkov Il'ič Frenkel. Wave mechanics; advanced general theory. *Bull. Amer. Math. Soc*, 41:776, 1935.

[72] A. D McLachlan. A variational solution of the time-dependent schrodinger equation. *Mol. Phys.*, 8(1):39–44, 1964.

[73] P. H Kramer and Marcos Saraceno. *Geometry of the time-dependent variational principle in quantum mechanics*. Springer, 1981.

[74] J Broeckhove, L Lathouwers, E Kesteloot, and P Van Leuven. On the equivalence of time-dependent variational principles. *Chem. Phys. Lett.*, 149(5-6):547–550, 1988.

[75] Xiao Yuan, Suguru Endo, Qi Zhao, Simon Benjamin, and Ying Li. Theory of variational quantum simulation. *arXiv preprint arXiv:1812.08767*, 2018.

[76] G. C. Wick. Properties of bethe-salpeter wave functions. *Phys. Rev.*, 96:1124–1134, Nov 1954.

[77] Lewis F Richardson, BA J Arthur Gaunt, et al. Viii. the deferred approach to the limit. *Phil. Trans. R. Soc. Lond. A*, 226(636-646):299–361, 1927.

[78] Xiao Yuan, Lütkenhaus Norbert Zhang, Zhen, and Xiongfeng Ma. Simulating single photons with realistic photon sources. *Phys. Rev. A*, 94:062305, Dec 2016.

[79] Abhinav Kandala, Kristan Temme, Antonio D Córcoles, Antonio Mezzacapo, Jerry M Chow, and Jay M Gambetta. Error mitigation extends the computational reach of a noisy quantum processor. *Nature*, 567(7749):491, 2019.

[80] Jarrod R. McClean, Mollie E. Kimchi-Schwartz, Jonathan Carter, and Wibe A. de Jong. Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states. *Phys. Rev. A*, 95:042308, Apr 2017.

[81] Sam McArdle, Xiao Yuan, and Simon Benjamin. Error-mitigated digital quantum simulation. *Phy. Rev. Lett.*, 122(18):180501, 2019.

[82] X Bonet-Monroig, R Sagastizabal, M Singh, and TE O'Brien. Low-cost error mitigation by symmetry verification. *Phys. Rev. A*, 98(6):062339, 2018.

[83] Ramiro Sagastizabal, Xavier Bonet-Monroig, Malay Singh, MA Rol, CC Bultink, X Fu, CH Price, VP Ostroukh, N Muthusubramanian, A Bruno, et al. Error mitigation by symmetry verification on a variational quantum eigensolver. *arXiv preprint arXiv:1902.11258*, 2019.

[84] Matthew Otten and Stephen K Gray. Accounting for errors in quantum algorithms via individual error reduction. *Npj Quantum Inf.*, 5(1):11, 2019.

[85] Richard N Zare and Dudley R Herschbach. Doppler line shape of atomic fluorescence excited by molecular photodissociation. *Proceedings of the IEEE*, 51(1):173–182, 1963.

[86] Thomas W Ebbesen, Katsumi Tanigaki, and Sadanori Kuroshima. Excited-state properties of c60. *Chem. Phys. Lett.*, 181(6):501–504, 1991.

[87] GA Chamberlain. Organic solar cells: A review. *Solar cells*, 8(1):47–83, 1983.

[88] Michael Grätzel. Dye-sensitized solar cells. *Journal of photochemistry and photobiology C: Photochemistry Reviews*, 4(2):145–153, 2003.

[89] George Blasse and BC Grabmaier. How does a luminescent material absorb its excitation energy? In *Luminescent materials*, pages 10–32. Springer, 1994.

[90] Raffaele Santagati, Jianwei Wang, Antonio A Gentile, Stefano Paesani, Nathan Wiebe, Jarrod R McClean, Sam Morley-Short, Peter J Shadbolt, Damien Bonneau, Joshua W Silverstone, et al. Witnessing eigenstates for quantum simulation of hamiltonian spectra. *Sci. Adv.*, 4(1):eaap9646, 2018.

[91] L. Cincio, Y.Suba, A. T. Sornborger, and Patrick J. Coles. Learning the quantum algorithm for state overlap. *arXiv preprint arXiv: 1803.04114*, 2018.

[92] Juan Carlos, Garcia-Escartin, and Pedro Chamorro-Posada. Swap test and hong-ou-mandel effect are equivalent. *Phys. Rev. A*, 87(052330), 2013.

[93] Gene H Golub, Per Christian Hansen, and Dianne P O'Leary. Tikhonov regularization and total least squares. *SIAM J MATRIX ANAL A*, 21(1):185–194, 1999.

[94] Artur K. Ekert, Carolina Moura Alves, Daniel K. L. Oi, Michał Horodecki, Paweł Horodecki, and L. C. Kwek. Direct estimations of linear and nonlinear functionals of a quantum state. *Phys. Rev. Lett.*, 88:217901, May 2002.

[95] Pierre-Luc Dallaire-Demers, Jonathan Romero, Libor Veis, Sukin Sim, and Alán Aspuru-Guzik. Low-depth circuit ansatz for preparing correlated fermionic states on a quantum computer. *arXiv preprint arXiv:1801.01053*, 2018.

[96] Richard M Karp. Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer, 1972.

[97] Ryan Babbush, Jarrod McClean, Ian Kivlichan, Damian Steiger, Kevin Sung, Yudong Cao, E Schuyler Fried, Craig Gidney, Thomas Haener, Vojtech Havlicek, et al. Openfermion: the electronic structure package for quantum computers. In *APS Meeting Abstracts*, 2018.

[98] Ken M Nakanishi, Kosuke Mitarai, and Keisuke Fujii. Subspace-search variational quantum eigensolver for excited states. *arXiv preprint arXiv:1810.09434*, 2018.

[99] Robert M. Parrish, Edward G. Hohenstein, Peter L. McMahon, and Todd J. Martínez. Quantum computation of electronic transitions using a variational quantum eigensolver. *Phys. Rev. Lett.*, 122:230401, Jun 2019.

[100] Jonathan Romero, Ryan Babbush, Jarrod R McClean, Cornelius Hempel, Peter Love, and Alán Aspuru-Guzik. Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz. *arXiv preprint arXiv:1701.02691*, 2017.

[101] Ya Wang, Florian Dolde, Jacob Biamonte, Ryan Babbush, Ville Bergholm, Sen Yang, Ingmar Jakobi, Philipp Neumann, Alán Aspuru-Guzik, James D Whitfield, et al. Quantum simulation of helium hydride cation in a solid-state spin register. *ACS nano*, 9(8):7769–7774, 2015.

[102] S. Paesani, A. A. Gentile, R. Santagati, J. Wang, N. Wiebe, D. P. Tew, J. L. O'Brien, and M. G. Thompson. Experimental bayesian quantum phase estimation on a silicon photonic chip. *Phys. Rev. Lett.*, 118:100503, Mar 2017.

[103] Kentaro Heya, Ken M Nakanishi, Kosuke Mitarai, and Keisuke Fujii. Subspace variational quantum simulator. *arXiv preprint arXiv:1904.08566*, 2019.

[104] Naomichi Hatano and David R. Nelson. Localization transitions in non-hermitian quantum mechanics. *Phys. Rev. Lett.*, 77:570–573, Jul 1996.

[105] Nimrod Moiseyev. *Non-Hermitian Quantum Mechanics*. Cambridge University Press, 2011.

[106] Fabio Bagarello, Jean-Pierre Gazeau, Franciszek Hugon Szafraniec, and Miloslav Znojil. *Non-selfadjoint operators in quantum physics: Mathematical aspects*. John Wiley & Sons, 2015.

[107] Hans C. Fogedby, Anders B. Eriksson, and Lev V. Mikheev. Continuum limit, galilean invariance, and solitons in the quantum equivalent of the noisy burgers equation. *Phys. Rev. Lett.*, 75:1883–1886, Sep 1995.

[108] Carl M Bender. Making sense of non-hermitian hamiltonians. *Rep. Prog. Phys.*, 70(6):947, 2007.

[109] Carl M Bender, Dorje C Brody, Hugh F Jones, and Bernhard K Meister. Faster than hermitian quantum mechanics. *Phys. Rev, Lett.*, 98(4):040403, 2007.

[110] Ingrid Rotter. A non-hermitian hamilton operator and the physics of open quantum systems. *J. Phys. A-Math Theor.*, 42(15):153001, 2009.

[111] Carl Edward Rasmussen. Gaussian processes in machine learning. In *Advanced lectures on machine learning*, pages 63–71. Springer, 2004.

[112] Nasser M Nasrabadi. Pattern recognition and machine learning. *J. Electron Imaging*, 16(4):049901, 2007.

[113] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103:150502, Oct 2009.

[114] A. Childs, R. Kothari, and R. Somma. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM J. Comput.*, 46(6):1920–1950, 2017.

[115] Andris Ambainis. Variable time amplitude amplification and quantum algorithms for linear algebra problems. In *STACS'12 (29th Symposium on Theoretical Aspects of Computer Science)*, volume 14, pages 636–647. LIPIcs, 2012.

[116] B. D. Clader, B. C. Jacobs, and C. R. Sprouse. Preconditioned quantum linear system algorithm. *Phys. Rev. Lett.*, 110:250504, Jun 2013.

[117] Leonard Wossnig, Zhikuan Zhao, and Anupam Prakash. Quantum linear system algorithm for dense matrices. *Phys. Rev. Lett.*, 120:050502, Jan 2018.

[118] Shantanav Chakraborty, András Gilyén, and Stacey Jeffery. The power of block-encoded matrix powers: improved regression techniques via faster hamiltonian simulation. *arXiv preprint arXiv:1804.01973*, 2018.

[119] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. *arXiv preprint arXiv:1806.01838*, 2018.

[120] Yi ğit Subaş ı, Rolando D. Somma, and Davide Orsucci. Quantum algorithms for systems of linear equations inspired by adiabatic quantum computing. *Phys. Rev. Lett.*, 122:060504, Feb 2019.

[121] Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. Quantum support vector machine for big data classification. *Phys. Rev. Lett.*, 113:130503, Sep 2014.

[122] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. Quantum principal component analysis. *Nat. Phys.*, 10(9):631, 2014.

[123] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195, 2017.

[124] Kosuke Mitarai, Makoto Negoro, Masahiro Kitagawa, and Keisuke Fujii. Quantum circuit learning. *Phys. Rev. A*, 98(3):032309, 2018.

[125] Heinz-Peter Breuer, Francesco Petruccione, et al. *The theory of open quantum systems*. Oxford University Press on Demand, 2002.

[126] HJ Carmichael. Quantum trajectory theory for cascaded open systems. *Phys. Rev. Lett.*, 70(15):2273, 1993.

[127] Hefeng Wang, Sahel Ashhab, and Franco Nori. Quantum algorithm for simulating the dynamics of an open quantum system. *Phys. Rev. A*, 83(6):062317, 2011.

[128] Dave Bacon, Andrew M Childs, Isaac L Chuang, Julia Kempe, Debbie W Leung, and Xinlan Zhou. Universal simulation of markovian quantum dynamics. *Phys. Rev. A*, 64(6):062302, 2001.

[129] Ryan Sweke, Ilya Sinayskiy, Denis Bernard, and Francesco Petruccione. Universal simulation of markovian open quantum systems. *Phys. Rev. A*, 91(6):062308, 2015.

[130] Ryan Sweke, Mikel Sanz, Ilya Sinayskiy, Francesco Petruccione, and Enrique Solano. Digital quantum simulation of many-body non-markovian dynamics. *Phys. Rev. A*, 94(2):022317, 2016.

[131] Richard Cleve and Chunhao Wang. Efficient quantum algorithms for simulating lindblad evolution. *arXiv preprint arXiv:1612.09512*, 2016.

[132] Andrew M Childs and Tongyang Li. Efficient simulation of sparse markovian quantum dynamics. *arXiv preprint arXiv:1611.05543*, 2016.

[133] Kosuke Mitarai and Keisuke Fujii. Methodology for replacing indirect measurements with direct measurements. *arXiv preprint arXiv:1901.00015*, 2018.

[134] Tony E Lee, Hartmut Haeffner, and MC Cross. Collective quantum jumps of rydberg atoms. *Phys. Rev. Lett.*, 108(2):023602, 2012.

[135] Cenap Ates, Beatriz Olmos, Juan P Garrahan, and Igor Lesanovsky. Dynamical phases and intermittency of the dissipative quantum ising model. *Phys. Rev. A*, 85(4):043620, 2012.

[136] Igor Lesanovsky, Merlijn van Horssen, Mădălin Guţă, and Juan P Garrahan. Characterization of dynamical phase transitions in quantum jump trajectories beyond the properties of the stationary state. *Phys. Rev. Lett.*, 110(15):150401, 2013.

[137] Dominic C Rose, Katarzyna Macieszczak, Igor Lesanovsky, and Juan P Garrahan. Metastability in an open quantum ising model. *Phys. Rev. E*, 94(5):052132, 2016.

[138] Dave Wecker, Matthew B. Hastings, and Matthias Troyer. Progress towards practical quantum variational algorithms. *Phys. Rev. A*, 92:042303, Oct 2015.

[139] Seth T Merkel, Jay M Gambetta, John A Smolin, Stefano Poletto, Antonio D Córcoles, Blake R Johnson, Colm A Ryan, and Matthias Steffen. Self-consistent quantum process tomography. *Phys. Rev. A*, 87(6):062119, 2013.

[140] Daniel Greenbaum. Introduction to quantum gate set tomography. *arXiv preprint arXiv:1509.02921*, 2015.

[141] Tyson Jones, Anna Brown, Ian Bush, and Simon Benjamin. Quest and high performance simulation of quantum computers, 2018.

[142] Bela Bauer, Dave Wecker, Andrew J Millis, Matthew B Hastings, and Matthias Troyer. Hybrid quantum-classical approach to correlated materials. *Phys. Rev. X*, 6(3):031045, 2016.

[143] Jonathan Romero, Jonathan P Olson, and Alan Aspuru-Guzik. Quantum autoencoders for efficient compression of quantum data. *Quantum Sci. Technol.*, 2(4):045001, 2017.

[144] Chao Song, Kai Xu, Wuxin Liu, Chui-ping Yang, Shi-Biao Zheng, Hui Deng, Qiwei Xie, Keqiang Huang, Qiujiang Guo, Libo Zhang, et al. 10-qubit entanglement and parallel logic operations with a superconducting circuit. *Phys. Rev. Lett.*, 119(18):180511, 2017.

[145] Mark Howard and Earl Campbell. Application of a resource theory for magic states to fault-tolerant quantum computing. *Phys. Rev. Lett.*, 118:090501, Mar 2017.

[146] Michael Grant and Stephen Boyd. Cvx: Matlab software for disciplined convex programming, version 2.1, 2014.

[147] Sergey Bravyi and David Gosset. Improved classical simulation of quantum circuits dominated by clifford gates. *Phys. Rev. Lett.*, 116(25):250501, 2016.

[148] Sergey Bravyi, Graeme Smith, and John A Smolin. Trading classical and quantum computational resources. *Phys. Rev. X*, 6(2):021043, 2016.

[149] Mark Howard and Earl Campbell. Application of a resource theory for magic states to fault-tolerant quantum computing. *Phys. Rev. Lett.*, 118(9):090501, 2017.

[150] E Knill. Fault-tolerant postselected quantum computation: Threshold analysis. *arXiv preprint quant-ph/0404104*, 2004.

[151] Joel J Wallman and Joseph Emerson. Noise tailoring for scalable quantum computation via randomized compiling. *Phys. Rev. A*, 94(5):052325, 2016.

[152] Dominic W. Berry, Andrew M. Childs, Richard Cleve, Robin Kothari, and Rolando D. Somma. Simulating hamiltonian dynamics with a truncated taylor series. *Phys. Rev. Lett.*, 114:090502, Mar 2015.

[153] Guang Hao Low and Isaac L. Chuang. Optimal hamiltonian simulation by quantum signal processing. *Phys. Rev. Lett.*, 118:010501, Jan 2017.

[154] Masuo Suzuki. Generalized trotter's formula and systematic approximants of exponential operators and inner derivations with applications to many-body problems. *Comm. Math. Phys.*, 51:183–190, 1976.

[155] Nathan Wiebe, Dominic Berry, Peter Høyer, and Barry C Sanders. Higher order decompositions of ordered operator exponentials. *J. Phys. A-Math. Theor.*, 43(6):065203, 2010.

[156] John Preskill. Quantum computing in the nisq era and beyond. *arXiv preprint arXiv:1801.00862*, 2018.

[157] George C. Knee and William J. Munro. Optimal trotterization in universal quantum simulators under faulty control. *Phys. Rev. A*, 91:052327, May 2015.

[158] Matthew Otten and Stephen K. Gray. Recovering noise-free quantum observables. *Phys. Rev. A*, 99:012338, Jan 2019.

[159] Christopher A Fuchs. Distinguishability and accessible information in quantum theory. *arXiv preprint quant-ph/9601020*, 1996.

[160] Alexei Gilchrist, Nathan K. Langford, and Michael A. Nielsen. Distance measures to compare real and ideal quantum processes. *Phys. Rev. A*, 71:062310, Jun 2005.

[161] Lorenzo Brualla, K Sakkos, Jordi Boronat, and J Casulleras. Higher order and infinite trotter-number extrapolations in path integral monte carlo. *J. Chem. Phys.*, 121(2):636–643, 2004.

[162] Masuo Suzuki. General correction theorems on decomposition formulae of exponential operators and extrapolation methods for quantum monte carlo simulations. *Phys. Lett. A*, 113(6):299–300, 1985.