Quantum algorithms for Hamiltonian learning problems

A thesis submitted in fulfilment of the requirements
for the degree of

Doctor of Philosophy
in
Computer Science (Quantum Computing)

by

Youle Wang

to

Centre for Quantum Software and Information, School of Computer Science
Faculty of Engineering and Information Technology
University of Technology Sydney
NSW - 2007, Australia

November 24, 2022

CERTIFICATE OF ORIGINAL AUTHORSHIP

I, Youle Wang, declare that this thesis is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the School of Computer Science, Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Signature: Production Note: Signature removed prior to publication.

Date: July 2022

Abstract

Learning the dynamic Hamiltonian of a quantum system is a fundamental task in studying condensed matter physics and verifying quantum technologies. Many protocols have been proposed for Hamiltonian learning. Still, many of them require the ability to prepare quantum Gibbs states and estimate entropies of quantum states, which are not easy for classical computers. Recent experimental progress of quantum hardware has drawn much attention, motivating us to investigate the application of near-term quantum computers in Hamiltonian learning. In this dissertation, we study the Hamiltonian learning problems and propose algorithms to recover interaction coefficients of a Hamiltonian, prepare quantum Gibbs states, and estimate the quantum entropies of quantum states. We employ quantum circuits that are expected to be implementable in the near-to-intermediate future.

First, we use the variational quantum algorithms to enable a hybrid quantum-classical algorithmic scheme to tackle the Hamiltonian learning problem. By transforming the Hamiltonian learning problem to an optimization problem using the Jaynes' principle, we employ a gradient-descent method to give the solution and could reveal the interaction coefficients from the system's Gibbs state measurement results. In particular, the computation of the gradients relies on the Hamiltonian spectrum and the log-partition function. Hence, as the main subroutine, we develop a variational quantum algorithm to extract the Hamiltonian spectrum and utilize convex optimization to compute the log-partition function. We also apply the importance sampling technique to circumvent the resource requirements for large-scale Hamiltonians.

Second, we propose variational quantum algorithms for quantum Gibbs state preparation. First, we take the loss function as the system's free energy and estimate it by a truncated version. Then we train a parameterized quantum circuit to optimize the loss function so that it can learn the desired quantum Gibbs state. Notably, our algorithms can be implemented on near-term quantum computers. Furthermore, by performing numerical experiments, we show that shallow parameterized circuits with only one additional qubit can be trained to prepare the Ising chain and spin chain Gibbs states with a fidelity higher than 95%. In particular, for the Ising chain model, we find that a simplified circuit ansatz with only one parameter and one additional qubit can be trained to realize a 99% fidelity in Gibbs state preparation at inverse temperatures larger than 2.

Third, we propose quantum algorithms to estimate the von Neumann and quantum α-Rényi entropies of an n-qubit quantum state ρ using independent copies of the input

state. We show how to efficiently construct the quantum circuits of both methods using primitive single/two-qubit gates. We prove that the number of required copies scales polynomially in $1/\epsilon$ and $1/\Lambda$, where ϵ denotes the additive precision and Λ denotes the lower bound on all non-zero eigenvalues. Notably, our method outperforms previous methods in the aspect of practicality since it does not require any quantum query oracles, which are usually necessary for previous methods. Furthermore, we conduct experiments to show the efficacy of our algorithms to single-qubit states and study the noise robustness.

Dedication

To my family.

STATEMENT OF AUTHORSHIP AND PUBLICATIONS

The dissertation is based on the following articles:

- Youle Wang, Guangxi Li, and Xin Wang. A hybrid quantum-classical Hamiltonian learning algorithm. Sci. China Inf. Sci. 66, 129502 (2023). https://doi.org/10.1007/s11432-021-3382-2
- Youle Wang, Guangxi Li, and Xin Wang. (2020). Variational quantum Gibbs state preparation with a truncated Taylor series. Physical Review Applied, 16(5), 054035. https://doi.org/10.1103/PhysRevApplied.16.054035
- Youle Wang, Benchi Zhao, and Xin Wang. (2022). Quantum algorithms for estimating quantum entropies. http://arxiv.org/abs/2203.02386

The first work above has been published in Science China Information Sciences and was also accepted as a short talk at the 21st Asian Quantum Information Science Conference, AQIS 2021. The second work has been published in Physical Review Applied. The third work is available online. In addition, I have co-authored the following articles that are not included in this dissertation. In the first work below, the authors are listed in alphabetical order.

- Xin Wang, Youle Wang, Zhan Yu, and Lei Zhang. Quantum Phase Processing: Transform and Extract Eigen-Information of Quantum Systems. http://arxiv.org/abs/2209.14278
- Xin Wang, Zhixin Song, and Youle Wang. (2021). Variational Quantum Singular Value Decomposition. Quantum, 5(1), 483. https://doi.org/10.22331/q-2021-06-29-483
- Guangxi Li, Youle Wang, Yu Luo, Yuan Feng, Quantum data fitting algorithm for non-sparse matrices. http://arxiv.org/abs/1907.06949

All my publications are available on arXiv and Google Scholar.

ACKNOWLEDGMENTS

This dissertation would not have been possible without the support I received during the doctoral study. I express my sincere gratitude and appreciation to all people who have helped me.

First, I want to thank my supervisor, professor Yuan Feng, who introduced me to quantum computing and taught me how to do research at the beginning of my research journey. In each meeting, he would listen carefully to my presentation and taught me how to make myself clear. His high research standards and profound knowledge in the area impressed me. I also sincerely thank him for giving me the freedom to communicate with other researchers and the support to visit overseas. In particular, I appreciate his support during the great pandemic of COVID-19.

I am very grateful to professor Sanjiang Li and professor Zhengfeng Ji for guiding my study. Sanjiang Li has provided much kind advice for me in research and life and is also very nice. Zhengfeng is a very expert in the area and has a very high taste for research. When I was wandering at the beginning of my study, Zhengfeng shared his experience in research to guide me through the time. And he would share his opinion about the research I focused on and push me to study further to deepen my understanding and do more meaningful research.

Moreover, I thank the partners and friends from Institute for Quantum Computing, Baidu, where I have had a great time and done much interesting research. I thank Institute for Quantum Computing, Baidu, for giving me the opportunity to join them as visiting student and research intern during my studies. I thank mentor Xin Wang for leading me to research projects during my internship. I am deeply inspired by his expertise in the area and the rigour he pursues in research, from which I can benefit a lot. I also appreciate him and Lijing Jin for hosting a birthday party.

During the visiting at Baidu, I had met many friends with diverse backgrounds. Their diverse research backgrounds and expertise have brought many meaningful discussions and collaborations. I cherish the opportunities to work Zhixin Song, Ranyiliu Chen, and Benchi Zhao and thank them for sharing their knowledge in different areas. I also have enjoyed in-depth discussions with many interns, including Xuanqiang Zhao, Zihe Wang, Xia Liu, Chengkai Zhu, Chenfeng Cao, Jiaqing Jiang, Sizhuo Yu, Kaiyan Shi, Zihan Xia, and many others not mentioned. I thank them for sharing their knowledge and perceptive opinions.

Finally, I thank my friends and colleagues at UTS: Ji Guan, Guangxi Li, Yu Luo, and Xiangzhen Zhou. When I first arrived in Sydney, Ji Guan helped me settle down. I

learned a lot from him about living in a foreign country. Guangxi Li is very expertise in computers, and he taught me a lot about programming. Yu Luo is very smart and patient. We often worked together to solve complex mathematical questions. Xiangzhen Zhou is a great friend and roommate. He also is a devout Christian. He often drove me to the Church on Sunday, where I learned about church and belief. We shared an apartment for one year, which was a great time, and I miss it. In particular, I thank my family for the endless love and support for allowing me to pursue my dreams.

Table of Contents

LIST OF FIGURES

FIGURE **Page**

List of Tables

Table **Page**

CHAPTER 1

INTRODUCTION

1.1 Hamiltonian learning

Learning the Hamiltonian dynamics is vital in studying quantum physics and realizing quantum computers in experiments. In the literature, it has been applied to predict the quantum system's locality that describes the effective interactions between particles, which plays a crucial role in quantum technology, such as quantum lattice models [162], quantum simulation [129], and adiabatic quantum computation [4]. With recent experimental advances in tools for studying complex interacting quantum systems [7], it is becoming more and more important to learn the dynamics of complicated physical systems, which can predict the evolution of any initial state governed by the Hamiltonian. Another critical utility is relevant to the verification of quantum devices and simulators towards building fault-tolerant quantum computers [89] since certifying that the engineered Hamiltonian matches the theoretically predicted models will always be an indispensable step in developing high-fidelity quantum gates [139].

In many-body physics, the system's Hamiltonian is often characterized by some parameters that describe the interactions between particles. Mathematically, the Hamiltonian can be decomposed into a linear combination of local Pauli matrices. To be more specific, let H denote the Hamiltonian, then

$$H = \sum_{\ell=1}^{m} \mu_\ell E_\ell, \tag{1.1}$$

where vector $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m) \in [-1, 1]^m$ consists of interaction parameters, and $\{E_\ell\}_{\ell=1}^m$

are n-qubit local Pauli operators that act non-trivially on at most fixed number particles, and integer $m = O(poly(n))$.

Despite the number of these parameters $\boldsymbol{\mu}$ in general scales polynomially in the system's size, it is pretty challenging to learn these parameters. Classically characterizing the system's Hamiltonian via tomography would require resources that exponentially scale in the system's size [63]. Other than tomography, there are methods [60, 144, 153–155] that cost polynomially many resources while requiring the ability to simulate the dynamics of the system, which is classically intractable. In particular, it is difficult to perform quantum simulation as a large amount of low-decoherence and fully-connected qubits are required, which are not available on noisy intermediate-scale quantum (NISQ) devices [118].

The main goal of our research is to employ a trusted NISQ devices to study complex quantum system. For this purpose, we exploit the variational quantum algorithms (VQAs) that have been gaining popularity in many areas [25, 32, 71, 90, 110, 114, 131, 146, 148, 158]. VQAs are a class of hybrid quantum-classical algorithms that are expected to be implementable on NISQ devices. The principal process is to optimize a certain loss function via parameterized quantum circuits (PQCs). In particular, the loss function depending on parameters of the circuit is evaluated on quantum devices, and then the parameters are updated using gradient-based methods classically.

In Chapter 3, we propose a hybrid quantum-classical algorithm to recover the interaction coefficients from the measurement results of Gibbs states. To this end, we take advantage of the strategy proposed recently in [5]. Specifically, let $e_\ell = \mathrm{tr}(\rho_\beta E_\ell)$, for all $\ell = 1, \ldots, m$, denote measurement results of a quantum Gibbs state $\rho_\beta = e^{-\beta H}/\mathrm{tr}(e^{-\beta H})$, where $\{E_\ell\}_{\ell=1}^m$ are given Pauli matrices. It has been shown that solving the optimization problem below suffices to complete the Hamiltonian learning task [5].

$$\boldsymbol{\mu} = \mathrm{argmin}_{\boldsymbol{\nu}} \log Z_\beta(\boldsymbol{\nu}) + \beta \sum_{\ell=1}^{m} \nu_\ell e_\ell. \tag{1.2}$$

Here, $Z_\beta(\boldsymbol{\nu}) = \mathrm{tr}(e^{-\beta \sum_{\ell=1}^m \nu_\ell E_\ell})$ denotes the partition function, parameterized by $\boldsymbol{\nu} = (\nu_1, ..., \nu_m) \in [-1, 1]^m$, and β denotes the inverse temperature of the system. Notably, the challenge of our approach is to compute the log-partition function $\log Z_\beta(\boldsymbol{\nu})$ and its gradient since computing partition function is #P-hard [57, 99]. Our idea is to solve the optimization problem in Eq. (1.2) by a gradient-descent method and compute the corresponding gradients utilizing variational quantum algorithms.

This Chapter is based on the following paper:
Youle Wang, Guangxi Li, and Xin Wang. A hybrid quantum-classical Hamiltonian

learning algorithm. Sci. China Inf. Sci. 66, 129502 (2023). https://doi.org/10.1007/s11432-021-3382-2

1.2 Quantum Gibbs state preparation

As shown in Sec. 1.1, quantum Gibbs states or thermal states are of significant importance for Hamiltonian learning. Thus, a subroutine for preparing Gibbs state of a given Hamiltonian is demanded. On top of that, quantum Gibbs states have many applications in quantum computing, which would boost the development of quantum algorithms. The reason is that quantum Gibbs states not only can be used to study many-body physics but also can be applied to quantum simulation [35], quantum machine learning [18, 85], and quantum optimization [133]. For example, sampling from well-prepared Gibbs states of Hamiltonians can be applied in solving combinatorial optimization problems [133], solving semi-definite programs [24], and training quantum Boltzmann machines [85].

In fact, the preparation of the desired quantum state is quite challenging. For instance, it is well-known that finding the ground state of a physical Hamiltonian is QMA-hard [151]. This is because, for Gibbs states, the preparation at arbitrary low temperature could be as hard as finding the ground state [3]. In the context, various methods are proposed to achieve this goal in classical and quantum computing [23, 82, 117, 137, 138]. Many quantum methods use quantum techniques, including quantum rejection sampling [156], quantum walk [161], dynamics simulation [80, 109, 124], dimension reduction [19]. Although in the worst case, the costs of these methods could be exponential in expectation, they can be efficient when some conditions are satisfied. Such as the ratios between the partition functions of the infinite temperature states and the Gibbs state is at most polynomially large [117], and the gap of the underlying Markov chain of the quantum walk is polynomially small [140, 161]. However, these methods require complex quantum subroutines such as quantum phase estimation, which are costly and hard to implement on near-term quantum computers.

In Chapter 4, we study how to prepare a high-fidelity Gibbs state and simultaneously minimize the task's quantum resources as much as possible, including qubit, gate counts, and circuit size. Minimizing resources is motivated by the implementation of near-term quantum computers. One feasible scheme is to take advantage of variational quantum algorithms (VQAs) [106] since VQAs are widely believed to be implementable on near-term quantum computers. This strategy succeeds in reducing the resources by using shallow quantum circuits. Notably, several methods for preparing Gibbs states using

VQAs have already been proposed [36, 74, 103, 104, 109, 142, 157, 160] to prepare Gibbs states. For instance, Wu and Hsieh [157] proposed a variational approach by using Rényi entropy estimation [74] and thermofield double states, Yuan et al. [160] discussed the application of imaginary time evolution to Gibbs state using parameterized circuits, and Chowdhury et al. [36] proposed entropy estimation method using tools such as quantum amplitude estimation and linear combination of unitaries. In contrast, we propose a variational Gibbs states preparation algorithm using shallow circuits in the thesis.

In this dissertation, we propose variational quantum algorithms by minimizing the system's free energy [122] based on the truncated Taylor series of the free energy. This is because Gibbs state minimizes the free energy. In other words, assuming the system's state is ρ, the free energy is defined by $\mathcal{F}(\rho) = \mathrm{tr}(H\rho) - \beta^{-1}S(\rho)$, where $S(\rho)$ denotes the von Neumann entropy. Then Gibbs state, denoted by ρ_G, is the global minimum of the functional $\mathcal{F}(\rho)$. Our approach uses parameterized quantum circuits (PQCs) with enough expressiveness to prepare the desired quantum Gibbs state or a very close state. For convenience, we denote the generated state via PQC by $\rho(\boldsymbol{\theta})$. Then the variational principle could be formulated as

$$\rho_G \approx \mathrm{argmin}_{\boldsymbol{\theta}} \mathcal{F}(\rho(\boldsymbol{\theta})). \tag{1.3}$$

Hence, our methods focus on finding the optimal parameters to minimize $\mathcal{F}(\rho(\boldsymbol{\theta}))$.

This Chapter is based on the following paper:
Youle Wang, Guangxi Li, and Xin Wang. (2020). Variational quantum Gibbs state preparation with a truncated Taylor series. Physical Review Applied, 16(5), 054035. https://doi.org/10.1103/PhysRevApplied.16.054035

1.3 Quantum entropy estimation

When estimating the system's free energy, it is essential to estimate the von Neumann entropy. In this dissertation, we consider the quantum entropy estimation problem to complete our Hamiltoninan learning method. In Chapter 5, we propose concrete quantum algorithms for von Neumann entropy and more general quantum Rényi entropy.

Entropy [10] is an important concept that characterizes the system's randomness and lead to many theoretical and practical applications in many fields including computer science and quantum physics. The classical Shannon entropy [130] and the Rényi entropies [123] are fundamental in information theory as they capture the operational quantities of information processing. The classical entropy depicts information by measuring the

uncertainty associated with a classical probability distribution. In the quantum setting, the corresponding concepts are the von Neumann [143] and quantum Rényi entropies [116], which are crucial to develop many areas such as quantum information theory [15], entanglement theory [70], and quantum chemistry [8]. Recently, many works apply quantum entropies for quantum computing applications. For instance, quantum entropies can provide the asymptotic lower bound for compressing quantum data [128].

Given a quantum state $\rho \in \mathbb{C}^{2^n \times 2^n}$, the von Neumann entropy is defined by $S(\rho) = -\mathrm{tr}(\rho \ln(\rho))$, and the quantum α-Rényi entropy is defined by $R_\alpha(\rho) = \frac{1}{1-\alpha} \log \mathrm{tr}(\rho^\alpha)$ with parameter $\alpha \in (0,1) \cup (1, +\infty)$. Taking the limit $\alpha \to 1$, $R_\alpha(\rho)$ converges to $S(\rho)$ up to a proportional factor. Additionally, if ρ is diagonal in the computational basis, $S(\rho)$ and $R_\alpha(\rho)$ degenerate to their classical counterparts. In past decades, various methods [108] have been proposed to estimate quantum entropies, while a large number of quantum resources are demanded as well. The most straightforward method to estimate quantum entropy is the tomography [2], which figures out the description of the density matrix. In that case, the cost increases exponentially with the size of the state. On top of that, the current optimal classical algorithm for quantum entropy estimation has a cost that is linear to the number of non-zero elements of the density matrix [88].

Although aforementioned quantum algorithms [36, 56, 66, 93, 100, 135] have promised speedups over the tomography method in the entropy estimation task, the quantum query model for the input state, the most crucial component of these algorithms, is still not known how to construct efficiently. And hence, the timescale for these algorithms to be effective in practice remains an open question. On the other hand, the fast development of quantum computing devices has brought us into the noisy intermediate-scale quantum (NISQ) era [118]. An emergence of studies have focused on delivering quantum applications via near-term quantum devices [17, 31, 49], and estimating quantum entropies with applications (e.g., entanglement spectroscopy, Gibbs state preparation) is one of the fundamental tasks in the field. To better exploit NISQ devices in the quantum entropy estimation task, it is highly desirable to devise quantum algorithms without using the quantum query model.

In this work, we propose quantum algorithms of concrete implementation to estimate the von Neumann and quantum Rényi entropies of an unknown quantum state using independent copies of the input state. To develop our algorithms, we firstly use the Fourier series approximation to decompose the entropy. Then, we devise quantum circuits to estimate each term in the Fourier series. When design quantum circuits, we synthesize several quantum gadgets, such as the iterative quantum phase estimation [87], the

exponentiation of the quantum state in [98], the linear combination of unitaries [16], and qubit reset [45]. These gadgets make the circuits slightly friendly to NISQ devices (only include primitive single/two-qubit gates, and two copies of the state are maintained at a time). In the end, we could obtain the estimated entropy by classical post-processing. In particular, we utilize the sampling method to reduce the computational resources and speedup the computation.

This Chapter is based on the following paper:
Youle Wang, Benchi Zhao, and Xin Wang. (2022). Quantum algorithms for estimating quantum entropies. 22-26. http://arxiv.org/abs/2203.02386

In Chapter 6, we conclude the dissertation by summarizing the contributions and discussing several directions for future work.

CHAPTER 2

PRELIMINARY

2.1 Notation and terminology

Throughout this dissertation, we assume the reader is familiar with basic notions of quantum computing, including quantum states, quantum circuits, measurements for closed/open systems. More information about quantum computing can be found in the textbook by Nielsen and Chuang [111] and the lecture notes by Lin Lin [94]. We also assume the basic familiarity of linear algebra [9].

We use Dirac notation $|\phi\rangle$ to denote the pure quantum state, which is also a unit vector. $\langle\phi| = |\phi\rangle^\dagger$ means the dual vector of $|\phi\rangle$. The notation $\langle\psi|\phi\rangle$ represents the inner product between states $|\psi\rangle$ and $|\phi\rangle$. For a finite-dimensional space, we use $\{|\psi_j\rangle\}_{j=1}^d$ to represent the computational basis of the Hilbert space, with d being the space's dimension. Furthermore, we use notations ρ and σ to represent the mixed quantum states. The state overlap is given by $\mathrm{tr}(\rho\sigma)$, where tr means the trace of the matrix.

We use uppercase letters U and W to denote the quantum circuit, $\boldsymbol{\theta}$ denotes the vector of parameters of the parameterized quantum circuit. Uppercase letter $L(\boldsymbol{\theta})$ represents the loss function. The indexed notation, e.g., $U_l(\theta_l)$ and W_l, means the subcircuit module in the parameterized quantum circuit. H is the Hamiltonian, and H_l is a Pauli string.

2.2 Variational quantum algorithms

Variational quantum algorithms (VQAs) are put forward to effectively exploit NISQ computers, which have already found applications in many areas. [25, 50, 71, 110, 114, 146–148]. The core of VQAs is to solve a certain optimization problem by training a parameterized quantum circuit (PQC) [12]. A gradient-based method is usually used to perform the optimization, where quantum devices are employed to evaluate the loss function, and classical devices are used to update parameters. When the loss function converges to the global optimum, the optimization loop halts and outputs the final parameters, at that time, the PQC with final parameters will prepare the desired quantum states and reveal the solutions.

2.2.1 Parameterized quantum circuit

In the NISQ era, parameterized quantum circuits (PQCs) provide a concrete way to implement quantum algorithms. Generally, PQC is composed of a series of single-qubit rotations gates (e.g., R_x,R_y.R_z) and two-qubit fixed gates (e.g., CNOT/CZ). This dissertation uses $U(\boldsymbol{\theta})$ to denote the PQC, where $\boldsymbol{\theta}$ represents the vector of all tunable parameters. The explicit expression is given below.

$$U(\boldsymbol{\theta}) = U(\theta_1, ..., \theta_L) = \prod_{l=1}^{L} U_l(\theta_l) W_l \,, \tag{2.1}$$

where $U_l(\theta_l) = \exp(-i\theta_l H_l/2)$, H_l is a Pauli string, and W_l is a generic unitary operator that does not depend on any angle $\boldsymbol{\theta}$. Integer L is the circuit depth. An illustration is depicted in Fig. 2.1.

Next, we explain how to use PQC to solve practical tasks by reviewing the well-known variational quantum eigensolver, which uses a parameterized quantum circuit to learn the ground state of a Hamiltonian and output the ground state energy.

2.2.2 Variational quantu eigensolver

Variational quantum eigensolvers (VQE) [114] are a class of algorithms that use parameterized quantum circuit (PQC) to find the ground state of a Hamiltonian of practical interest, e.g., Hamiltonians of chemical molecules. In general, we are able to write the

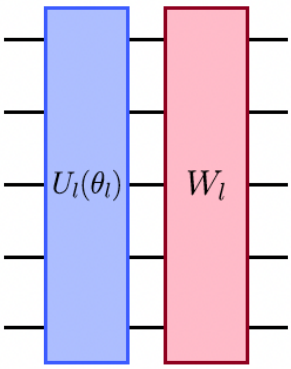


Figure 2.1: A circuit module of PQC. $U_l(\theta_l)$ is a gate with tunable parameter θ_l, and W_l is a parameter-free gate, e.g., CNOT.

n-qubit Hamiltonian of practical interest as a weighted sum of Pauli strings.

$$H = \sum_{l=1}^{m} \mu_l E_l, \tag{2.2}$$

where $\mu_l \in \mathbb{R}$ are interaction coefficients, $E_l \in \{X, Y, Z, I\}^{\otimes n}$ are Pauli strings that act on constant qubits, and integer $m = O(\text{poly}(n))$. The key of VQE is to solve an optimization problem.

$$\boldsymbol{\theta}^* = \text{argmin}_{\boldsymbol{\theta}} \langle 0^n | U^\dagger(\boldsymbol{\theta}) H U(\boldsymbol{\theta}) | 0^n \rangle. \tag{2.3}$$

Here, we suppose the input state of VQE is an n-qubit all-zero state, and H is the Hamiltonian of interest, and $U(\boldsymbol{\theta})$ means the parameterized quantum circuit (PQC) to use. The optimization is solved straightforwardly by the gradient-based or gradient-free method. In the next section, we discuss more details about estimating the gradient when employing a gradient-based method. In contrast to the classical simulation, the loss function here can be evaluated efficiently using quantum circuits.

2.2.3 Gradient estimation

The gradient descent method is the most often used method to solve the optimization problem in Eq. (2.3). We show how to transfer the gradient estimation to loss evaluation in the following.

Suppose we want to compute the partial derivative of $L(\boldsymbol{\theta}) = \langle 0^n | U^\dagger(\boldsymbol{\theta}) H U(\boldsymbol{\theta}) | 0^n \rangle$, i.e.,

$\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_k}$. The partial derivative can be written in the following form.

(2.4)
$$\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_k} = \frac{\partial \langle 0^n | \prod_{l=1}^{L} W_l^\dagger U_l^\dagger(\theta_l) H \prod_{l=1}^{L} U_l(\theta_l) W_l | 0^n \rangle}{\partial \theta_k}$$

(2.5)
$$= \langle 0^n | \frac{\partial}{\partial \theta_k} \prod_{l=1}^{L} W_l^\dagger U_l^\dagger(\theta_l) H \prod_{l=1}^{L} U_l(\theta_l) W_l | 0^n \rangle + \langle 0^n | \prod_{l=1}^{L} W_l^\dagger U_l^\dagger(\theta_l) H \frac{\partial}{\partial \theta_k} \prod_{l=1}^{L} U_l(\theta_l) W_l | 0^n \rangle.$$

Notice that $\frac{\partial}{\partial \theta_k} U_k(\theta_k) = \frac{-i}{2} H_k U_k(\theta_k)$. For convenience, we write the operations between layer 1 to $k-1$ as $U_{1:k-1}$, and operations between layer $k+1$ and L as $U_{k+1:L}$, respectively. Then the gradient can be rewritten as

$$\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_k} = \frac{i}{2} \langle 0^n | U_{1:k-1}^\dagger W_k^\dagger U_k^\dagger(\theta_k) H_k U_{k+1:L}^\dagger H U_{k+1:L} U_k(\theta_k) W_k U_{1:k-1} | 0^n \rangle$$

(2.6)
$$+ \frac{-i}{2} \langle 0^n | U_{1:k-1}^\dagger W_k^\dagger U_k^\dagger(\theta_k) U_{k+1:L}^\dagger H U_{k+1:L} H_k U_k(\theta_k) W_k U_{1:k-1} | 0^n \rangle$$

(2.7)
$$= \frac{i}{2} \langle 0^n | U_{1:k-1}^\dagger W_k^\dagger U_k^\dagger(\theta_k) \left[H_k U_{k+1:L}^\dagger H U_{k+1:L} - U_{k+1:L}^\dagger H U_{k+1:L} H_k \right] U_k(\theta_k) W_k . U_{1:k-1} | 0^n \rangle.$$

Next, we use the relation $i[H_k, M] = U_k^\dagger(\frac{\pi}{2}) M U_k(\frac{\pi}{2}) - U_k^\dagger(-\frac{\pi}{2}) M U_k(-\frac{\pi}{2})$ to simplify the formula of the gradient. At the same time, we absorb the $U_k^\dagger(\pm\frac{\pi}{2})$ and $U_k(\pm\frac{\pi}{2})$ into $U_k^\dagger(\theta_k)$ and $U_k(\frac{\pi}{2})$, respectively.

(2.8)
$$\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_k} = \frac{1}{2} \langle 0^n | U_{1:k-1}^\dagger W_k^\dagger U_k^\dagger(\theta_k) \times$$
$$\left[U_k^\dagger(\frac{\pi}{2}) U_{k+1:L}^\dagger H U_{k+1:L} U_k(\frac{\pi}{2}) - U_k^\dagger(-\frac{\pi}{2}) U_{k+1:L}^\dagger H U_{k+1:L} U_k(-\frac{\pi}{2}) \right] U_k(\theta_k) W_k . U_{1:k-1} | 0^n \rangle$$

(2.9)
$$= \frac{1}{2} \left[L(\hat{\boldsymbol{\theta}}_{k,+}) - L(\hat{\boldsymbol{\theta}}_{k,-}) \right].$$

Here, $\hat{\boldsymbol{\theta}}_{k,\pm} = (\theta_1, \ldots, \theta_k \pm \frac{\pi}{2}, \ldots)$ indicates that parameter θ_k is shifted by a phase of $\frac{\pi}{2}$, while others do not change.

Clearly, to compute the gradient, we can shift the corresponding parameter by a phase of $\pm\frac{\pi}{2}$. This method is called parameter shift rule [107]. As a result, the gradient estimation is transferred to the loss evaluation.

CHAPTER 3

QUANTUM HAMILTONIAN LEARNING ALGORITHM

3.1 Problem Statement

Recall the goal of Hamiltonian learning is to learn the interaction coefficients $\boldsymbol{\mu}$ from the measurement results of a quantum Gibbs state. We assume that the Hamiltonian to be learned H is composed of local Pauli operators $\{E_\ell\}_{\ell=1}^m$, and the measurements corresponding to $\{E_\ell\}_{\ell=1}^m$ are performed on the Gibbs state $\rho_\beta = e^{-\beta H}/\mathrm{tr}(e^{-\beta H})$ at an inverse temperature β. The measurement results are denoted by $\{e_\ell\}_{\ell=1}^m$, given by

$$e_\ell = \mathrm{tr}(\rho_\beta E_\ell), \quad \forall \ell \in [m]. \tag{3.1}$$

Notice that many methods proposed to efficiently obtain measurement results $\{e_\ell\}_{\ell=1}^m$ [21, 39, 72]. We, therefore, assume the measurement results $\{e_\ell\}_{\ell=1}^m$ have been given previously and focus on learning interaction coefficients from them. Formally, we define the Hamiltonian learning problem (HLP) as follows:

Definition 1 (HLP)**.** *Consider a many-body Hamiltonian with a decomposition given in Eq.* (1.1)*, where* $|\mu_\ell| \leq 1$ *for all* $\ell = 1, ..., m$*. Suppose we are given measurement results* $\{e_\ell\}_{\ell=1}^m$ *of the quantum Gibbs state* ρ_β*, then the goal is to find an estimate* $\widehat{\boldsymbol{\mu}}$ *of* $\boldsymbol{\mu}$ *such that*

$$\| \widehat{\boldsymbol{\mu}} - \boldsymbol{\mu} \|_\infty \leq \epsilon, \tag{3.2}$$

where $\| \cdot \|_\infty$ *means the maximum norm.*

To solve the HLP, we adopt a strategy that is proposed recently in Ref. [5], which transforms HLP into an optimization problem by using the Jaynes' principle (or maximal entropy principle) [75]. This strategy is to find a quantum state with the maximal entropy from all states whose measurement results under $\{E_\ell\}_{\ell=1}^m$ match $\{e_\ell\}_{\ell=1}^m$.

$$\begin{aligned} \max_{\rho} \quad & S(\rho) \\ s.t. \quad & \mathrm{tr}(\rho E_\ell) = e_\ell, \ \forall \ell = 1,...,m \\ & \rho > 0, \ \mathrm{tr}(\rho) = 1. \end{aligned} \tag{3.3}$$

It has been shown in [75] that the optimal state is of the following form:

$$\sigma = \frac{\exp(-\beta \sum_{\ell=1}^m \mu_\ell^* E_\ell)}{\mathrm{tr}(\exp(-\beta \sum_{\ell=1}^m \mu_\ell^* E_\ell))}. \tag{3.4}$$

Here, state σ is a quantum Gibbs state of a Hamiltonian with interaction coefficients $\boldsymbol{\mu}^* = (\mu_1^*, ..., \mu_m^*)$. As a result, Ref. [5] shows that coefficients of σ is the target interaction coefficients, i.e., $\boldsymbol{\mu}^* = \boldsymbol{\mu}$. Moreover, Ref. [5] also points out an approach for obtaining $\boldsymbol{\mu}^*$ that is to solve the dual optimization problem (cf. Eq. (1.2)). More specifically, the dual problem in Eq. (1.2) is a result of applying the Lagrange multiplier method to problem in Eq. (3.3), and coefficients $\boldsymbol{\nu}$ are the corresponding Lagrange multipliers.

In Ref. [5], it has shown that the loss function is strongly convex. Hence, a gradient descent method can steadily find the desired solution and recover the unknown Hamiltonian. To this end, we develop a gradient descent method tailored to quantum computers to solve the problem in Eq. (1.2). A flowchart for illustration is shown in Figure 3.1. Clearly, the main obstacle is to compute the corresponding gradients of the objective function, which involves computing the partition function. Then, we formalize the gradient estimation problem below.

Definition 2 (Gradient estimation)**.** *Given a Hamiltonian parameterized by coefficients $\boldsymbol{\nu}$, i.e., $H(\boldsymbol{\nu}) = \sum_{\ell=1}^m \nu_\ell E_\ell$, let $L(\boldsymbol{\nu})$ be the objective function*

$$L(\boldsymbol{\nu}) = \log Z_\beta(\boldsymbol{\nu}) + \beta \sum_{\ell=1}^m \nu_\ell e_\ell, \tag{3.5}$$

where $Z_\beta(\boldsymbol{\nu}) = \mathrm{tr}(e^{-\beta H(\boldsymbol{\nu})})$. Then the goal is to estimate the gradient $\nabla L(\boldsymbol{\nu})$ with respect to $\boldsymbol{\nu}$.

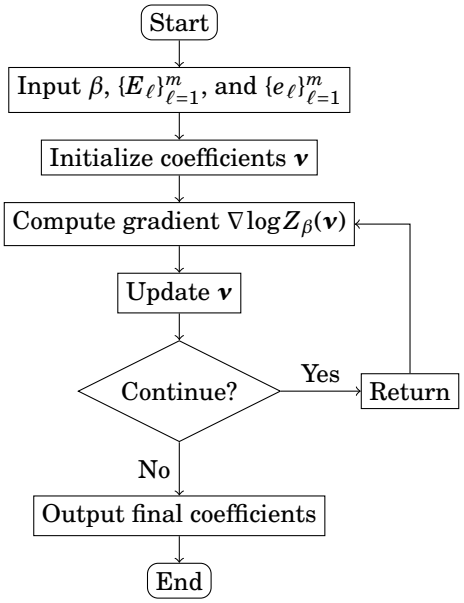


Figure 3.1: Flowchart of the gradient-descent method for Hamiltonian learning.

3.2 A gradient-descent solution

According to Eq. (4.2), one way to compute the loss function $L(\boldsymbol{\nu})$ and the gradient requires the ability to prepare the Gibbs states. In the literature, there are many proposals [43, 74, 148, 157, 158, 160] of Gibbs state preparation to this end. In this chapter, we propose a method that does not demand the Gibbs state preparation.

To compute the log-partition function, we develop a method based on the relation between the log-partition function and the system's free energy [122]. In general, suppose the state of the system is ρ, then the free energy is given by $F(\rho) = \mathrm{tr}(H\rho) - \beta^{-1}S(\rho)$, where $S(\rho)$ is the von Neumann entropy. The relation states that the global minimum of $F(\rho)$ is proportional to the log-partition function, i.e.,

$$\log \mathrm{tr}(e^{-\beta H}) = -\beta \min_{\rho} F(\rho). \tag{3.6}$$

Our method for minimizing the free energy depends on two key steps. First, we choose a suitable PQC with enough expressiveness; then, we train it to learn the eigenvectors of the Hamiltonian and output the corresponding eigenvalues. Second, we combine the post-training PQC with the classical methods for convex optimization to find the global minimum of the free energy. Next, we utilize the post-training PQC and the optimizer of the convex optimization to compute the gradients. We also theoretically analyze the estimation precision of the gradients. And, we discuss the efficiency of loss evaluation

and gradients estimation by the importance sampling technique when the underlying Hamiltonian is large.

Here, we consider computing the log-partition function $\log Z_\beta(\boldsymbol{v})$. Motivating our method is the relationship between the log-partition function and free energy. Recall that free energy of the system being state ρ is given by $F(\rho) = \mathrm{tr}(H(\boldsymbol{v})\rho) - \beta^{-1}S(\rho)$, assuming the parameterized Hamiltonian is $H(\boldsymbol{v}) = \sum_{\ell=1}^{m} v_\ell E_\ell$. Then the relation states that

$$\log Z_\beta(\boldsymbol{v}) = -\beta \min_{\rho} F(\rho). \tag{3.7}$$

As shown in Eq. (3.7), it is natural to minimize the free energy to obtain the value of $\log Z_\beta(\boldsymbol{v})$. However, it is infeasible to directly minimize the free energy on NISQ devices since performing entropy estimation with even shallow circuits is difficult [55]. To deal with this issue, we choose an alternate version of Eq. (3.7):

$$\log Z_\beta(\boldsymbol{v}) = -\beta \min_{\mathbf{p}} \sum_{j=1}^{N} p_j \cdot \lambda_j + \beta^{-1} \sum_{j=1}^{N} p_j \log p_j, \tag{3.8}$$

where $\boldsymbol{\lambda} = (\lambda_1, ..., \lambda_N)$ is the vector of eigenvalues of $H(\boldsymbol{v})$, and $\mathbf{p} = (p_1, ..., p_N)$ represents an N-dimensional probability distribution, with N the Hamiltonian's dimension. Please note that proofs for Eqs. (3.7)-(3.8) are provided in Appendix A.1.

Thus, optimizing the R.H.S of Eq. (3.8) could obtain the desired quantity and avoid the von Neumann entropy estimation simultaneously, assuming eigenvalues of the Hamiltonian $H(\boldsymbol{v})$ is given previously. As a result, our task is reduced to solve the following optimization program based on the equality in Eq. (3.8):

$$\begin{aligned} \min_{\mathbf{p}} \quad & C(\mathbf{p}) \\ \text{s.t.} \quad & \sum_{j=1}^{N} p_j = 1 \\ & p_j \geq 0, \forall j = 1, \ldots, N \end{aligned} \tag{3.9}$$

where

$$C(\mathbf{p}) = \sum_{j=1}^{N} p_j \cdot \lambda_j + \beta^{-1} \sum_{j=1}^{N} p_j \log p_j. \tag{3.10}$$

The optimization program in Eq. (3.9) is a typical convex optimization program. In the context of convex optimization, there are many classical algorithms to solve the optimization program, such as the interior-point method [81], ellipsoid method [64], cutting-plane method [83], and random walks [78], etc. For example, we consider using the cutting

plane method [76, 92], which requires the membership and evaluation procedures [91]. Concerning the program in Eq. (3.9), the membership procedure determines whether a point belongs to the set of probability distributions, and the evaluation procedure takes in a probability distribution $\mathbf{p}$ and returns the value $C(\mathbf{p})$ with high accuracy. Clearly, it is easy to determine whether the given point is a probability distribution while challenging to efficiently evaluate the function value. Thus, we provide a procedure to solve the convex optimization problem as well as overcome this challenge at the same time in Algorithm 1.

Algorithm 1 Log-partition function estimation

Input: Parameterized quantum circuit $U(\boldsymbol{\theta})$, Hamiltonian $H(\boldsymbol{\nu})$, constant β;
Output: An estimate for $\log Z_\beta(\boldsymbol{\nu})$;

1: # Evaluation procedure construction
2: Take probability distribution $\mathbf{p}$ as input;
3: Set integer T and D (cf Proposition 1);
4: Sample TD integers $t_1^1,...,t_T^1,...,t_1^D,...,t_T^D$ according to $\mathbf{p}$;
5: Prepare computational states $|\psi_{t_1^1}\rangle$, ..., $|\psi_{t_T^1}\rangle$, ..., $|\psi_{t_1^D}\rangle$, ..., $|\psi_{t_T^D}\rangle$;
6: Compute approximate eigenvalues: $\lambda_{t_j^s} = \langle\psi_{t_j^s}|U^\dagger(\boldsymbol{\theta})H(\boldsymbol{\nu})U(\boldsymbol{\theta})|\psi_{t_j^s}\rangle$ for all $j = 1,\ldots,T$ and $s = 1,\ldots,D$;
7: Compute averages: $ave_s = \frac{1}{T}\sum_{j=1}^{T}\lambda_{t_j^s}$ for all $s = 1,...,D$;
8: Take the median value $C(\mathbf{p}) \leftarrow median(ave_1,...,ave_D) + \beta^{-1}\sum_{j=1}^{N} p_j \log p_j$;
9: # Membership procedure construction
10: Construct a membership procedure;
11: # Convex optimization solution
12: Compute the function's global minimum value $C(\mathbf{p}^*)$ and the optimal point $\mathbf{p}^*$ via the cutting plane method.
13: **return** value $-\beta C(\mathbf{p}^*)$ and the final point $\mathbf{p}^*$.

In Algorithm 1, we compute the log-partition function using a classical convex optimization method. For this purpose, we first show the construction process of evaluation procedure. That is, given a point $\mathbf{p}$, find an estimate for $C(\mathbf{p})$. We assume we are given a parameterized quantum circuit $U(\boldsymbol{\theta})$ that can learn eigenvectors of the Hamiltonian $H(\boldsymbol{\nu})$. In our approach, the $U(\boldsymbol{\theta})$ is combined with the importance sampling technique (cf. lines 3-8) to deal with the large-sized Hamiltonians. The procedure is shown below:

1. Sample TD indices according to the distribution $\mathbf{p}$ (cf. line 4);
2. Evaluate the eigenvalues associated with the sampled indices (cf. lines 5-6);
3. Take the average over T (cf. line 7) and the median over D (cf. line 8) to evaluate the function value $C(\mathbf{p})$ with high accuracy and success probability.

Eventually, with the evaluation procedure and the membership procedure, the global minimum of $C(\mathbf{p})$ could be obtained via the cutting plane method [76, 91, 92]. Finally, based on the relationship between $\log Z_\beta(\boldsymbol{\nu})$ and $C(\mathbf{p}^*)$ (cf. Eq. (3.8)), we could derive the log-partition function value. Here $\mathbf{p}^*$ denotes the optimal distribution of the optimization in Eq. (3.8)).

Remark 1 Notice that a crucial gadget in Algorithm 1 is the PQC $U(\boldsymbol{\theta})$, which we have assumed to be accessible. To complement the assumption, we provide a procedure for extracting eigenvalues in the next section, i.e., *Stochastic variational quantum eigensolver*. In particular, we present a variational quantum algorithm for learning the eigenvectors of the parameterized Hamiltonians.

Now we discuss the cost of applying Algorithm 1. As the efficiency of Algorithm 1 mainly relies on the cost of the evaluation procedure, we only discuss it here. Suppose we have access to Hamiltonian $H(\boldsymbol{\nu})$'s eigenvalues $\boldsymbol{\lambda}$, then the objective function $C(\mathbf{p})$ can be effectively evaluated. Recall that $C(\mathbf{p})$ contains two parts $\sum_{j=1}^N p_j \cdot \lambda_j$ and $\beta^{-1}\sum_{j=1}^N p_j \log p_j$. On the one hand, the latter value can be computed immediately since $\mathbf{p}$ is stored on classical devices. On the other hand, value $\sum_{j=1}^N p_j \cdot \lambda_j$ can be regarded as an expectation of the probability $\mathbf{p}$, where value λ_j is sampled with probability p_j. Notably, the total cost for estimating $C(\mathbf{p})$ is dominated by the number of samples. Then we analyze the number of required samples for loss evaluation in Proposition 1.

Proposition 1. *For any constant $\beta > 0$ and parameterized Hamiltonian $H(\boldsymbol{\nu}) = \sum_{\ell=1}^m \nu_\ell E_\ell$ with $E_\ell \in \{X, Y, Z, I\}^{\otimes n}$ and $\boldsymbol{\nu} \in \mathbb{R}^m$, suppose we are given access to a parameterized quantum circuit $U(\boldsymbol{\theta})$ that can prepare $H(\boldsymbol{\nu})$'s eigenvectors, then the objective function $C(\mathbf{p})$ can be computed up to precision ϵ with probability larger than $2/3$ by taking $T = O(m\|\boldsymbol{\nu}\|_2^2/\epsilon^2)$ samples. Furthermore, the probability can be improved to $1-\eta$ costing an additional multiplicative factor of $D = O(\log(1/\eta))$.*

Sketch of proof In general, the expectation can be approximated by the sample mean according to Chebyshev's inequality. Specifically speaking, the expectation can be estimated up to precision ϵ with high probability (e.g., larger than 2/3) by taking $O(\mathbf{V}ar/\epsilon^2)$ samples, where $\mathbf{V}ar$ denotes the variance of the distribution. Here, the number of samples is $T = O(m\|\boldsymbol{\nu}\|_2^2/\epsilon^2)$, since the variance is bounded by the squared spectral norm of $H(\boldsymbol{\nu})$, which is less than $\sqrt{m}\|\boldsymbol{\nu}\|_2$. Furthermore, Chernoff bounds allow improving success probability to $1-\eta$ at an additional cost of a multiplicative factor of $D = O(\log(1/\eta))$. More details are deferred to Appendix A.1. ■

As shown in Proposition 1, our evaluation method is computationally efficient, since the number of samples scales polynomially with the number of qubits. Hence Algorithm 1 could be applied to compute the partition function of the parameterized Hamiltonian, given the suitable PQC $U(\boldsymbol{\theta})$.

3.3 Hamiltonian learning algorithm

Eventually, we present our hybrid quantum-classical algorithm for Hamiltonian learning (HQHL) in Algorithm 2. The main idea of HQHL is to find the target interaction coefficients by a gradient-descent method (cf. Figure 3.1). Thus, HQHL's main process is to compute the gradient of the objective function. Specifically, we take Pauli operators $\{E_\ell\}_{\ell=1}^m$, $\{e_\ell\}_{\ell=1}^m$, and β as input. Then we initialize the coefficients by choosing $\boldsymbol{\nu}$ from $[-1,1]^m$ uniformly at random. Next, we compute the gradient of the objective function $L(\boldsymbol{\nu})$ by Algorithm 4. Then update the coefficients by choosing a suitable learning rate r and using the estimated gradient. In consequence, after repeating the training process sufficiently many times, the final coefficients are supposed to approximate the target coefficients $\boldsymbol{\nu}$.

Algorithm 2 Hybrid quantum-classical Hamiltonian learning algorithm (HQHL)

Input: Pauli operators $\{E_\ell\}_{\ell=1}^m$, constants $\{e_\ell\}_{\ell=1}^m$, and β;
Output: An estimate for target coefficients $\boldsymbol{\nu}$;

1: Initialize coefficients $\{\nu_\ell\}_{\ell=1}^m$;
2: Set number of iterations I and $l = 1$;
3: Set parameterized quantum circuit $U(\boldsymbol{\theta})$;
4: Set learning rate r;
5: **while** $l \leq I$ **do**
6: Set Hamiltonian $H(\boldsymbol{\nu}) = \sum_{\ell=1}^m \nu_\ell E_\ell$;
7: Train $U(\boldsymbol{\theta})$ by SVQE with $H(\boldsymbol{\nu})$;
8: Derive a probability $\widehat{\mathbf{p}}^*$ by performing log-partition function estimation with $U(\boldsymbol{\theta})$ and β;
9: Compute gradient $\nabla L(\boldsymbol{\nu})$ by gradient estimation with $U(\boldsymbol{\theta})$, $\widehat{\mathbf{p}}^*$, and β;
10: Update coefficients $\boldsymbol{\nu} \leftarrow \boldsymbol{\nu} - r\nabla L(\boldsymbol{\nu})$;
11: Set $l \leftarrow l + 1$;
12: **end while**
13: **return** the final coefficients $\boldsymbol{\nu}$.

Notably, the learning process is in the "**while**" loop of HQHL. In the loop, the subroutine SVQE (cf. Sec. 3.3.1) is first called to learn Hamiltonian's eigenvectors and eigenvalues. Here, we choose a suitable parameterized quantum circuit $U(\boldsymbol{\theta})$ and train it

to prepare the eigenvectors of the Hamiltonian $H(\boldsymbol{\nu})$. Afterwards, we enter the process of the log-partition function estimation. It first exploits the $U(\boldsymbol{\theta})$ to output the estimated eigenvalues of the parameterized Hamiltonian $H(\boldsymbol{\nu})$ and then computes the objective function $L(\boldsymbol{\nu})$. We would obtain a probability distribution $\widehat{\mathbf{p}}^*$ that consists of eigenvalues of the associated Gibbs state $\rho_\beta(\boldsymbol{\nu}) = e^{-\beta H(\boldsymbol{\nu})}/Z_\beta(\boldsymbol{\nu})$. Lastly, we exploit the resultant results (post-training circuit $U(\boldsymbol{\theta})$ and distribution $\widehat{\mathbf{p}}^*$) to compute the gradients following the procedure in Algorithm 4 and update the coefficient $\boldsymbol{\nu}$ accordingly (cf. Eq. (3.13)).

Remark 2 To improve the scalability of our method, on the one hand, we use the importance sampling technique to circumvent many resource requirements for Hamiltonian diagonalization. On the other hand, we discuss that learning partial eigenvalues could also lead to the target Hamiltonian. In particular, we numerically show that learning several low-lying eigenvectors could help recover the unknown Hamiltonian in Sec. 3.4.3.

3.3.1 Variational quantum Hamiltonian spectrum solver

This section discusses learning the eigenvectors of the parameterized Hamiltonian $H(\boldsymbol{\nu})$ using variational quantum algorithms and the importance sampling technique. First, we outline the algorithm in Algorithm 3 and then discuss the fundamental theory. Second, we circumvent the cost for coping with large-scaled Hamiltonians by the importance sampling technique. We also analyze the cost of loss evaluation in the algorithm.

To incorporate variational quantum algorithms, we utilize the variational principle of Hamiltonian's eigenvalues. That is, Hamiltonian's eigenvalues majorize the diagonal elements, and the dot function with an increasingly ordered vector is Schur concave [126]. A similar idea has already been discussed in [110]. In contrast, our method learns the full spectrum of the Hamiltonian. We define a function $M(\boldsymbol{\theta})$ over all parameters $\boldsymbol{\theta}$ of the circuit.

$$M(\boldsymbol{\theta}) = \sum_{j=1}^{N} q_j \cdot \langle \psi_j | U^\dagger(\boldsymbol{\theta}) H(\boldsymbol{\nu}) U(\boldsymbol{\theta}) | \psi_j \rangle, \tag{3.11}$$

where $\mathbf{q} = (q_1, ..., q_N)$ is a probability distribution such that $q_1 < q_2 < ... < q_N$, and notations $|\psi_1\rangle, \ldots, |\psi_N\rangle$ denote the computational basis. Suppose that PQC $U(\boldsymbol{\theta})$ has enough expressiveness, then $U(\boldsymbol{\theta})|\psi_j\rangle$ could learn the j-th eigenvector of the Hamiltonian $H(\boldsymbol{\nu})$ with suitable parameters. Particularly, $M(\boldsymbol{\theta})$ will reach the global minimum when all eigenvectors are learned. In other words, we use the PQC $U(\boldsymbol{\theta})$ to learn eigenvectors via finding the global minimum of $M(\boldsymbol{\theta})$ over all parameters $\boldsymbol{\theta}$.

Algorithm 3 Stochastic variational quantum eigensolver (SVQE)

Input: Parameterized quantum circuit $U(\boldsymbol{\theta})$, Hamiltonian $H(\boldsymbol{\nu})$, and weights $\mathbf{q}$;
Output: Optimal PQC $U(\boldsymbol{\theta})$;

Set number of iterations I and $l = 1$;
Set integers T and D;
Set learning rate r_θ;
Set probability distribution $\mathbf{q}$;
Sample TD integers $k_1^1,\ldots,k_T^1,\ldots,k_1^D,\ldots,k_T^D$ according to $\mathbf{q}$;
Prepare computational states $|\psi_{k_1^1}\rangle, \ldots, |\psi_{k_T^1}\rangle, \ldots, |\psi_{k_1^D}\rangle, \ldots, |\psi_{k_T^D}\rangle$;
while $l \leq I$ **do**
    Compute value $\langle\psi_{k_j^s}|U^\dagger(\boldsymbol{\theta})H(\boldsymbol{\nu})U(\boldsymbol{\theta})|\psi_{k_j^s}\rangle$ for all $j = 1,\ldots,T$ and $s = 1,\ldots,D$;
    Compute averages: $ave_s = \frac{1}{T}\sum_{j=1}^{T}\langle\psi_{k_j^s}|U^\dagger(\boldsymbol{\theta})H(\boldsymbol{\nu})U(\boldsymbol{\theta})|\psi_{k_j^s}\rangle$ for all $s = 1,...,D$;
    Let $M(\boldsymbol{\theta}) \leftarrow median(ave_1,...,ave_D)$;
    Use $M(\boldsymbol{\theta})$ to compute the gradient ∇ by parameter shift rules [107];
    Update parameters $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - r_\theta\nabla$;
    Set $l \leftarrow l + 1$;
end while
return the final $U(\boldsymbol{\theta})$.

Remark 3 Choosing a suitable $U(\boldsymbol{\theta})$ is critical to many variational quantum algorithms as well as our Algorithm 3. With enough expressibility, training the PQC $U(\boldsymbol{\theta})$ would allow us to exactly or approximately learn the solution to the certain problem. The expressibility of PQCs has been recently studied in [132]. Note that PQCs with high expressive power generally suffer from the barren plateaus [29, 30, 33, 67] and there exhibits a trade-off between their trainability and expressivity [34, 44, 69]. We also note that we can dynamically design a problem-specific ansatz [20, 42, 119, 163].

Remark 4 In the learning process, we employ a gradient-based method to update the parameters $\boldsymbol{\theta}$ iteratively. In each iteration, the corresponding gradients are computed via the parameter shift rule [107], which outsources the gradient estimation to the loss evaluation. As this is similar to other variational quantum algorithms, we omit the details of gradient computation. For details of gradient derivation, please refer to the proof of Proposition 3 in [146].

Notice that for large Hamiltonians, the loss $M(\boldsymbol{\theta})$ may consist of exponentially many terms, which would be a huge burden to the loss evaluation. However, we could employ the importance sampling technique to circumvent this issue. To this end, $M(\boldsymbol{\theta})$ is taken as an expectation of the distribution $\mathbf{q}$. Hence, $M(\boldsymbol{\theta})$ is to be estimated by the sample mean. Notably, the cost of loss evaluation is dominated by the number of samples, which

is why we call our method stochastic variational quantum eigensolver (SVQE). Our algorithm with importance sampling for minimizing $M(\boldsymbol{\theta})$ is depicted in Algorithm 3. In the following, we analyze the sample complexity in the loss evaluation.

Proposition 2. *Consider a Hamiltonian $H(\boldsymbol{v}) = \sum_{\ell=1}^{m} v_\ell E_\ell$ with Pauli operators $E_\ell \in \{X,Y,Z,I\}^{\otimes n}$ and constants $v_\ell \in [-1,1]$. Given any constants $\epsilon > 0$, $\eta \in (0,1)$, $\beta > 0$, the objective function $M(\boldsymbol{\theta})$ in SVQE can be estimated up to precision ϵ with probability at least $1-\eta$, costing TD samples with $T = O(m\|\boldsymbol{v}\|_2^2/\epsilon^2)$ and $D = O(\log(1/\eta))$. Besides, the total number of measurements is given below:*

$$O\left(\frac{mTD\|\boldsymbol{v}\|_1^2(n+\log(m/\eta))}{\epsilon^2}\right). \tag{3.12}$$

Sketch of proof The number of samples is determined by the accuracy ϵ and Hamiltonian $H(\boldsymbol{v})$. By Chebyshev's inequality, estimating $M(\boldsymbol{\theta})$ up to precision ϵ with high probability requires $T = O(m\|\boldsymbol{v}\|_2^2/\epsilon^2)$ samples, since the variance is bounded by the spectral norm, which is less than $\sqrt{m}\|\boldsymbol{v}\|_2$. Meanwhile, the expectation value $\langle\psi_j|U^\dagger(\boldsymbol{\theta})H(\boldsymbol{v})U(\boldsymbol{\theta})|\psi_j\rangle$ is evaluated by measurements. We compute the expectation value of the observable $H(\boldsymbol{v})$ by measuring each Pauli operator E_ℓ separately, since there are only $m = O(poly(n))$ Pauli operators. More details are deferred to Appendix A.1. ■

Remark 5 Other methods for computing expectation value of Hamiltonians can be found in Ref. [6, 136], where importance sampling is employed to sample Pauli operator E_l of the Hamiltonian. Moreover, a technique called classical shadow [72] could also be exploited to this end. Particularly, it can save the resources of quantum states to estimate the expectation of Hamiltonian that consists of many terms of Pauli strings.

Remark 6 In the context of quantum algorithms, there are many proposed methods for learning the low-lying eigenvectors of the Hamiltonian and diagonalizing Hamiltonian. Some known quantum algorithms for Hamiltonian diagonalization are based on quantum fast Fourier transform [1], which may be too costly for NISQ computers and thus not suitable for our purpose. Recently, there have already been some works on finding ground and excited eigenstates of the Hamiltonian with NISQ devices, i.e., variational quantum eigensolvers [38, 68, 77, 79, 106, 110, 114, 146]. They may be employed to learn eigenvectors in the Hamiltonian learning framework.

3.3.2 Gradient estimation

Recall that we employ a gradient-based method to do the optimization in the Hamiltonian learning (cf. Figure 3.1). We use the tools developed previously to derive the gradient estimation procedure.

Usually, with the estimated gradient, parameters are updated in the following way:

$$\boldsymbol{\nu} \leftarrow \boldsymbol{\nu} - r\nabla L(\boldsymbol{\nu}), \tag{3.13}$$

where r is the learning rate. The expression of the gradient is given below.

$$\nabla L(\boldsymbol{\nu}) = \left(\frac{\partial L(\boldsymbol{\nu})}{\partial \nu_1}, ..., \frac{\partial L(\boldsymbol{\nu})}{\partial \nu_m}\right). \tag{3.14}$$

Furthermore, the explicit formula of each partial derivative is given in [5]:

$$\frac{\partial L(\boldsymbol{\nu})}{\partial \nu_\ell} = \frac{\partial}{\partial \nu_\ell} \log Z_\beta(\boldsymbol{\nu}) + \beta e_\ell = -\beta \operatorname{tr}(\rho_\beta(\boldsymbol{\nu}) E_\ell) + \beta e_\ell, \tag{3.15}$$

where $\rho_\beta(\boldsymbol{\nu}) = e^{-\beta H(\boldsymbol{\nu})}/Z_\beta(\boldsymbol{\nu})$ represents the Gibbs state associated with the parameterized Hamiltonian $H(\boldsymbol{\nu})$.

Algorithm 4 Gradient estimation

Input: Post-training circuit $U(\boldsymbol{\theta})$, Pauli operators $\{E_\ell\}_{\ell=1}^m$, optimal $\widehat{\mathbf{p}}^*$, and constants β and $\{e_\ell\}_{\ell=1}^m$;

Output: Gradient estimate $\nabla L(\boldsymbol{\nu})$;

1: Set $\ell = 1$;
2: Set integer K and D;
3: Sample K integers $l_1^1, ..., l_K^1, ..., l_1^D, ..., l_K^D$, according to $\widehat{\mathbf{p}}^*$;
4: Prepare computational states $|\psi_{l_1^1}\rangle, \ldots, |\psi_{l_K^1}\rangle, \ldots, |\psi_{l_1^D}\rangle, \ldots, |\psi_{l_K^D}\rangle$;
5: **while** $\ell \leq m$ **do**
6: Compute value $\langle\psi_{l_j^s}|U^\dagger(\boldsymbol{\theta})E_\ell U(\boldsymbol{\theta})|\psi_{l_j^s}\rangle$ for $j = 1, .., K$ and $s = 1, ..., D$;
7: Calculate averages: $ave_s = \frac{1}{K}\sum_{j=1}^K \langle\psi_{l_j^s}|U^\dagger(\boldsymbol{\theta})E_\ell U(\boldsymbol{\theta})|\psi_{l_j^s}\rangle$ for all $s = 1, ..., D$;
8: Take the median value: $s_\ell = -\beta \cdot median(ave_1, \ldots, ave_D) + \beta e_\ell$;
9: Set $\ell \leftarrow \ell + 1$;
10: **end while**
11: **return** vector $(s_1, ..., s_m)$.

Here we provide a procedure for gradient estimation without preparing the Gibbs state $\rho_\beta(\boldsymbol{\nu})$ in Algorithm 4. We use the post-training PQC $U(\boldsymbol{\theta})$ and the optimal distribution $\widehat{\mathbf{p}}^*$ (cf. Algorithm 1), respectively. And the component of the gradient can be computed in the sense that

$$\frac{\partial L(\boldsymbol{\nu})}{\partial \nu_\ell} \approx -\beta \sum_{j=1}^N \widehat{p}_j^* \cdot \langle\psi_j|U^\dagger(\boldsymbol{\theta})E_\ell U(\boldsymbol{\theta})|\psi_j\rangle + \beta e_\ell. \tag{3.16}$$

The validity of the relation in Eq. (3.16) is proved in Proposition 3.

Proposition 3 (Correctness)**.** *Consider a parameterized Hamiltonian $H(\boldsymbol{\nu})$ and its Gibbs state $\rho_\beta(\boldsymbol{\nu})$. Suppose the $U(\boldsymbol{\theta})$ from SVQE (cf. Algorithm 3) and $\widehat{\mathbf{p}}^*$ from log-partition function estimation procedure (cf. Algorithm 1) are optimal. Define a density operator ρ_β^* as follows:*

$$\rho_\beta^* = \sum_{j=1}^{N} \widehat{p}_j^* \cdot U(\boldsymbol{\theta}) \left|\psi_j\right\rangle\!\left\langle\psi_j\right| U^\dagger(\boldsymbol{\theta}), \tag{3.17}$$

where $\{|\psi\rangle_j\}$ denote the computational basis. Denote the estimated eigenvalues by $\widehat{\boldsymbol{\lambda}}$, where $\widehat{\lambda}_j = \langle\psi_j|U^\dagger(\boldsymbol{\theta})H(\boldsymbol{\nu})U(\boldsymbol{\theta})|\psi_j\rangle$ for all $j = 1,\ldots,N$. Then, ρ_β^* is an approximation of $\rho_\beta(\boldsymbol{\nu})$ in the sense that

$$D(\rho_\beta^*, \rho_\beta(\boldsymbol{\nu})) \leq \sqrt{2\beta \max\left\{\mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda} - \lambda|], \mathbf{E}_{\mathbf{p}^*}[|\widehat{\lambda} - \lambda|]\right\}}. \tag{3.18}$$

where $D(\cdot,\cdot)$ denotes the trace distance, $\boldsymbol{\lambda}$ represent $H(\boldsymbol{\nu})$'s true eigenvalues, $\mathbf{p}^*$ is the distribution corresponding to $\boldsymbol{\lambda}$, i.e., $\lambda_j = e^{-\beta\lambda_j}/\sum_l e^{-\beta\lambda_l}$, and

$$\mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda} - \lambda|] = \sum_{j=1}^{N} \widehat{p}_j^* |\widehat{\lambda}_j - \lambda_j|, \quad \mathbf{E}_{\mathbf{p}^*}[|\widehat{\lambda} - \lambda|] = \sum_{j=1}^{N} p_j^* |\widehat{\lambda}_j - \lambda_j|. \tag{3.19}$$

Note that the quantity in Eq. (3.16) contains an expectation of distribution $\widehat{\mathbf{p}}^*$, then the partial derivative $\frac{\partial L(\boldsymbol{\nu})}{\partial \nu_\ell}$ is estimated by the sample mean. Specifically, we first randomly select the computational basis vectors $|\psi_j\rangle$ complying with distribution $\widehat{\mathbf{p}}^*$ and then compute the associated eigenvalues via $U(\boldsymbol{\theta})$. The detailed procedure of sampling and estimate computation is laid out in Algorithm 4. The number of required samples is analyzed in Proposition 4.

Proposition 4 (Sample complexity)**.** *Given $\epsilon > 0$ and $\eta \in (0,1)$, Algorithm 4 can compute an estimate for the gradient $\nabla L(\boldsymbol{\nu})$ up to precision ϵ with probability larger than $1-\eta$. Particularly, the overall number of samples is $KD = O(\beta^2 \log(2m/\eta)/\epsilon^2)$ with $K = O(\beta^2/\epsilon^2)$ and $D = O(\log(2m/\eta))$. Besides, the total number of measurements is $O(KD \cdot m\beta^2(n + \log(m/\eta))/\epsilon^2)$.*

The proofs for Propositions 3-4 are deferred to Appendix A.1.

To validate the gradient estimation, we show that the average of the overall errors determines the accuracy of the gradient estimation. For this purpose, Proposition 3 shows that matrix ρ_β^* is an approximation of the desired density matrix $\rho_\beta(\boldsymbol{\nu})$. Specifically, the trance distance between ρ_β^* and $\rho_\beta(\boldsymbol{\nu})$ is dependent on the averaged errors $\mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda} -$

$\lambda|]$ and $\mathbf{E}_{\mathbf{p}^*}[|\widehat{\lambda}-\lambda|]$. Here, notation $|\widehat{\lambda}-\lambda|$ denotes the difference between estimated eigenvalue and the associated real eigenvalue. $\widehat{\mathbf{p}}^*$ and $\mathbf{p}^*$ are probability distributions, corresponding to $\widehat{\boldsymbol{\lambda}}$ and $\boldsymbol{\lambda}$, respectively. In particular, the distributions $\mathbf{p}^*$ and $\hat{\mathbf{p}}^*$ contain components that are decreasing exponentially and hence $\mathbf{E}_{\mathbf{p}^*}[|\hat{\lambda}-\lambda|]$ and $\mathbf{E}_{\hat{\mathbf{p}}^*}[|\hat{\lambda}-\lambda|]$ can be determined with high accuracy by several components of $|\hat{\lambda}-\lambda|$. As a result, it implies that learning several low-lying eigenvectors with high accuracy may lead to a high precision estimate of the gradient. We numerically verify this feature in Sec. 3.4.3.

Moreover, Proposition 4 shows the feasibility of our approach as the number of measurements scales polynomially in parameters n, $1/\epsilon$, and β.

Remark 7 Note that the convergence rate may slow down if the estimated gradient has an error. However, we can always set the error as small as possible to alleviate this issue. At the same time, despite the estimation error, the effect on the convergence rate can be suppressed due to the strong convexity of the loss function [22]. For example, experimental results in Sec. 3.4.3 show that even though the estimated gradients are implicit, the loss function could still converge in a reasonable time.

3.4 Numerical Results

In this section, we conduct numerical experiments to verify the correctness of our algorithm. Specifically, we consider recovering interactions coefficients of several Hamiltonians, including randomly generated Hamiltonians and many-body Hamiltonians. To ensure the performance of the algorithm, we choose a PQC (shown in Fig. 3.2) and set the circuit with enough expressibility. When testing our algorithm, we first use SVQE to learn the full spectrum of Hamiltonians, where size of the Hamiltonian varies from 3 to 5. In SVQE, weights $\mathbf{q}$ consists of a normalized sequence of arithmetic sequence. For instance, when $n=3$, $\mathbf{q}=(1,2,3,\ldots,8)/S_3$, where $S_3=\sum_{l=1}^{8} l$. Furthermore, in order to reduce quantum resources, we also partially learn the few smallest eigenvalues of the selected Ising models and derive estimates for coefficients up to precision 0.05. With fewer eigenvalues to be learned, the depth of the used PQC is significantly reduced.

3.4.1 Random Hamiltonian models

This section shows the effectiveness of our algorithm with random Hamiltonians from three aspects: different β, different numbers of μ (# μ) and a different number of qubits (# qubits).

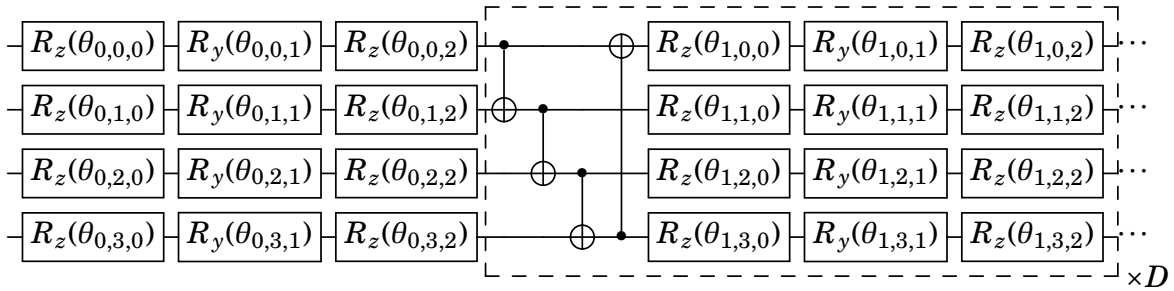


Figure 3.2: The selected quantum circuit $U(\boldsymbol{\theta})$ for stochastic variational quantum eigensolver (SVQE). Here, D represents circuit depth. Parameters $\boldsymbol{\theta}$ are randomly initialized from a uniform distribution in [0, 2π] and updated via gradient descent method.

Table 3.1: Hyper-parameters setting. The number of qubits (# qubits) varies from 3 to 5, and the number of μ (# μ) from 3 to 6. β is chosen as 0.3, 1, 3. "LR" denotes learning rate. The values of μ are sampled uniformly in the range of [-1, 1]. The term, likes "[[0 2 1] [2 1 3] [0 3 3]]", indicates there are three E_l's and each has three qubits with the corresponding Pauli tensor product. Here "0,1,2,3" represent "I, X, Y, Z" respectively. For example, for the first sample, the corresponding Hamiltonian is taken as H=0.3408 $\cdot I \otimes Y \otimes X$ -0.6384 $\cdot Y \otimes X \otimes Z$ -0.4988 $\cdot I \otimes Z \otimes Z$.

Aspects	n	# μ	β	LR	$\boldsymbol{\mu}$	E_l
	3	3	1	1.0	[0.3408 -0.6384 -0.4988]	[[0 2 1] [2 1 3] [0 3 3]]
β	3	3	0.3	8.0	[-0.4966 -0.8575 -0.7902]	[[1 0 0] [3 0 2] [3 1 3]]
			3	0.1	[0.5717 -0.1313 0.2053]	[[1 0 0] [3 3 3] [0 2 3]]
# μ	3	4	1	1.0	[-0.7205 -0.3676 -0.7583 -0.3002]	[[3 2 1] [2 1 3] [0 0 2] [2 0 0]]
		5			[-0.5254 -0.1481 -0.0037 -0.4373 0.7326]	[[1 3 0] [2 1 1] [3 3 2] [2 3 1] [0 2 0]]
		6			[-0.5992 0.7912 0.5307 -0.5422 -0.9239 0.0354]	[[3 2 2] [0 2 1] [1 2 1] [2 2 0] [0 1 2] [3 2 1]]
# qubits	4	3	1	1.0	[0.0858 0.3748 -0.1007]	[[0 2 0 1] [1 0 0 1] [2 0 1 0]]
	5				[-0.0411 0.7882 0.6207]	[[2 2 2 1 2] [2 3 3 2 1] [1 2 0 2 3]]

In the experimental setting, we randomly choose Pauli tensor products E_ℓ from $\{X, Y, Z, I\}^{\otimes n}$ and target coefficients $\boldsymbol{\mu}$ by a uniform distribution over $[-1, 1]$. Specifically, we first vary the values of β by fixing the number of μ and the number of qubits to explore our method's sensitivity to temperature. We similarly vary the number of μ and the number of qubits by fixing other hyper-parameters to explore our method's scalability. The actual values of these hyper-parameters sampled/chosen in each trial are concluded in Table 3.1. In addition, the depth, D, of the PQC $U(\boldsymbol{\theta})$ is set according to the size of Hamiltonian. As number of qubits ranges from $n = 3$ to $n = 5$, the depth D is set to be $10, 20, 40$, respectively.

Table 3.2: Hyper-parameters setting for many-body models. For each Hamiltonian model, the number of qubits varies from 3 to 5, and the number of μ is determined by the number of Pauli operators. “LR" denotes learning rate. The values of μ are sampled uniformly in the range of $[-1,1]$.

Many-body models	# qubits	# μ	β	LR	$\boldsymbol{\mu}$
Ising model	3	6	1.0	2.0	$[J_0 = 0.1981, \quad h_0 = 0.7544]$
	4	8		1.0	$[J_0 = 0.5296, \quad h_0 = 0.4996]$
	5	10		0.5	$[J_0 = -0.6916, \quad h_0 = 0.4801]$
XY model	3	6	1.0	1.0	$J_1 = -0.0839$
	4	8		1.0	$J_1 = 0.2883$
	5	10		0.6	$J_1 = -0.7773$
Heisenberg	3	12	1.0	1.0	$[J_2 = 0.0346, \quad h_2 = 0.8939]$
	4	16		1.0	$[J_2 = -0.5831, \quad h_2 = -0.0366]$
	5	20		1.0	$[J_2 = 0.2883, \quad h_2 = -0.2385]$

In Table 3.1, Hamioltonian is represented by a tuple. Each number $0,1,2,3$ corresponds to matrices I,X,Y,Z, respectively. $\boldsymbol{\mu}$ denotes the interaction coefficients to be learned. For instance, [[0 2 1] [2 1 3] [0 3 3]] means that the Hamiltonian consists of three Pauli operators, where each term represents a Pauli operator, e.g., [0 2 1] means $I \otimes Y \otimes X$. Then, the parameters in the top second row represents the following Hamiltonian.

$$0.3408 I \otimes Y \otimes X - 0.6384 Y \otimes X \otimes Z - 0.4988 I \otimes Z \otimes Z. \tag{3.20}$$

Other Hamiltonians to be tested are represented in a similar fashion.

The results for these three aspects are illustrated in Fig. 3.3. We find that all curves converge to the values close to 0 in less than ten iterations, which shows our method is effective. In particular, our method works for low temperatures β means that it is robust to temperature. And the results for the different number of μ and qubits reveals our method’s scalability to a certain extent.

3.4.2 Quantum many-body models

Here, we demonstrate the performance of our algorithm for quantum many-body models. Specifically, we consider the one-dimensional nearest-neighbor Ising model, XY model, and Heisenberg model. These many-body models are described by the Hamiltonians

shown below:

$$\text{(3.21)} \qquad \text{(Ising model)} \quad H_0 = J_0 \sum_{l=1}^{n} Z^l Z^{l+1} + h_0 \sum_{l=1}^{n} X^l,$$

$$\text{(3.22)} \qquad \text{(XY model)} \quad H_1 = J_1 \sum_{l=1}^{n} (X^l X^{l+1} + Y^l Y^{l+1}),$$

$$\text{(3.23)} \qquad \text{(Heisenberg model)} \quad H_2 = J_2 \sum_{l=1}^{n} (X^l X^{l+1} + Y^l Y^{l+1} + Z^l Z^{l+1}) + h_2 \sum_{l=1}^{n} Z^l,$$

where periodic boundary conditions are assumed (i.e., $X^{n+1} = X^1$, $Y^{n+1} = Y^1$, and $Z^{n+1} = Z^1$). Notation $Z^l = I \otimes \ldots \otimes Z \otimes \ldots I$ that acts on the l-th qubit. Coefficient J is the coupling constant for the nearest neighbor interaction, and h represents the external transverse magnetic field. The experimental parameters are concluded in Table 3.2.

We consider the models with a different number of qubits, varying from $n = 3$ to $n = 5$. The inverse temperature is set as $\beta = 1$. The coefficients J_0, J_1, J_2 and h_0, h_2 are sampled uniformly from a uniform distribution on [-1,1]. We also employ the parameterized quantum circuit $U(\boldsymbol{\theta})$ in Fig. 3.2 for the SVQE. And the depth of $U(\boldsymbol{\theta})$ is also set as $D = 10, 20, 40$ for different n. Moreover, the numerical results are shown in Fig. 3.3, which imply our method is applicable to recover quantum many-body Hamiltonians.

3.4.3 Numerical results using fewer eigenvalues of Ising Hamiltonians

Notice that we use a PQC $U(\boldsymbol{\theta})$ with deep depths to learn the full spectrum of small-sized Hamiltonians in Secs. 3.4.1-3.4.2, which may be beyond the capacity of NISQ devices. However, this section demonstrates the efficacy of HQHL in learning the Ising Hamiltonians using a circuit with reduced depth, where few eigenvalues (instead of the full spectrum) are learned. In particular, only halved circuit depths are needed for Hamiltonians with 3-5 qubits, given in Table 3.2. Furthermore, the performance on $n = 6$ and $n = 7$-qubit Ising models, given below, is tested as well.

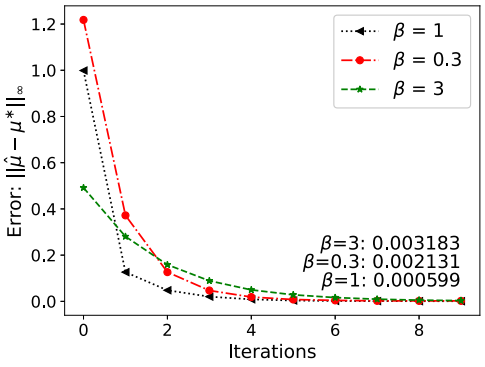
$$\text{(3.24)} \qquad H = 0.1981 \sum_{l=1}^{n} Z^l Z^{l+1} + 0.7544 \sum_{l=1}^{n} X^l.$$

To reduce the number of eigenvalues to be learned, we tune the weights $\mathbf{q}$ of the SVQE such that the $U(\boldsymbol{\theta})$ can output several smallest eigenvalues. For instance, five eigenvalues are learned for 4 & 5-qubit Ising Hamiltonians, and four eigenvalues are learned for 3-qubit Ising Hamiltonians. As a result, the circuit depth of the used $U(\boldsymbol{\theta})$ is significantly reduced. For example, we only use depth $D = 20$ to learn the coefficients

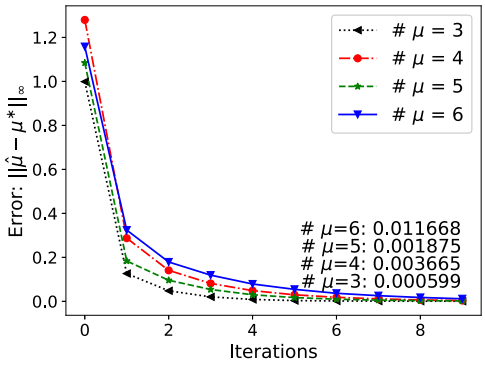
Table 3.3: Parameters setting for HQHL. The script index means the length of the tuple, e.g., $()_8$ indicates the tuple consists of 8 entries. The notation $0,\ldots$ means the entries following 0 are all zeros as well. Notation $\#\lambda$ means the number of eigenvalues we learned. Please note that we omit the $\beta = 1$ in the table.

# qubits n	weights $\mathbf{q}$	# μ	depth D	LR	$\#\lambda$
3	$(0.1, 0.2, 0.3, 0.4, 0, \ldots)_8$	6	5	0.4	4
4	$(0.1, 0.15, 0.2, 0.25, 0.3, 0, \ldots)_{16}$	8	10	0.55	5
5	$(0.1, 0.15, 0.2, 0.25, 0.3, 0, \ldots)_{32}$	10	20	0.7	5
6	$(1/21, 2/21, 3/21, 4/21, 5/21, 6/21, 0, \ldots)_{64}$	12	30	0.55	6
7	$(1/21, 2/21, 3/21, 4/21, 5/21, 6/21, 0, \ldots)_{128}$	14	40	0.6	6

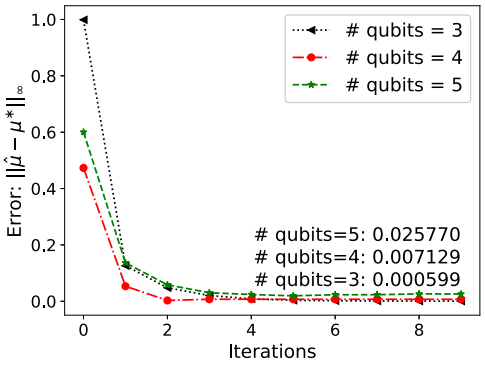
with precision 0.05 for 5-qubit Ising models. While, in Sec. 3.4.2, we use the depth $D = 40$. Moreover, we find out that using a circuit with 35 depths suffices to learn well the 6-qubit Ising model, where SVQE only learns six eigenvalues. Using the circuit with depth 40 could also reach a precision of 0.05 for the 7-qubit Ising Hamiltonian. The details of parameters setting (weights, depth, learning rate, etc.) are given in Table 3.3. Besides, the experimental results are depicted in Figure 3.4.



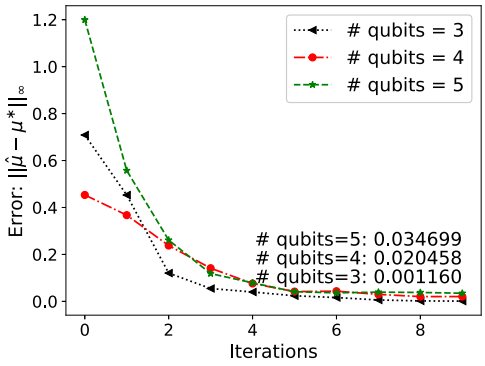
(a) Vary β



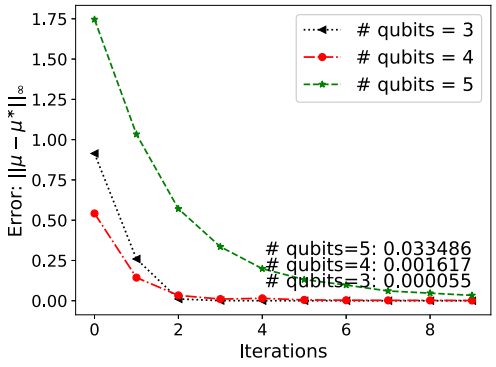
(b) Vary # μ



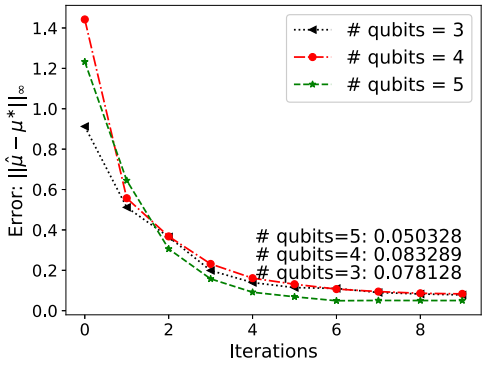
(c) Vary # qubits



(d) Ising model



(e) XY model



(f) Heisenberg model

Figure 3.3: The curves in (a), (b), (c) represent the infinity norm of the error of μ with different β, different number of μ, and different number of qubits, respectively. In (d), (e), (f), the curves represent the infinity norm of the error of μ for different many-body Hamiltonians with the number of qubits varies from 3 to 5. The numbers on the line represent the values of the last iteration. These numbers close to 0 indicate that our algorithm is effective.

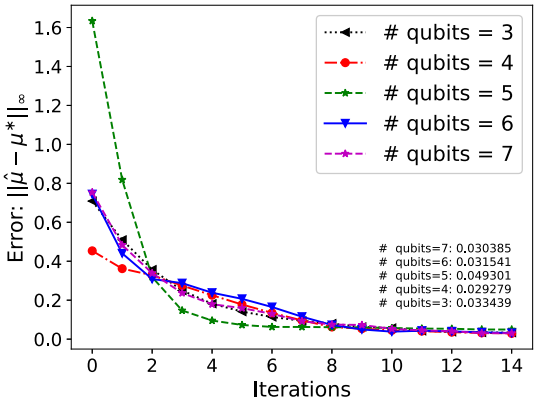


Figure 3.4: Experimental results by using fewer eigenvalues. Each line corresponds to the results by running HQHL with Ising Hamiltonians of different sizes. Results show that using halved circuit depth, compared to the setting in Sec. 3.4.2, could learn coefficients up to precision 0.05 for different sized Ising models and a different number of μ.

CHAPTER 4

QUANTUM GIBBS STATE PREPARATION

4.1 Loss function

Recall the definition of the Gibbs state for a quantum Hamiltonian H:

$$\rho_G = \frac{\exp(-\beta H)}{\mathrm{tr}(\exp(-\beta H))}. \tag{4.1}$$

A closely related concept is "free energy" of the system, which is described by a density operator ρ is given by

$$\mathcal{F}(\rho) = \mathrm{tr}(\rho H) - \beta^{-1} S(\rho), \tag{4.2}$$

where $\beta = (k_B T)^{-1}$ is the inverse temperature of the system, k_B is the Boltzmann's constant, and $S(\rho) := -\mathrm{tr}\,\rho \ln \rho$ is the von Neumann entropy of ρ. As the Gibbs state minimizes the free energy of the Hamiltonian H, it holds that

$$\rho_G = \mathrm{argmin}_\rho \mathcal{F}(\rho). \tag{4.3}$$

Therefore, if we could generate parameterized quantum states $\rho(\boldsymbol{\theta})$ and find a way to measure or estimate the loss function $\mathrm{tr}(\rho H) - k_B T \cdot S(\rho)$, then one could design variational algorithms via the optimization over $\boldsymbol{\theta}$ [36, 157].

To design a suitable and efficient variational quantum algorithm for near-term quantum devices, we design a loss function in the similar spirit of free energy. We focus

on finding the optimal parameters that minimize the free energy. The main challenge of minimizing the free energy comes from the quantum entropy estimation, which is well-known to be difficult [55]. To overcome this challenge, we truncate the Taylor series of the entropy at order K and set the truncated free energy as the loss function in our variational quantum algorithms. Explicitly, the loss function is represented as a linear combination of system's energy and higher-order state overlaps, i.e., $\mathrm{tr}(\rho^k)$, and estimate each $\mathrm{tr}(\rho^k)$ via quantum gadgets, e.g., Swap test, respectively.

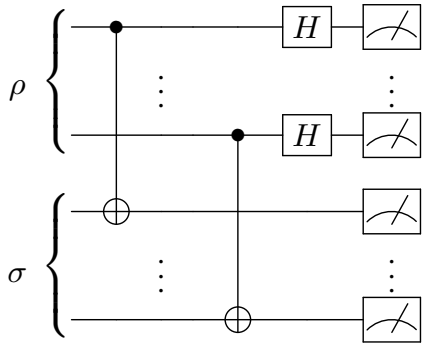


Figure 4.1: Quantum circuit for implementing Destructive Swap Test. In the circuit, two states ρ and σ are prepared at different registers. Then CNOT and Hadamard gates are performed as shown. The state overlap can be estimated via post-processing.

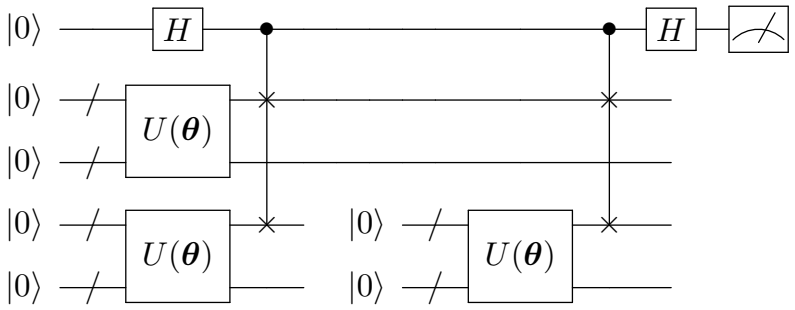


Figure 4.2: Quantum circuit for computing $\mathrm{tr}(\rho^3)$. In the circuit, the $U(\boldsymbol{\theta})$ denotes the state preparation circuit, and H denotes the Hadamard gate. Four registers are used to prepare states by $U(\boldsymbol{\theta})$, and one ancillary qubit is used to perform the controlled swap operator. The qubit reset occurs on the bottom two registers, where the break in the wire means the reset operation. Notably, the state on the bottom two registers are first implemented with a circuit $U(\boldsymbol{\theta})$ and controlled swap operator and then reset to state $|0\rangle$. Again, $U(\boldsymbol{\theta})$ and controlled swap operator are performed on the bottom registers. Finally, $\mathrm{tr}(\rho^3)$ can be obtained via post-processing the measurement results.

Algorithm 5 Variational quantum Gibbs state preparation with truncation order 2

1: Choose the ansatz of unitary $U(\boldsymbol{\theta})$, tolerance ε, truncation order 2, and initial parameters of $\boldsymbol{\theta}$;
2: Compute coefficients C_0, C_1, C_2 according to Eq. (4.6).
3: Prepare the initial states $|00\rangle$ in registers AB and apply $U(\boldsymbol{\theta})$.
4: Measure and compute $\mathrm{tr}(H\rho_{B_1})$ and compute the loss function $L_1 = \mathrm{tr}(H\rho_{B_1})$;
5: Measure and compute $\mathrm{tr}(\rho_{B_2}\rho_{B_3})$ via Destructive Swap Test and compute the loss function $L_2 = -\beta^{-1}C_1\,\mathrm{tr}(\rho_{B_2}\rho_{B_3})$;
6: Measure and compute $\mathrm{tr}(\rho_{B_4}...\rho_{B_6})$ via higher-order state overlap estimation and compute the loss function $L_3 = -\beta^{-1}C_2\,\mathrm{tr}(\rho_{B_4}...\rho_{B_6})$.
7: Perform optimization of $\mathcal{F}_2(\boldsymbol{\theta}) = \sum_{k=1}^{3} L_k - \beta^{-1}C_0$ and update parameters of $\boldsymbol{\theta}$;
8: Repeat 3-7 until the loss function $\mathcal{F}_2(\boldsymbol{\theta})$ converges with tolerance ε;
9: Output the state $\rho^{out} = \mathrm{tr}_A\, U(\boldsymbol{\theta})|00\rangle\langle 00|_{AB}\, U(\boldsymbol{\theta})^\dagger$.

4.2 Variational quantum Gibbs state preparation

In this section, we present a hybrid algorithm with the second-order loss function in Algorithm 5, and a picture for illustration is depicted in Fig. 4.3. Please refer to Appendix A.2 for the variational quantum algorithm for general truncation order K.

Clearly, Algorithm 5 can be efficiently implemented on near-term quantum devices since the estimation of loss function $\mathcal{F}_2$ only requires measuring the expected value $\langle H\rangle_\rho$, the purity or the state overlap $\mathrm{tr}(\rho^2)$, and the higher-order state overlap $\mathrm{tr}(\rho^3)$. To compute the state overlap, one approach is to utilize the well-known Swap test [26, 58], which has a simple physical implementation in quantum optics [48, 53] and can be experimentally implemented on near-term quantum hardware [74, 95, 113]. For instance, we could use a variant version of the Swap test (see Fig. 4.1), named destructive Swap test [37, 53]. Compared to the general Swap test, destructive Swap test is more practical on near term devices, since it is ancilla-free and costs less circuit depth and the number of the gates. Using the circuit in Fig. 4.1, the quantity $\mathrm{tr}(\rho^2)$ is expected to be estimated on near-term quantum hardware.

Regarding higher order state overlaps, e.g., $\mathrm{tr}(\rho^3)$, there are methods using the similar circuit to that of the destructive Swap test [134], whose depth is only 2. For more information, please refer to [134]. We can also use the qubit-efficient circuit proposed by Yirka and Yirka [159] to compute $\mathrm{tr}(\rho^k)$, for larger k. Particularly, the circuit only uses a constant number of qubits, where the key is that some subset of qubits can be reset in the course of quantum computation. For convenience, we call this method the higher-order

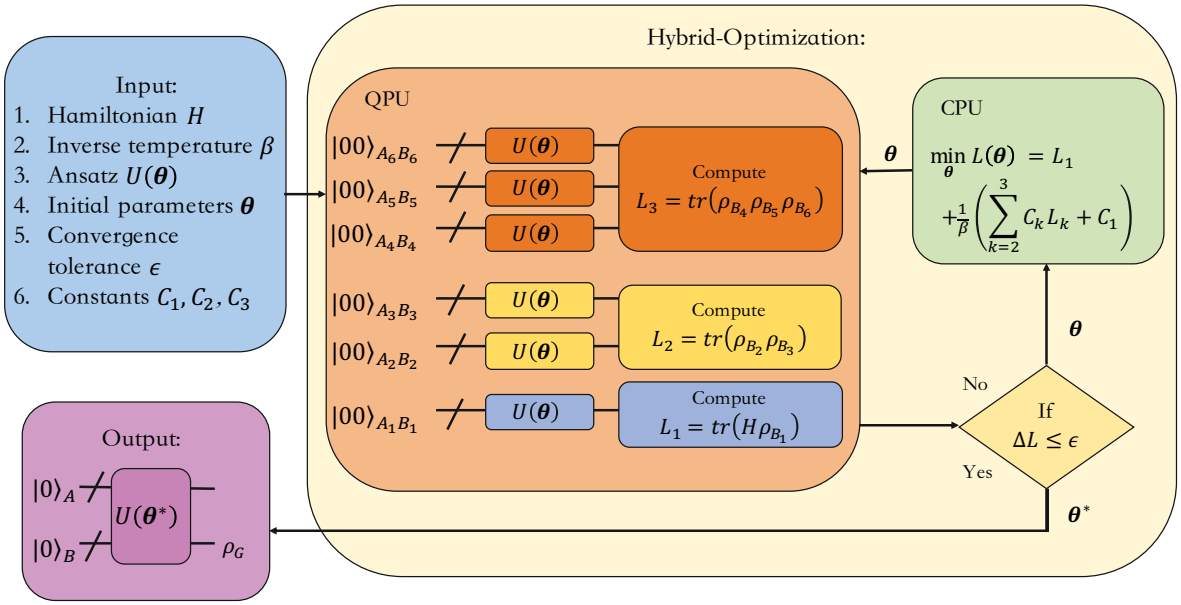


Figure 4.3: Schematic representation of the variational quantum Gibbs state preparation with truncation order 2. First, we prepare the Hamiltonian H and inverse temperature β and then send them into the Hybrid Optimization. Second, we choose an ansatz and employ it to evaluate the loss function L_1,L_2,L_3 on quantum devices. Then we calculate the difference $\Delta\mathcal{F}_2(\boldsymbol{\theta})$ by using L_1,L_2,L_3. Next, if the condition $\Delta\mathcal{F}_2 \leq \epsilon$ is not satisfied, then we perform classical optimization to update parameters $\boldsymbol{\theta}$ of the ansatz and return to the loss evaluation. Otherwise, we output the current parameters $\boldsymbol{\theta}^*$, which could be used to prepare Gibbs state ρ_G via $U(\boldsymbol{\theta})$. Here in the quantum device, registers A_2,B_2,A_3,B_3 are used to evaluate $\mathrm{tr}(\rho_{B_2}\rho_{B_3})$ and registers A_4,B_4,...,A_6,B_6 are used to evaluate $\mathrm{tr}(\rho_{B_4}\rho_{B_5}\rho_{B_6})$.

state overlap estimation and provide an example for computing $\mathrm{tr}(\rho^3)$ in Fig. 4.2. We also refer interested readers to [159] for more details on qubit-efficient algorithms for computing $\mathrm{tr}(\rho^k)$. Hence, using these qubit-efficient quantum circuits will significantly circumvent our approach's resource requirements for computing $\mathrm{tr}(\rho^k)$ for $k \geq 3$ and enable our approach to be implementable on NISQ computers.

In general, the variational quantum circuit contains a series of parameterized single-qubit Pauli rotation operators and CNOT/CZ gates alternately [13]. Here we follow this circuit pattern and mainly use Pauli-Y rotation operators and CNOT gates. For the optimization part, a variety of approaches have been proposed to optimize such variational quantum circuits, including Nelder-Mead [65, 141], Monte-Carlo [152], quasi-Newton [65], gradient descent [150], and Bayesian methods. We employ a classical optimizer to minimize the loss function via tuning the parameters and then use the

optimized circuit to prepare the target Gibbs state. For instance, we choose ADAM [86] as our gradient-based optimizer in the numerical experiments. Note that the subroutine of loss evaluation occurs on the quantum devices, and the procedure of optimization is entirely classical. Then, classical optimization tools such as gradient-based or gradient-free methods can be employed in the optimization loop.

4.3 Error analysis

In this section, we analyze the performance of our variational algorithm. Specifically, we first define a formal optimization problem that aims to find the global minimum of the truncated free energy. Second, we show that the prepared state has a higher overlap with the desired Gibbs state, using higher-order loss functions in our approach.

In our algorithm (Algorithm 8), the K-truncated-free energy $\mathcal{F}_K$ is taken as the loss function. To find the global minimum of the loss function $\mathcal{F}_K$, we update the parameters $\boldsymbol{\theta}$ till the termination condition is reached. We denote the obtained optimal parameters by $\boldsymbol{\theta}_{opt}$. Then we can prepare an approximation operator for the Gibbs state by performing the parameterized circuit $U(\boldsymbol{\theta}_{opt})$.

The loss function $\mathcal{F}_K$ is obtained by truncating the Taylor series of von Neumann entropy at order K. Specifically, let $K \in \mathbb{Z}_+$ be a positive integer, and denote the truncated entropy by $S_K(\rho)$. Let H denote the Hamiltonian and $\beta > 0$ be the inverse temperature, then the loss function $\mathcal{F}_K(\boldsymbol{\theta})$ is defined as

$$\mathcal{F}_K(\boldsymbol{\theta}) = \mathrm{tr}(H\rho(\boldsymbol{\theta})) - \beta^{-1} S_K(\rho(\boldsymbol{\theta})). \tag{4.4}$$

Here the free energy is determined by parameters $\boldsymbol{\theta}$, since the state $\rho(\boldsymbol{\theta})$ is prepared by the PQC $U(\boldsymbol{\theta})$. Particularly, the K-truncated entropy $S_K(\rho)$ is given as follows,

$$S_K(\rho) = \sum_{k=1}^{K} \frac{(-1)^k}{k} \mathrm{tr}\left((\rho - I)^k \rho\right) = \sum_{j=0}^{K} C_j \,\mathrm{tr}(\rho^{j+1}). \tag{4.5}$$

In Eq. (4.5), coefficients C_j's of $S_K(\rho)$ are given in the form below:

$$C_0 = \sum_{k=1}^{K} \frac{1}{k}, C_j = \sum_{k=j}^{K} \binom{k}{j} \frac{(-1)^j}{k}, C_K = \frac{(-1)^K}{K}. \tag{4.6}$$

where $j = 1, \ldots, K-1$.

Recall that our goal is to find parameters $\boldsymbol{\theta}_{opt}$ that minimize the value of the loss function $\mathcal{F}_K(\boldsymbol{\theta})$, i.e., $\boldsymbol{\theta}_{opt} = \mathrm{argmin}_{\boldsymbol{\theta}} \mathcal{F}_K(\boldsymbol{\theta})$. In practice, the optimization loop only terminates if some condition given previously is reached. Therefore, one cannot obtain the true

optimum, but some parameters $\boldsymbol{\theta}_0$ that will approximately minimize the loss function in the sense that

$$\mathcal{F}_K(\boldsymbol{\theta}_0) \leq \min_{\boldsymbol{\theta}} \mathcal{F}_K(\boldsymbol{\theta}) + \epsilon, \tag{4.7}$$

where ϵ is the error tolerance in the optimization problem. Especially, we assume the used PQC $U(\boldsymbol{\theta})$ endows sufficient expressiveness to prepare the desired Gibbs state or a state very close to it. Hence, the state $\rho(\boldsymbol{\theta}_0)$ could be taken to approximate the desired Gibbs state.

Since the loss function $\mathcal{F}_K(\boldsymbol{\theta})$ is a truncated version of the free energy, the solution $\boldsymbol{\theta}_0$ to the optimization problem in Eq. (4.7) is not exactly the quantum Gibbs state ρ_G. However, the obtained state $\rho(\boldsymbol{\theta}_0)$ is not far away from the Gibbs state ρ_G. Here, we use the fidelity to characterize the distance between two states. In the following, we show the validity of this claim by providing a lower bound on the fidelity between $\rho(\boldsymbol{\theta}_0)$ and ρ_G in Theorem 1. In particular, the result in Theorem 1 implies that the larger truncation order K is, the state $\rho(\boldsymbol{\theta}_0)$ is closer to the state ρ_G.

Theorem 1. *Given a positive integer K and error tolerance $\epsilon > 0$, let $\beta > 0$ be the inverse temperature, and $\boldsymbol{\theta}_0$ be the solution to the optimization in Eq.* (4.7)*. Assume the rank of the output state $\rho(\boldsymbol{\theta}_0)$ is r, then the fidelity between the state $\rho(\boldsymbol{\theta}_0)$ and the Gibbs state ρ_G is lower bounded as follows*

$$F(\rho(\boldsymbol{\theta}_0), \rho_G) \geq 1 - \sqrt{2\left(\beta\epsilon + \frac{2r}{K+1}(1-\Delta)^{K+1}\right)}, \tag{4.8}$$

where $\Delta \in (0, e^{-1})$ is a constant determined by K.

Theorem 1 implies that we can regard the output state $\rho(\boldsymbol{\theta}_0)$ as an approximation for the Gibbs state for a given error tolerance in the optimization problem and a truncation order K. And the quantity in the right-hand-side of Eq. (4.8) quantifies the extent that $\rho(\boldsymbol{\theta}_0)$ approximates ρ_G. Also, we can easily see that the quantity becomes larger when the order K increases.

Next, we prove Theorem 1 by building a connection between the relative entropy and the fidelity and then deriving an upper bound on the truncation error.

Lemma 1. *Given quantum states ρ and σ and a constant $\delta > 0$, suppose that the relative entropy $S(\rho\|\sigma)$ is less than δ, i.e., $S(\rho\|\sigma) \leq \delta$. Then the fidelity between ρ and σ is lower bounded. To be specific, $F(\rho,\sigma) \geq 1 - \sqrt{2\delta}$.*

Proof. Recall the relationship between the trace distance and the fidelity $D(\rho,\sigma) \geq 1 - F(\rho,\sigma)$, and Pinsker's inequality $D(\rho,\sigma) \leq \sqrt{2S(\rho\|\sigma)}$, then we have the following inequality,

$$F(\rho,\sigma) \geq 1 - D(\rho,\sigma) \geq 1 - \sqrt{2S(\rho\|\sigma)} \geq 1 - \sqrt{2\delta}. \tag{4.9}$$

■

Lemma 1 states that, if one wants to lower bound the fidelity $F(\rho(\boldsymbol{\theta}_0), \rho_G)$ between the obtained state $\rho(\boldsymbol{\theta}_0)$ and the Gibbs state ρ_G, then it suffices to upper bound the relative entropy $S(\rho(\boldsymbol{\theta}_0)\|\rho_G)$ between them. Thus we proceed to give an upper bound of the relative entropy.

Let δ_0 be the truncation error of $S_K(\rho)$, then the definition of the free energy allows to bound the difference between the free energy and its truncated version, i.e., $|\mathcal{F}_K(\rho) - \mathcal{F}(\rho)| \leq \beta^{-1}\delta_0$. Recall the well-known free energy equation, $\mathcal{F}(\rho) = \mathcal{F}(\rho_G) + \beta^{-1}S(\rho\|\rho_G)$, which indicates that, for arbitrary density ρ, the free energy $\mathcal{F}(\rho)$ can be represented as a linear combination of the free energy $\mathcal{F}(\rho_G)$ of the quantum Gibbs state ρ_G and the relative entropy between ρ and ρ_G. Therefore, an upper bound on the relative entropy $S(\rho(\boldsymbol{\theta}_0)\|\rho_G)$ is readily derived as follows:

$$S(\rho(\boldsymbol{\theta}_0)\|\rho_G) = \beta|\mathcal{F}(\rho(\boldsymbol{\theta}_0)) - \mathcal{F}(\rho_G)| \tag{4.10}$$

$$= \beta|\mathcal{F}(\rho(\boldsymbol{\theta}_0)) - \mathcal{F}_K(\rho(\boldsymbol{\theta}_0)) + \mathcal{F}_K(\rho(\boldsymbol{\theta}_0)) - \mathcal{F}(\rho_G)| \tag{4.11}$$

$$= \delta_0 + \beta|\mathcal{F}_K(\rho(\boldsymbol{\theta}_0)) - \mathcal{F}(\rho_G)| \tag{4.12}$$

$$\leq 2\delta_0 + \beta\epsilon, \tag{4.13}$$

where the inequality in Eq. (4.13) is due to the fact that $\mathcal{F}(\rho_G) \leq \mathcal{F}_K(\rho(\boldsymbol{\theta}_0)) \leq \mathcal{F}(\rho_G) + \beta^{-1}\delta_0 + \epsilon$, which is stated in Lemma 2. In particular, to obtain the result in Lemma 2, we assume the used PQC is expressive enough to prepare the target Gibbs state or a state very close to it.

Lemma 2. *Given the error tolerance $\epsilon > 0$ in the optimization problem in Eq. (4.7), suppose the truncation error of the free energy is $\beta^{-1}\delta_0 > 0$. Then we can derive a relation between $\mathcal{F}(\rho_G)$ and $\mathcal{F}_K(\rho(\boldsymbol{\theta}_0))$ below, where $\boldsymbol{\theta}_0$ is the output of the optimization and ρ_G is the Gibbs state.*

$$\mathcal{F}(\rho_G) \leq \mathcal{F}_K(\boldsymbol{\theta}_0) \leq \mathcal{F}(\rho_G) + \beta^{-1}\delta_0 + \epsilon. \tag{4.14}$$

Proof. First, we show that left inequality in Eq. (4.14). For arbitrary density operator ρ, we have $\mathcal{F}_K(\rho) - \mathcal{F}(\rho) > 0$. To be specific,

$$\mathcal{F}_K(\rho) - \mathcal{F}(\rho) = \beta^{-1}(S(\rho) - S_K(\rho)) \tag{4.15}$$

$$= -\beta^{-1}\operatorname{tr}\left(\sum_{j=K+1}^{\infty}\frac{(-1)^{j+1}}{j}(\rho - I)^j\rho\right) > 0 \tag{4.16}$$

In Eq. (4.16), we expand the von Neumann entropy into the Taylor series, i.e., $S(\rho) = -\operatorname{tr}\left(\sum_{j=1}^{\infty}\frac{(-1)^{j+1}}{j}(\rho - I)^j\rho\right)$, and the result holds immediately.

Second, the right inequality in Eq. (4.14) is a direct result of the definition of truncated free energy $\mathcal{F}_K(\rho)$.

$$\mathcal{F}_K(\boldsymbol{\theta}_0) - \mathcal{F}(\rho_G) = \mathcal{F}_K(\boldsymbol{\theta}_0) - \min_{\boldsymbol{\theta}}\mathcal{F}_K(\boldsymbol{\theta}) + \min_{\boldsymbol{\theta}}\mathcal{F}_K(\boldsymbol{\theta}) - \mathcal{F}(\rho_G) \tag{4.17}$$

$$\leq \epsilon + \mathcal{F}_K(\rho_G) - \mathcal{F}(\rho_G) \tag{4.18}$$

$$\leq \epsilon + \beta^{-1}\delta_0, \tag{4.19}$$

where we use the fact that $\min_{\boldsymbol{\theta}}\mathcal{F}_K(\boldsymbol{\theta}) \leq \mathcal{F}_K(\boldsymbol{\theta}_0) \leq \min_{\boldsymbol{\theta}}\mathcal{F}_K(\boldsymbol{\theta}) + \epsilon$ in Eq. (4.18), and the inequality in Eq. (4.19) is due to the truncation. Especially, we here assume the PQC endows sufficient expressiveness to prepare the desired Gibbs state or a state very close to it, which allows $\min_{\boldsymbol{\theta}}\mathcal{F}_K(\boldsymbol{\theta}) \leq \mathcal{F}_K(\rho_G)$. ■

Now, given the truncation order K, we derive an upper bound on the difference between $S_K(\rho)$ and $S(\rho)$ in the following lemma.

Lemma 3. *Given a quantum state ρ, assume the truncation order of the truncated von Neumann entropy is $K \in \mathbb{Z}_+$, and choose $\Delta \in (0, e^{-1})$ such that $-\Delta\ln(\Delta) < \frac{1}{K+1}(1-\Delta)^{K+1}$. Let δ_0 denote the truncation error, i.e., the difference between the von Neumann entropy $S(\rho)$ and its K-truncated entropy $S_K(\rho)$. Then the truncated error δ_0 is upper bounded in the sense that*

$$\delta_0 \leq \frac{r}{K+1}(1-\Delta)^{K+1}, \tag{4.20}$$

where r denotes the rank of density operator.

Proof. The proof proceeds by expanding the logarithm function in the entropy into Taylor series. The upper bound of the difference between the entropy $S(\rho)$ and its

truncated version $S_K(\rho)$ for density ρ is given as follows,

$$\delta_0 = \left|S(\rho) - S_K(\rho)\right| \tag{4.21}$$

$$= \left|\mathrm{tr}\left(\sum_{k=K+1}^{\infty} \frac{(-1)^k}{k}(\rho - I)^k \rho\right)\right| \tag{4.22}$$

$$= \left(\sum_{j:\ \lambda_j \geq \Delta} + \sum_{j:\ 0<\lambda_j<\Delta}\right) \sum_{k=K+1}^{\infty} \frac{\lambda_j}{k}(1-\lambda_j)^k. \tag{4.23}$$

In the above Eq. (4.23) we use the spectral decomposition of $\rho = \sum_{j=1}^{r} \lambda_j \left|\psi_j\right\rangle\left\langle\psi_j\right|$.

To give an upper bound on truncation error δ_0, we give upper bounds on two terms in Eq. (4.23). First, we consider the term with eigenvalues larger than Δ.

$$\sum_{j:\ \lambda_j \geq \Delta} \sum_{k=K+1}^{\infty} \frac{\lambda_j}{k}(1-\lambda_j)^k = \sum_{j:\ \lambda_j \geq \Delta} \sum_{k=K+1}^{\infty} \left[\frac{1}{k}(1-\lambda_j)^k - \frac{1}{k}(1-\lambda_j)^{k+1}\right] \tag{4.24}$$

$$\leq \sum_{j:\ \lambda_j \geq \Delta} \sum_{k=K+1}^{\infty} \frac{1}{k}(1-\lambda_j)^k - \sum_{j:\ \lambda_j \geq \Delta} \sum_{k=K+1}^{\infty} \frac{1}{k+1}(1-\lambda_j)^{k+1} \tag{4.25}$$

$$= \frac{1}{K+1} \sum_{j:\ \lambda_j \geq \Delta} (1-\lambda_j)^{K+1}. \tag{4.26}$$

The equality in Eq. (4.24) is due to the substitution of λ_j with $1-(1-\lambda_j)$, and the inequality in (4.25) follows by replacing $1/k$ with $1/(k+1)$ in the right summation of Eq. (4.24).

Then we consider the term with non-zero eigenvalues less than Δ.

$$\sum_{j:\ 0<\lambda_j<\Delta} \lambda_j \sum_{k=K+1}^{\infty} \frac{1}{k}(1-\lambda_j)^k \leq \sum_{j:\ 0<\lambda_j<\Delta} -\lambda_j \ln(\lambda_j) \tag{4.27}$$

$$\leq \sum_{j:\ 0<\lambda_j<\Delta} -\Delta \ln(\Delta), \tag{4.28}$$

where the inequality in Eq. (4.27) follows from replacing the series with $-\ln(\lambda_j)$, since function $S(x) = -x\ln(x) = \sum_{l=1}^{\infty} \frac{1}{l} x(1-x)^l$, and the second inequality is due to the fact that $S(x)$ increases as x increases in the interval $(0, e^{-1})$.

In all, an upper bound on δ_0 can be given as

$$\delta_0 \leq \frac{1}{K+1} \sum_{j:\ \lambda_j \geq \Delta} (1-\lambda_j)^{K+1} + \sum_{j:\ 0<\lambda_j<\Delta} -\Delta \ln(\Delta) \tag{4.29}$$

$$\leq r \cdot \left(\frac{r_0}{r}\frac{(1-\Delta)^{K+1}}{K+1} + \frac{r_1}{r}(-\Delta\ln(\Delta))\right) \tag{4.30}$$

$$\leq r \cdot \max\{\frac{(1-\Delta)^{K+1}}{K+1}, -\Delta\ln(\Delta)\}. \tag{4.31}$$

where r_0 (r_1) denotes the number of non-zero eigenvalues larger (less) than Δ. As $-\Delta\ln(\Delta) < \frac{1}{K+1}(1-\Delta)^{K+1}$, the claim is proved. ■

Replacing δ_0 in Eq. (4.13) with its upper bound in Eq. (4.20) immediately leads to a bound on the relative entropy $S(\rho(\boldsymbol{\theta}_0)\|\rho_G)$. Taking this bound into Lemma 1, a lower bound on the fidelity $F(\rho(\boldsymbol{\theta}),\rho_G)$ is then derived, which is exactly the one in Eq. (4.8). Now, the proof of Theorem 1 is completed. Although the choice of Δ may be demanded when the truncation order K is large, we could focus on a low-order loss function sufficient to provide a high-fidelity Gibbs state approximation. We show this in more detail in numerical experiments.

4.4 Gradient

Finding optimal parameters $\boldsymbol{\theta}_{opt}$ is a major part of our variational algorithm. Both gradient-based and gradient-free methods could be used to do the optimization. Here, we provide analytical details on the gradient-based approach, and we refer to [13] for more information on the optimization subroutines in variational quantum algorithms.

The choice of truncation order K could be various and depends on the required accuracy for Gibbs state preparation. Here we mainly focus on the two-order loss function $\mathcal{F}_2(\boldsymbol{\theta})$ as higher fidelity could be expected by the result in Theorem 1. We numerically show the validity of this choice in the next section. Particularly, the numerical results show we can use a two-order loss function to prepare high-fidelity Gibbs states of several many-body Hamiltonians.

Now, we show that $\mathcal{F}_2(\boldsymbol{\theta})$ is convex, which indicates that the gradient-based method could efficiently minimize it. We also derive the analytical expressions for its gradients and show that these analytical gradients could also be evaluated efficiently on NISQ devices. Especially, the same circuit for estimating $\mathcal{F}_2(\rho)$ can also be used to calculate their gradients.

Convexity of 2-truncated free energy Recall the definition of K-truncated entropy $S_K(\rho)$ in Eq. (4.5), and, in this section, we take $K=2$. Given truncation order 2, the loss function $\mathcal{F}_2(\rho)$ is defined in the following form:

$$\mathcal{F}_2(\boldsymbol{\theta}) = \mathrm{tr}(H\rho(\boldsymbol{\theta})) + \beta^{-1}\left(2\,\mathrm{tr}(\rho(\boldsymbol{\theta})^2) - \frac{1}{2}\,\mathrm{tr}(\rho(\boldsymbol{\theta})^3) - \frac{3}{2}\right). \tag{4.32}$$

Notice that the functional $\mathrm{tr}(H\rho)$ is linear for a given Hamiltonian H and $\beta > 0$, therefor the convexity of loss function $\mathcal{F}_2$ is determined by the convexity of the functional

$g(\rho) = 2\,\mathrm{tr}(\rho^2) - \frac{1}{2}\,\mathrm{tr}(\rho^3) - \frac{3}{2}$. Hence, to prove the convexity of $\mathcal{F}_2$, we only need to show the convexity of functional $g(\rho)$.

Lemma 4. *The functional* $g(\rho) = 2\,\mathrm{tr}(\rho^2) - \frac{1}{2}\,\mathrm{tr}(\rho^3) - \frac{3}{2}$ *is convex, where* ρ *is a density operator.*

Proof. According to Theorem 2.10 of [28], the functional $\mathrm{tr}(f(\rho))$ is convex if the associated function $f : \mathbb{R} \to \mathbb{R}$ is convex. In the scenario, where $\mathcal{F}_2(\rho)$ is given in Eq. (4.32), the associated function of g is defined as $f(x) = 2x^2 - \frac{1}{2}x^3 - \frac{3}{2}$ for all $x \in [0,1]$. The claim follows from proving that f is convex, and the second order derivative $f^{(2)}$ of f is positive, since

$$f^{(2)}(x) = 4 - 3x \geq 1 \quad \forall x \in [0,1]. \tag{4.33}$$

Therefore, the positivity of the second order derivative of f leads to the convexity of $\mathcal{F}_2(\rho)$ in the set of densities operators. ■

Analytical gradient Here we discuss the computation of the gradient of the global loss function $\mathcal{F}_2(\boldsymbol{\theta})$. Inspired by previous works [107, 112, 127], we compute the gradients of the 2-truncated free energy $\mathcal{F}_2$ by shifting the parameters of the same circuit for estimating $\mathcal{F}_2$. Note that there is an alternative method to estimate the partial derivative with a single circuit [51], but at the cost of using an ancillary qubit.

In Fig. 4.3, the density operator $\rho(\boldsymbol{\theta})$ is prepared in the register B by performing a sequence of unitaries $U = U_N ... U_1$ on the state $|00\rangle_{AB}$ in registers AB. Each gate U_m is either fixed, e.g., a C-NOT gate, or parameterized. The parameterized gates are of the form $U_m = e^{-iH_m\theta_m/2}$, where θ_m's are real parameters and H_m's are a tensor product of Pauli matrices. Then the loss function $\mathcal{F}_2$ is related to parameters $\boldsymbol{\theta}$, and its gradient is explicitly given by

$$\nabla_{\boldsymbol{\theta}} \mathcal{F}_2(\boldsymbol{\theta}) = \left(\frac{\partial \mathcal{F}_2(\boldsymbol{\theta})}{\partial \theta_1}, ..., \frac{\partial \mathcal{F}_2(\boldsymbol{\theta})}{\partial \theta_N} \right). \tag{4.34}$$

Particularly, the partial derivatives of $\mathcal{F}_2$ with respect to θ_m is

$$\begin{aligned} \frac{\partial \mathcal{F}_2(\boldsymbol{\theta})}{\partial \theta_m} =& \frac{1}{2}(\langle K \rangle_{\theta_m + \frac{\pi}{2}} - \langle K \rangle_{\theta_m - \frac{\pi}{2}}) \\ &+ \beta^{-1}[(2(\langle O \rangle_{\theta_m + \frac{\pi}{2}, \theta_m} - \langle O \rangle_{\theta_m - \frac{\pi}{2}, \theta_m}) - \frac{3}{4}(\langle G \rangle_{\theta_m + \frac{\pi}{2}, \theta_m, \theta_m} - \langle G \rangle_{\theta_m - \frac{\pi}{2}, \theta_m, \theta_m})], \end{aligned} \tag{4.35}$$

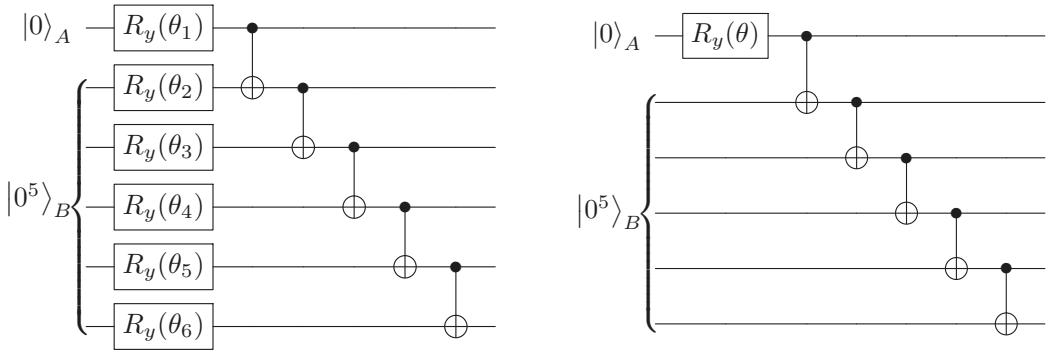
where $\langle K\rangle$, $\langle O\rangle$, and $\langle G\rangle$ are used to estimate $\mathrm{tr}(H\rho(\boldsymbol{\theta}))$, $\mathrm{tr}(\rho(\boldsymbol{\theta})^2)$, and $\mathrm{tr}(\rho(\boldsymbol{\theta})^3)$, respectively, and their definitions are given as follows:

$$\langle K\rangle_{\theta_\alpha} = \mathrm{tr}(U_\alpha \psi_{A_1B_1} U_\alpha^\dagger \cdot K), \tag{4.36}$$

$$\langle O\rangle_{\theta_\alpha,\theta_\beta} = \mathrm{tr}([\otimes_{l=2}^{3} U_\alpha \psi_{A_lB_l} U_\alpha^\dagger] \cdot O), \tag{4.37}$$

$$\langle G\rangle_{\theta_\alpha,\theta_\beta,\theta_\gamma} = \mathrm{tr}([\otimes_{l=4}^{6} U_\alpha \psi_{A_lB_l} U_\alpha^\dagger] \cdot G). \tag{4.38}$$

Here, we defer the definitions of notations K, O, G and the process of deriving the gradient to the Appendix A.4. From Eq. (4.35), we can see that the gradient can be efficiently computed by shifting the parameters in the loss function.



(a) Ansatz with 6 parameters (b) Ansatz with only 1 parameter

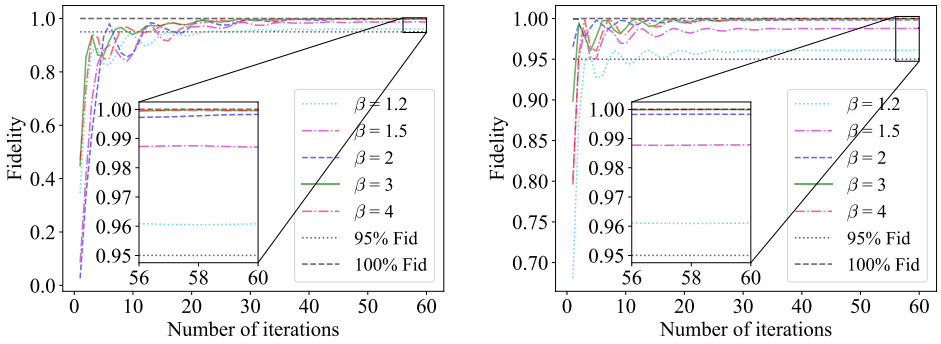
Figure 4.4: Two ansatzes for Ising chain model. These ansatzes are composed of two registers A and B, where one ancillary qubit is set in A and 5 qubits are set in B. Notably, the qubits in B are performed with rotations $R_y(\theta)$ and CNOT gates in (a), while only CNOT gates in (b).

To summarize, the above results entail that the partial derivatives of our loss function $\mathcal{F}_2(\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$ are completely determined by Eq. (4.35). This indicates that the analytical gradient of our loss function $\mathcal{F}_2(\boldsymbol{\theta})$ can be efficiently computed on near-term quantum devices by shifting parameters and performing measurements. With the analytical gradients, one could apply the gradient-based methods to minimize the loss function. Specifically, parameters $\boldsymbol{\theta}$ in the unitary $U(\boldsymbol{\theta})$ are updated towards the steepest direction of the loss function, i.e.,

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \mathcal{F}_2(\boldsymbol{\theta}), \tag{4.39}$$

where $\nabla_{\boldsymbol{\theta}} \mathcal{F}_2(\boldsymbol{\theta})$ is the gradient vector and η is the learning rate that determines the step sizes. Under suitable assumptions, the loss functions converge to the global minimum

after certain iterations of the above procedure. Notice that other gradient-based methods, e.g., stochastic gradient descent, ADAM [86], can also be used in the optimization loop of our variational Gibbs state preparation algorithm.



(a) Ansatz with 6 parameters (b) Ansatz with only 1 parameter

Figure 4.5: Fidelity curves for the Ising chain Gibbs state preparation with different β. In (a), we use the Ansatz with 6 parameters (cf. Fig. 4.4(a)); In (b), we use the Ansatz with only 1 parameter (cf. Fig. 4.4(b)). We can see that they have almost the same performance, which indicates only 1 parameter is enough for this task.

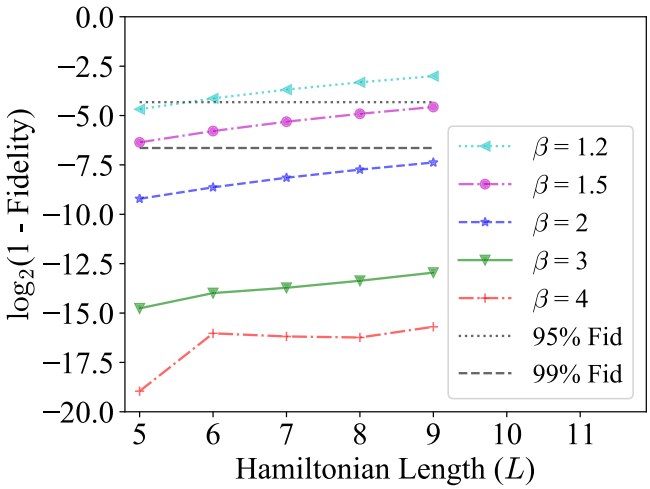


Figure 4.6: Semilog plot of the fidelity vs. the Ising Hamiltonian length (L) with different β for the Ising chain model. Here, $\log_2$ means logarithm with base 2. We can see that the fidelity increases exponentially with β growing.

4.5 Numerical results

This section has conducted numerical experiments for preparing the Gibbs states of two common Hamiltonian examples: the Ising chain model and XY spin-1/2 chain model. Specifically, in subsection 4.5.1, we study the Ising chain model and show that a parameterized circuit with one ancillary qubit and shallow depth could be trained to produce the Ising Gibbs state with fidelity higher than 99%, especially at higher β or lower inverse temperature ($\beta \geq 2$). Here, we use the order-2 loss function in our learning algorithms. Furthermore, we also give a more sophisticated ansatz with only one parameter that can do the same thing. As the second example, in subsection 4.5.2, we study the spin chain model. We show that our approach could achieve a fidelity higher than 95% via an ansatz with 30 parameters for $\beta \geq 1.5$ for a 5-length XY spin-1/2 chain Hamiltonian. In particular, the fidelity could also achieve 99% for the lower inverse temperature case. In our numerical experiments, the classical parameters of the parameterized circuits are initialized from a uniform distribution in $[0, 2\pi]$, and then updated via the ADAM gradient-based method [86] until the loss function is converged.

4.5.1 Ising model

As our first example, we consider the spin 1/2 chain B of length $L = 5$, with the Ising Hamiltonian

$$H_B = -\sum_{i=1}^{L} Z_{B,i} Z_{B,i+1} \tag{4.40}$$

and periodic boundary conditions (i.e., $Z_{B,6} = Z_{B,1}$). Our goal is to prepare the corresponding Gibbs state

$$\rho_G = e^{-\beta H_B} / \mathrm{tr}(e^{-\beta H_B}). \tag{4.41}$$

To prepare this state, we choose a 6-qubit parameterized circuit with $n_A = 1$ and $n_B = 5$ (cf. Fig. 4.4), where A is the ancillary system that used for producing a mixed state on system B. Here we need to note that we only use a 1-qubit ancillary system in our ansatz, which is significantly less than [157] where $n_A = n_B$. In Fig. 4.4(a), the ansatz consists of 6 single qubit Pauli-Y rotation operators with different classic parameters θ_i ($i \in [6]$) and 5 CNOT gates.

After applying this ansatz on the input zero state $|0\rangle_A |0^5\rangle_B$, we can get the output state on system B, which is desired to get close to the Gibbs state in Eq. (4.41). The fidelity

between this output state and the Gibbs state, in the training process with different β, is depicted in Fig. 4.5(a). When $\beta \geq 1.2$, after 30 iterations of updating parameters, our method can easily achieve a fidelity higher than 95%. Specifically, if $\beta \geq 2$, the fidelity is higher than 99%, which indicates that our approach can almost exactly prepare the Gibbs state in Eq. (4.41), especially at higher inverse temperatures.

We also test the preparation of the Ising Gibbs state for different length (i.e., $L = 5,6,7,8,9$), and all of the ansatzes are similar to Fig. 4.4(a), which only uses one additional qubit. The curves of the logarithmic form of the fidelity between the output state ρ_B and the Gibbs state ρ_G are depicted in Fig. 4.6. We can intuitively see that the larger the Hamiltonian length is, the lower fidelity we achieve. However, we can also find that the temperature has a significant impact on fidelity: the larger the β is, the higher the fidelity is (see Proposition 6 for a detailed analysis). In particular, when $\beta \geq 2$, the fidelity is already higher than 99%, for the Hamiltonian length we have listed in the figure.

An interesting experimental phenomenon in the training process is that the first parameter θ_1 in the system A is approaching $\pi/2$ while other parameters in the system B are approaching 0. Hence, we update the ansatz to a simplified one in Fig. 4.4(b) and implement the numerical simulations in Fig. 4.5(b). Notably, the overall performance is almost the same as using the ansatz with 6 parameters (cf. Fig.4.5(a)). To further explore this interesting behaviour of the Ising chain Gibbs state preparation, we analyze the states generated using different loss functions.

Proposition 5. *Given the circuit in Fig. 4.4(b) and denote $\rho_B(\theta)$ its output state on system B. For the Ising chain model, if we compute its free energy in Eq.* (4.2) *and our truncated cost in Eq.* (4.4)*, then the optimal parameters that minimize these two loss functions are both*

$$\theta = \pi/2 + k\pi, \tag{4.42}$$

where $k \in \mathbb{Z}$. As a result, $\rho_B(\pi/2)$ is the best state, under this circuit, that approaches the Gibbs state in Eq. (4.41)*, with a fidelity larger than 95% for any $\beta \geq 1.25$.*

Proof. This claim can be directly derived by computing the global minimum of both loss functions $\mathcal{F}$ and $\mathcal{F}_2$ using the ansatz in Fig. 4.4(b). Assuming $n_A = 1$, $n_B = n$ and denoting the output state as $|\psi\rangle_{AB}$, we can easily obtain the state ρ_B as

$$\rho_B(\theta) = \mathrm{tr}_A(|\psi\rangle\langle\psi|_{AB}) \tag{4.43}$$

$$= \cos^2(\theta/2)\left|0^n\right\rangle\left\langle 0^n\right|_B + \sin^2(\theta/2)\left|1^n\right\rangle\left\langle 1^n\right|_B. \tag{4.44}$$

To compute the derivatives of $\mathcal{F}(\theta)$ and $\mathcal{F}_2(\theta)$, we first present their explicit expressions.

$$\begin{aligned}\mathcal{F}(\theta) &= \mathrm{tr}(H_B\rho_B(\theta)) - \beta^{-1}S(\rho_B(\theta)) \\ &= \lambda_0 a + \lambda_1 b + \beta^{-1}[a\ln(a) + b\ln(b)], \end{aligned} \tag{4.45}$$

$$\begin{aligned}\mathcal{F}_2(\theta) &= \mathrm{tr}(H_B\rho_B(\theta)) + \beta^{-1}\left[2\,\mathrm{tr}(\rho_B(\theta)^2) - \frac{1}{2}\mathrm{tr}(\rho_B(\theta)^3) - \frac{3}{2}\right] \\ &= \lambda_0 a + \lambda_1 b + \beta^{-1}\left[2a^2 + 2b^2 - \frac{a^3+b^3}{2} - \frac{3}{2}\right], \end{aligned} \tag{4.46}$$

where we denote $\cos^2(\theta/2)$ by a and $\sin^2(\theta/2)$ by b, and λ_0 and λ_1 are the eigenvalues of H associated with eigenvectors $|0^n\rangle$ and $|1^n\rangle$, respectively.

Actually, in the Ising chain model, λ_0 and λ_1 are equal (c.f. Lemma 5). Thus, derivatives of $\mathcal{F}(\rho_B(\theta))$ and $\mathcal{F}_2(\rho_B(\theta))$ with respect to θ have the following form.

$$\partial_\theta \mathcal{F}(\theta) = \frac{\beta^{-1}}{2}\sin(\theta)(\ln(b) - \ln(a)) \tag{4.47}$$

$$\partial_\theta \mathcal{F}_2(\theta) = \frac{5\beta^{-1}}{2}\sin(\theta)(b-a). \tag{4.48}$$

From Eqs. (4.47) (4.48), the global minimums of $\mathcal{F}$ and $\mathcal{F}_2$ are

$$\theta = \frac{\pi}{2} + k\pi \quad \forall k \in \mathbb{Z}. \tag{4.49}$$

The fidelity between $\rho_B(\pi/2)$ and ρ_G could be derived from Proposition 6 and Lemma 5, where if we set $N = 2^5$, $\Delta = 4$ and $\beta = 1.25$ and then we get $F(\rho_B(\pi/2), \rho_G) \geq 95.3\%$. Hence, we could achieve a fidelity higher than 95%, provided that $\beta \geq 1.25$. ■

Here, we need to note that this fidelity is just a lower bound, actually, when $\beta = 1.2$, we have still achieved a fidelity greater than 95% for $n_B = 5$, as demonstrated in our experiments. And in Proposition 5, the number of qubits in system B is not limited to 5, instead it can be any positive integer that greater than two.

Another interesting experimental result (c.f. Fig. 4.6) shows that the fidelity between the Ising chain Gibbs state ρ_G and the output state ρ_B of our method increases exponentially when β increases. The result is shown below.

Proposition 6. *Given the circuit in Fig. 4.4(b) and let $\rho_B(\theta)$ be its output state on system B. Then the fidelity between $\rho_B(\pi/2)$ and the Gibbs state ρ_G is lower bounded. To be more specific,*

$$F(\rho_B(\pi/2), \rho_G) \geq \frac{1}{\sqrt{1 + (N/2-1)e^{-\beta\Delta}}}, \tag{4.50}$$

where N is the dimension of system B, i.e., $N = 2^{n_B}$ and Δ is the spectral gap of the Hamiltonian H_B on system B, i.e., the discrepancy between the minimum and the second minimum eigenvalues.

Proof. To prove this result, we assume the eigenvalues, associated with the eigenvectors $|0\rangle$, $|1\rangle$, …, $|N-1\rangle$, for the Hamiltonian H are denoted by λ_0, λ_1, …, λ_{N-1}. Specifically, eigenvalues λ_0 and λ_{N-1} are associated with eigenvectors $|0^n\rangle$ and $|1^n\rangle$, respectively. A key feature of the Ising model is that $\lambda_0 = \lambda_{N-1}$ and they are minimum among all eigenvalues and let Δ denote the spectral gap of the Hamiltonian H_B, which implies $\lambda_j - \lambda_0 \geq \Delta$, where $j \neq 0, N-1$.

Let $\hat{\lambda}_j$ denote the eigenvalues of the Gibbs state. Then, according to the definition of Gibbs state, $\hat{\lambda}_j$ have the following form.

$$\hat{\lambda}_j = \frac{e^{-\beta\lambda_j}}{Z}. \tag{4.51}$$

where $Z = \sum_{l=0}^{N-1} e^{-\beta\lambda_l}$.

Next, we derive bounds on eigenvalues $\hat{\lambda}_0$ and $\hat{\lambda}_{N-1}$. Note that, in the Ising model, eigenvalues λ_0 and λ_{N-1} are equal, then the associated eigenvalues $\hat{\lambda}_0 = \hat{\lambda}_{N-1}$, and they have the explicit forms in the following.

$$\hat{\lambda}_0 = \hat{\lambda}_{N-1} = \frac{e^{-\beta\lambda_0}}{Z} \tag{4.52}$$

$$= \frac{1}{2 + \sum_{j\neq 0,N-1} e^{\beta(\lambda_0 - \lambda_j)}} \tag{4.53}$$

$$\geq \frac{1}{2 + (N-2)e^{-\beta\Delta}}. \tag{4.54}$$

Recall the output state of our algorithm is $\rho_B(\pi/2)$ in Proposition 5. The inequality in Eq. (4.50) is immediately acquired by calculating the fidelity $F(\rho_B(\pi/2), \rho_G)$:

$$F(\rho_B(\pi/2), \rho_G) = \mathrm{tr}\sqrt{\rho_B^{1/2}(\pi/2)\rho_G\rho_B^{1/2}(\pi/2)} \tag{4.55}$$

$$= \mathrm{tr}\sqrt{1/\sqrt{2}\cdot\hat{\lambda}_0\cdot 1/\sqrt{2}\,|0^n\rangle\langle 0^n| + 1/\sqrt{2}\cdot\hat{\lambda}_{N-1}\cdot 1/\sqrt{2}\,|1^n\rangle\langle 1^n|} \tag{4.56}$$

$$= \sqrt{2\hat{\lambda}_0} \tag{4.57}$$

$$\geq \frac{1}{\sqrt{1 + (N/2-1)e^{-\beta\Delta}}}. \tag{4.58}$$

This completes the proof. ■

The following lemma states some facts about the Ising model, which are helpful for the above proofs.

Lemma 5. *Given an Ising model Hamiltonian in Eq.* (4.40)*, the eigenvalues* λ_0 *and* λ_{N-1}*, associated with eigenvectors* $|0^{n_B}\rangle$ *and* $|1^{n_B}\rangle$*, are equal, i.e.,* $\lambda_0 = \lambda_{N-1} = -L$*. Particularly, the spectral gap is exactly* 4 *for all* $n_B \geq 2$*.*

Proof. To prove that eigenvalues of $|0^{n_B}\rangle$ and $|1^{n_B}\rangle$ are equal, we compute their corresponding eigenvalues of each term $Z_{B,i}Z_{B,i+1}$. Notice that, for all $i = 1,...,L$,

$$Z_{B,i}Z_{B,i+1}|0^{n_B}\rangle = |0^{n_B}\rangle, \tag{4.59}$$

$$Z_{B,i}Z_{B,i+1}|1^{n_B}\rangle = |1^{n_B}\rangle. \tag{4.60}$$

Hence, in the Ising model, the eigenvalues of $|0^{n_B}\rangle$ and $|1^{n_B}\rangle$ are $-L$.

As for the rest eigenvectors $|j\rangle$, $j \neq 0, N-1$, their eigenvalues of $Z_{B,i}Z_{B,i+1}$ are given as follows.

$$Z_{B,i}Z_{B,i+1}|j\rangle = (-1)^{k_i+k_{i+1}}|j\rangle, \tag{4.61}$$

where k_i and k_{i+1} are the bits in the i-th and $(i+1)$-th position of $|j\rangle$. Particularly, in the Ising model, the eigenvalue of $|j\rangle$ is represented as $-\sum_{i=1}^{L}(-1)^{k_{B,i}+k_{B,i+1}}$. To be specific,

$$H_B|j\rangle = -\sum_{i=1}^{L}(-1)^{k_i+k_{i+1}}|j\rangle. \tag{4.62}$$

Overall, the eigenvalues of $|j\rangle$ are larger than $-L$, which implies the eigenvalues of $|0^{n_B}\rangle$ and $|1^{n_B}\rangle$ are minimum.

Now we show that the spectral gap of the Ising model with more than two qubits is 4. The minimum eigenvalue of H_B is $-L$ means that $k_{B,i} + k_{B,i+1} = 0/2$ for all i, and hence the ground states are $|0^{n_B}\rangle$ and $|1^{n_B}\rangle$. If we flip one qubit of the eigenvector $|j\rangle$, then two terms like $(-1)^{k_{B,i}+k_{B,i+1}}$ of its eigenvalues will change by 2. If we flip more qubits, then more terms will change. Notice that the eigenvectors with an eigenvalue larger than $-L$ will differ from those with minimum eigenvalue at least one qubit, resulting in at least two terms change. Then, the overall difference between the minimum and second minimum eigenvalue is at least 4. Clearly, the difference of 4 can be obtained, and then the spectral gap is 4. ■

Notably, our approach prepare Gibbs state with high accuracy when β is large, as the fidelity $F(\rho_B(\pi/2), \rho_G)$ converges fast to 1 with β increasing.

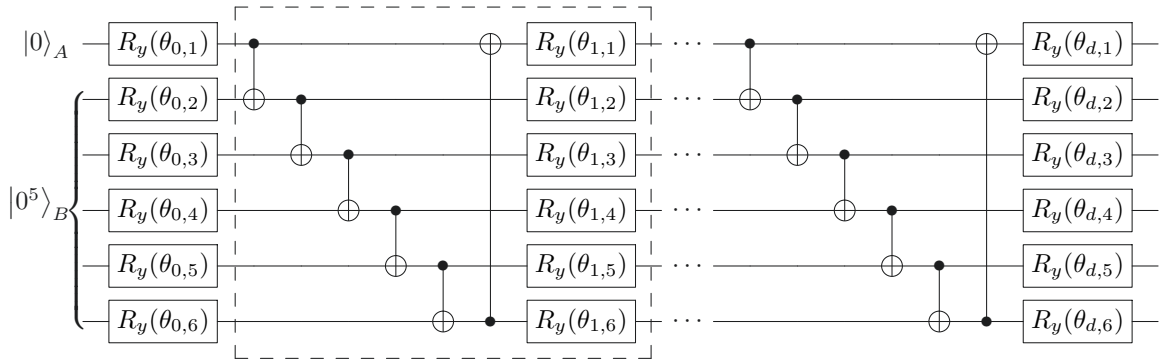


Figure 4.7: The ansatz for XY spin-1/2 chain model. In this ansatz, it contains one ancilla qubit in register A and 5 qubits in register B. Rotation gates $R_y(\theta)$ are first applied on all qubits. Then, a basic circuit module (denoted in the dashed-line box) composed of CNOT gates and rotation gates $R_y(\theta)$ is repeatedly applied. Here, d means repeating d times.

4.5.2 XY spin-1/2 chain model

Our second instance is the XY spin-1/2 chain B of length $L = 5$, with the Hamiltonian

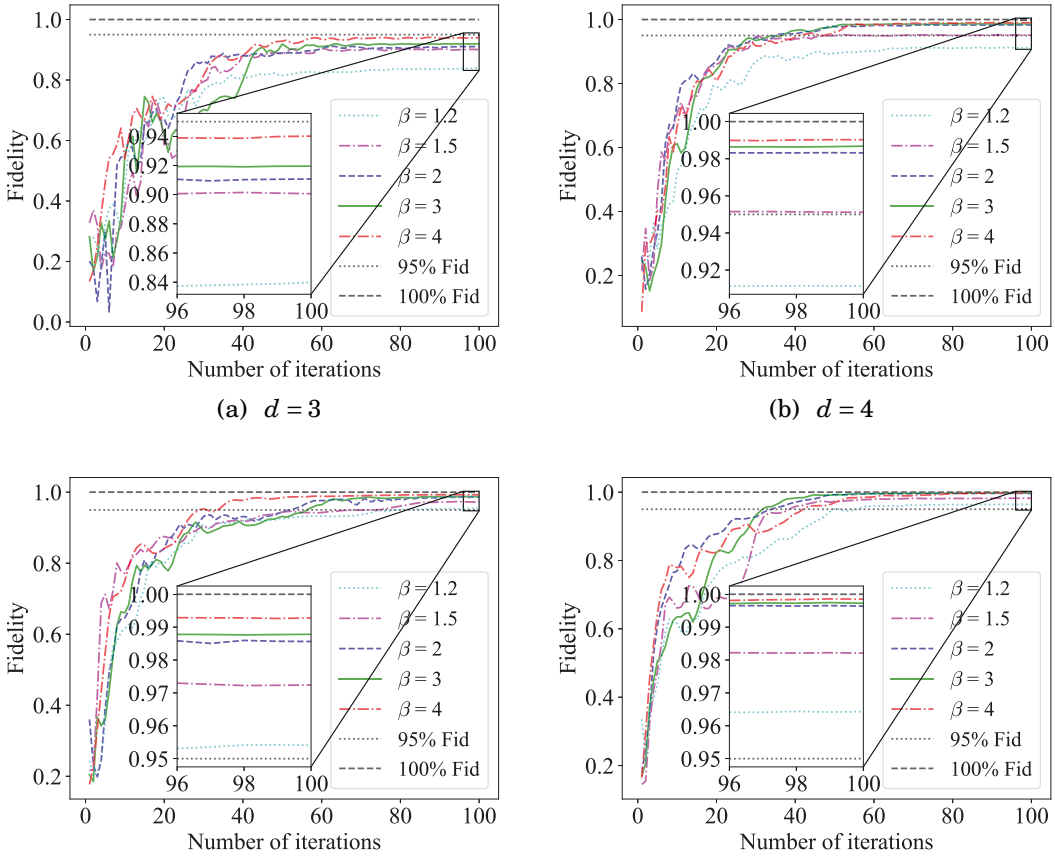
$$H_B = -\sum_{i=1}^{L} X_{B,i}X_{B,i+1} + Y_{B,i}Y_{B,i+1} \tag{4.63}$$

and periodic boundary conditions (i.e., $Z_{B,6} = Z_{B,1}$).

To prepare the spin chain Gibbs state, we first choose a 6-qubit parameterized circuit with $n_A = 1$ and $n_B = 5$ (cf. Fig. 4.7), where the basic circuit module (which contains a CNOT layer and a layer of single-qubit Pauli-Y rotation operators) is repeated d times, and the total number of parameters of this circuit is $(n_A + n_B)(d + 1)$.

The fidelity between the output state of this circuit and the Gibbs state is shown in Fig. 4.8, where different d's are included. We see that when $d \geq 4$ and $\beta \geq 1.5$, our approach can easily achieve a fidelity greater than 95% and if $\beta \geq 2$, the fidelity could be higher than 98%. Furthermore, if β is equaling to 4, the fidelity can be even higher than 99%, which means our approach could almost generate the Gibbs state exactly in higher β (or lower temperature). One possible reason that we need larger d for this instance is that the Hamiltonian in Eq. (4.63) is not directly generated via the CNOT module. Hence we will need multiple CNOT modules to fully entangle the state.

We need to note that the above experiments mainly focus on lower temperatures, i.e., $\beta > 1$, where smaller ancillary systems and lower truncation order $K = 2$ are usually sufficient to achieve a higher fidelity. In order to test our algorithm's performance with



(a) $d = 3$ (b) $d = 4$

(c) $d = 5$ (d) $d = 6$

Figure 4.8: Fidelity curves for the XY spin-1/2 chain Gibbs state preparation with different β. The results of the fidelity obtained with different β are represented by coloured lines. In (a)-(d), numerical experiments are performed using different ansatzes. In each ansatz, the basic circuit module (cf. Fig. 4.7) is repeated different times, i.e., d. Note that each ansatz has $(n_A + n_B)(d+1) = 6(d+1)$ parameters. Here better performance are obtained with larger d.

different truncation order K under higher temperatures (e.g., $\beta < 0.5$), we choose a 6-qubit parameterized circuit with $n_A = 3$ and $n_B = 3$ for a 3-length XY spin-1/2 chain Hamiltonian. Here, the ansatz is similar to Fig. 4.7 and we set $d = 8$ to make it expressive enough. $n_A = n_B$ means the ancillary systems can cover all the Hamiltonian's ranks. The boxplots of the fidelity versus various truncation order K are illustrated in Fig. 4.9, where three higher temperatures ($\beta = 0.1, 0.2, 0.3$) are included. From the results, we could intuitively see that the larger truncation order, the higher fidelity we achieve, which is in line with our theoretical analysis. And the phenomenon that we could still achieve a fidelity higher than 98% even under higher temperatures indicates our hybrid algorithm

has a powerful ability in preparing Gibbs states of certain many-body Hamiltonians.

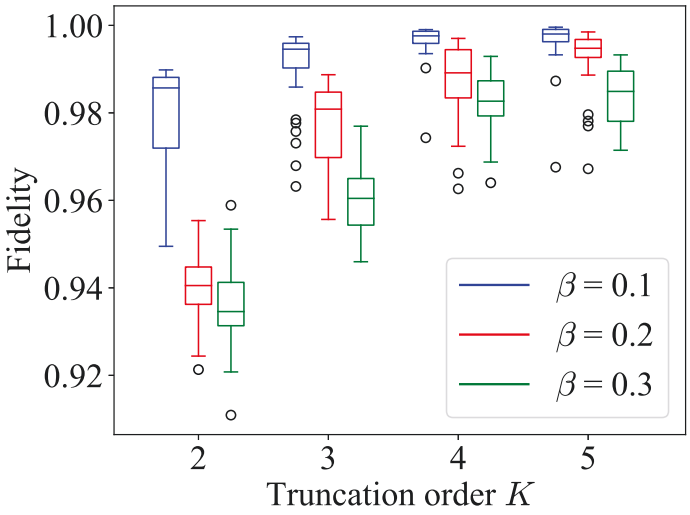


Figure 4.9: Boxplot of the fidelity vs. the truncation order K with different β for the XY spin-1/2 chain model. Here the ansatz is similar to Fig. 4.7 while $n_A = n_B = 3$. Each box consists of 30 runs with different parameter initializations.

CHAPTER 5

QUANTUM ENTROPY ESTIMATION

In this chapter, we consider estimating quantum entropies of an unknown quantum state on quantum computers. To begin with, we assume that copies of the input state ρ can be accessed and have no constraint on the number of copies. Then, we formally state the task of estimating the quantum entropies below.

Definition 3 (Entropy estimation)**.** *Given access to the copies of a quantum state $\rho \in \mathbb{C}^{2^n \times 2^n}$, the aim is to estimate the von Neumann entropy $S(\rho) = -\operatorname{tr}(\rho \ln \rho)$ and α-Rényi entropy $R_\alpha(\rho) = \frac{1}{1-\alpha} \log \operatorname{tr}(\rho^\alpha)$. To be more specific, find $S(\rho)_{est}$ and $R_\alpha(\rho)_{est}$ such that, for any constant $\alpha \in (0,1) \cup (1,+\infty)$,*

$$\Pr\left[\left|S(\rho)_{est} - S(\rho)\right| \leq \epsilon\right] \geq 1-\delta, \tag{5.1}$$

$$\Pr[|R_\alpha(\rho)_{est} - R_\alpha(\rho)| \leq \epsilon] \geq 1-\delta, \tag{5.2}$$

where $\epsilon \in (0,1)$ and $\delta \in (0,1)$ denote the estimation accuracy and the failure probability, respectively.

The main idea of our approaches is to find a Fourier series approximation of the entropy and evaluate the Fourier series by constructing explicit quantum circuits. Overall speaking, we establish the following: In Sec. 5.1, we propose the Fourier series as approximations of the von Neumann and Rényi entropies, which means we decompose the von Neumann and quantum Rényi entropy into the combination of many terms that are easy to estimate. In Sec. 5.2, we provide explicit quantum circuits to evaluate the Fourier series of the entropy approximations. In Sec. 5.3, combining the Fourier series

approximation and explicit circuit schemes, we propose near-term quantum algorithms for estimating von Neumann and quantum Rényi entropies.

5.1 Quantum entropy approximations

This section presents Fourier series approximations of $S(\rho)$ and $R_\alpha(\rho)$, i.e., a combination of terms that can be evaluated on quantum computers.

5.1.1 Approximation of von Neumann entropy

To provide the Fourier series, we follow the method given in Lemma 37 of [140] to contruct the Fourier series from a truncated Taylor series. Here, we use the truncated Taylor series of $S(\rho)$ shown below.

$$S(\rho) \approx \sum_{k=1}^{K} \frac{1}{k} \mathrm{tr}\left(\rho(I-\rho)^k\right), \tag{5.3}$$

where integer K is the truncation order determining the accuracy. The larger K gives a more accuracy entropy $S(\rho)$. The choice of truncation order can be determined by the rank of the state, which can be found in an updated version of [149]. Here we give a choice of truncation order depending on the eigenvalue lower bound.

Lemma 6. *Suppose the minimal non-zero eigenvalue of state ρ is at least Λ, then there exists an integer K such that*

$$\left| S(\rho) - \sum_{k=1}^{K} \frac{1}{k} \mathrm{tr}(\rho(I-\rho)^k) \right| \leq \epsilon, \tag{5.4}$$

where $K \in \Theta(\log(1/\epsilon\Lambda)/\log(1/(1-\Lambda))$, and $\sum_{k=1}^{K} \frac{1}{k} \leq \log(K+1)+1$.

Proof. First, we assume ρ has a spectral decomposition $\rho = \sum_j \lambda_j \left|\mathbf{e}_j\right\rangle\left\langle\mathbf{e}_j\right|$. Then the truncation error is given by

$$error = \left| \sum_{k=K+1}^{\infty} \frac{1}{k} \mathrm{tr}\left(\rho(I-\rho)^k\right) \right| \tag{5.5}$$

$$= \sum_{k=K+1}^{\infty} \sum_j \frac{1}{k} \lambda_j (1-\lambda_j)^k \tag{5.6}$$

$$\leq \sum_{k=K+1}^{\infty} \frac{1}{k} (1-\Lambda)^k \tag{5.7}$$

$$\leq \frac{(1-\Lambda)^{K+1}}{\Lambda(K+1)}. \tag{5.8}$$

Here we have used facts that $\sum_j \lambda_j = 1$ and $\lambda_j \geq \Lambda$.

To suppress the truncation error to ϵ, we choose $K \in O(\log(1/\epsilon\Lambda)/\log(1/(1-\Lambda)))$. This can be easily verified. As we aim to find K such that $(1-\Lambda)^{K+1}/\Lambda(K+1) \leq \epsilon$, we take logarithm on both sides. Then we have

$$(K+1)\log(1-\Lambda) - \log(K+1) \leq \log(\epsilon\Lambda). \tag{5.9}$$

Multiply -1 on both sides.

$$(K+1)\log(1/(1-\Lambda)) + \log(K+1) \geq \log(1/\epsilon\Lambda). \tag{5.10}$$

Clearly, $\log(K+1) \geq 0$. We then choose K such that $(K+1)\log(1/(1-\Lambda))\log(4/\epsilon\Lambda)$, immediately leading to the desired result.

Finally, we derive an upper bound on the Harmonic number $\sum_{k=1}^{K} \frac{1}{k}$. Notice that

$$\frac{1}{k} = \frac{1}{k} \cdot 1 \leq \int_{x=k}^{k+1} \frac{1}{x} \mathrm{d}x + 1 \cdot (\frac{1}{k} - \frac{1}{k+1}). \tag{5.11}$$

Here, $\frac{1}{k} \cdot 1$ denotes the area of a rectangle. Accordingly, the R.H.S denotes the area of two regions, one of which is determined by $1/x$, varying from $1/k$ to $1/k+1$, and the other is a rectangle with height $1/k - 1/(k+1)$ and width 1. Thus,

$$\sum_{k=1}^{K} \frac{1}{k} \leq \sum_{k=1}^{K} \left[\int_{x=k}^{k+1} \frac{1}{x} \mathrm{d}x + 1 \cdot (\frac{1}{k} - \frac{1}{k+1}) \right] \tag{5.12}$$

$$= \int_{x=1}^{K+1} \frac{1}{x} \mathrm{d}x + \sum_{k=1}^{K} (\frac{1}{k} - \frac{1}{k+1}) \tag{5.13}$$

$$\leq \ln(K+1) + 1. \tag{5.14}$$

Finally, the proof is completed. ■

Using this Taylor series, we can find a Fourier series approximation $S(\rho)_{est}$, which is presented in Lemma 7.

Lemma 7. *For arbitrary quantum state $\rho \in \mathbb{C}^{2^n \times 2^n}$, let Λ be the lower bound on all non-zero eigenvalues of ρ. There exists a Fourier series $S(\rho)_{est}$ such that $\left|S(\rho) - S(\rho)_{est}\right| \leq \epsilon$ for any $\epsilon \in (0,1)$, where*

$$S(\rho)_{est} = \sum_{l=0}^{\lfloor L \rfloor} \sum_{s=D_l}^{U_l} \sum_{k=1}^{K} \frac{b_l^{(k)} \binom{l}{s}}{k 2^l} \mathrm{tr}(\rho \cos(\rho \cdot t(s,l))). \tag{5.15}$$

Particularly, coefficient $U_l = \min\{l, \lceil \frac{l}{2} \rceil + M_l\}$ *and* $D_l = \max\{0, \lfloor \frac{l}{2} \rfloor - M_l\}$*, and* $\binom{l}{s}$ *denotes the binomial coefficient. Meanwhile, coefficients* $t(s,l) = (2s-l)\pi/2$*, and coefficients* K, L, M_l *are given by*

$$K \in \Theta\left(\frac{\log(\epsilon\Lambda)}{\log(1-\Lambda)}\right), \quad L = \ln\left(\frac{4\sum_{k=1}^{K} 1/k}{\epsilon}\right)\frac{1}{\Lambda^2}, \quad M_l = \left\lceil \sqrt{\ln\left(\frac{4\sum_{k=1}^{K} 1/k}{\epsilon}\right)\frac{l}{2}} \right\rceil. \tag{5.16}$$

For any $k = 1, \ldots, K, l = 0, \ldots, \lfloor L \rfloor$*, the coefficients* $b_l^{(k)}$ *are positive and defined inductively. Explicitly,*

(5.17)

$$b_l^{(1)} = 0, \quad \textit{if } l \textit{ is even}, \quad b_l^{(1)} = \frac{2\binom{l-1}{(l-1)/2}}{\pi 2^{l-1} l}, \quad \textit{if } l \textit{ is odd}, \quad b_l^{(k+1)} = \sum_{l'=0}^{l} b_{l'}^{(k)} b_{l-l'}^{(1)}, \quad \forall k \geq 1.$$

Moreover, overall weights of $S(\rho)_{est}$ *is bounded as follows.*

$$\sum_{l=0}^{\lfloor L \rfloor} \sum_{s=D_l}^{U_l} \sum_{k=1}^{K} \frac{b_l^{(k)} \binom{l}{s}}{2^l k} \in O(\log(K)). \tag{5.18}$$

Proof. To prove the result in the Lemma, we only focus on the Taylor series of the natural logarithm. By Lemma 6, for $K \in \Theta(\log(1/\epsilon)/\log(1/(1-\Lambda))$,

$$\left| -\ln(x) - \sum_{k=1}^{K} \frac{1}{k}(1-x)^k \right| \leq \frac{\epsilon}{4}. \tag{5.19}$$

We use the method in Lemma 37 of [140] to construct the Fourier series. The process proceeds as three major steps: 1. Transform the Taylor series into a linear combination of cosines. 2. Truncate the series to obtain a high-precision approximation. 3. Substitute all cosines with complex exponents.

Step 1. As all eigenvalues λ of ρ in $[0,1]$, the identity below holds.

$$1 - \lambda = \frac{\arcsin(\sin((1-\lambda)\pi/2))}{\pi/2} = \frac{\arcsin(\cos(\lambda\pi/2))}{\pi/2}. \tag{5.20}$$

Substitute λ with x in the L.H.S of Eq. (5.19) and $1-\lambda$ with $\arcsin(\cos(\lambda\pi/2))/\pi/2$.

$$\left| -\ln(\lambda) - \sum_{k=1}^{K} \frac{1}{k}\left(\frac{\arcsin(\cos(\lambda\pi/2))}{\pi/2}\right)^k \right| \leq \frac{\epsilon}{4}. \tag{5.21}$$

Recall the Taylor series of $\arcsin(y)$, for all $y \in [-1,1]$,

$$\arcsin(y) = \sum_{l=0}^{\infty} \frac{(2l)!}{4^l (l!)^2 (2l+1)} y^{2l+1} = y + \frac{y^3}{6} + \frac{3y^5}{40} + \ldots \tag{5.22}$$

Using the Taylor series of $\arcsin(y)$, we deduce the Taylor series of $(\arcsin(y)/\pi/2)^k$, given by

$$\left(\frac{\arcsin(y)}{\pi/2}\right)^k = \sum_{l=0}^{\infty} b_l^{(k)} y^l. \tag{5.23}$$

Particularly, the coefficients $b_l^{(k)}$ could be inductively computed. Specifically speaking,

$$\left(\frac{\arcsin(y)}{\pi/2}\right)^{k+1} = \left(\frac{\arcsin(y)}{\pi/2}\right)^k \left(\sum_{l=0}^{\infty} b_l^{(1)} y^l\right). \tag{5.24}$$

For $k=1$, all $b_l^{(1)}$ correspond to the coefficients of the Taylor series of arcsin times $\frac{2}{\pi}$. That is,

$$b_{2l}^{(1)} = 0, \quad b_{2l+1}^{1} = \frac{2}{\pi}\frac{(2l)!}{4^l (l!)^2 (2l+1)} = \frac{2/\pi}{4^l (2l+1)}\binom{2l}{l}. \tag{5.25}$$

For general k, suppose all coefficients $b_l^{(k)}$ are obtained, then the coefficients $b_l^{(k+1)}$ are computed as follows:

$$b_l^{(k+1)} = \sum_{l'=0}^{l} b_{l'}^{(k)} b_{l-l'}^{(1)}. \tag{5.26}$$

So, we could inductively compute all coefficients $b_l^{(k)}$ using Eqs. (5.25)-(5.26).

Consequently, we have deducted a series to approximate $-\ln(\lambda)$.

$$\left| -\ln(\lambda) - \sum_{k=1}^{K} \frac{1}{k} \sum_{l=0}^{\infty} b_l^{(k)} \cos^l(\lambda\pi/2) \right| \leq \frac{\epsilon}{4}. \tag{5.27}$$

Step 2. Now, we truncate the series in Eq. (5.27) to obtain a high-precision approximation series.

First, we truncate the infinity series at order $L = \ln\left(\frac{4\sum_{k=1}^{K} 1/k}{\epsilon}\right)\frac{1}{\Lambda^2}$. Next, we show the

truncation error.

$$\left|\sum_{l=\lceil L\rceil}^{\infty} b_l^{(k)} \cos^l(\lambda\pi/2)\right| \le \sum_{l=\lceil L\rceil}^{\infty} b_l^{(k)} \left|\cos^l(\lambda\pi/2)\right| \tag{5.28}$$

$$\le \cos^L(\lambda\pi/2) \sum_{l=\lceil L\rceil}^{\infty} b_l^{(k)} \tag{5.29}$$

$$\le \cos^L(\lambda\pi/2) \tag{5.30}$$

$$= \sin^L((1-\lambda)\pi/2) \tag{5.31}$$

$$\le (1-\lambda^2)^L \tag{5.32}$$

$$\le e^{-\lambda^2 L} \tag{5.33}$$

$$\le e^{-\Lambda^2 L} \tag{5.34}$$

$$\le \frac{\epsilon}{4\sum_{k=1}^{K} 1/k}. \tag{5.35}$$

Here, we have used facts that $b_l^{(k)} \ge 0$, $\sum_{l=1}^{\infty} b_l^{(k)} = 1$, $\sin((1-\delta)\pi/2) \le 1-\delta^2$ for all $\delta \in (0,1)$, $(1-\delta) \le e^{-\delta}$, and $\lambda \in [\Lambda, 1]$.

As shown above, the truncated series could act well as an approximation.

$$\left|-\ln(\lambda) - \sum_{k=1}^{K} \frac{1}{k} \sum_{l=0}^{\lfloor L\rfloor} b_l^{(k)} \cos^l(\lambda\pi/2)\right| \le \frac{\epsilon}{2}. \tag{5.36}$$

Step 3. Now, we use the equality $\cos(z) = \frac{e^{iz}+e^{-iz}}{2}$ to rewrite the series in Eq. (5.36).

$$\sum_{k=1}^{K} \frac{1}{k} \sum_{l=0}^{\lfloor L\rfloor} b_l^{(k)} \left[\frac{e^{i\lambda\pi/2} + e^{-i\lambda\pi/2}}{2}\right]^l = \sum_{k=1}^{K} \frac{1}{k} \sum_{l=0}^{\lfloor L\rfloor} b_l^{(k)} 2^{-l} \sum_{s=0}^{l} \binom{l}{s} e^{i(2s-l)\lambda\pi/2}. \tag{5.37}$$

Particularly, this series could be further truncated by using the property of binomial distribution. By Chernoff's inequality, we have

$$\sum_{s=\lceil l/2\rceil + M_l}^{l} 2^{-l} \binom{l}{s} \le e^{-\frac{2M_l^2}{l}}. \tag{5.38}$$

Setting $M_l = \left\lceil \sqrt{\ln\left(\frac{4\sum_{k=1}^{K} 1/k}{\epsilon}\right) \frac{l}{2}} \right\rceil$, suppose that $M_l \le \lfloor l/2\rfloor$, then we could find that

$$\sum_{s=0}^{\lfloor l/2\rfloor - M_l} 2^{-l} \binom{l}{s} = \sum_{s=\lceil l/2\rceil + M_l}^{l} 2^{-l} \binom{l}{s} \le e^{-\frac{2M_l^2}{l}} \le \frac{\epsilon}{4\sum_{k=1}^{K} 1/k}. \tag{5.39}$$

Eventually, we have obtained the desired series, shown below, up to precision ϵ.

$$\sum_{l=0}^{\lfloor L\rfloor} \sum_{s=\lfloor l/2\rfloor - M_l}^{\lceil l/2\rceil + M_l} \left(\sum_{k=1}^{K} \frac{b_l^{(k)}}{k}\right) 2^{-l} \binom{l}{s} e^{i(2s-l)\lambda\pi/2}. \tag{5.40}$$

The claimed result follows immediately from the formula of entropy.

In addition, the sum of all coefficients is bounded. This is because

$$\sum_{l=0}^{\lfloor L\rfloor}\sum_{s=\lfloor l/2\rfloor-M_l}^{\lceil l/2\rceil+M_l}\left(\sum_{k=1}^{K}\frac{b_l^{(k)}}{k}\right)2^{-l}\binom{l}{s}\leq\sum_{l=0}^{\lfloor L\rfloor}\left(\sum_{k=1}^{K}\frac{b_l^{(k)}}{k}\right) \tag{5.41}$$

$$\leq\sum_{k=1}^{K}\frac{1}{k} \tag{5.42}$$

$$\leq\ln(K+1)+1. \tag{5.43}$$

Here we have used facts that $2^{-l}\sum_{s=0}^{l}\binom{l}{s}=1$, $\sum_{l=0}^{\infty}b_l^{(k)}=\left(\frac{\arcsin(1)}{\pi/2}\right)^k=1$, and the result in Lemma 6. ■

5.1.2 Approximation of α-Rényi entropy

As for the quantum α-Rényi entropy, the estimation task can be simplified. Notice the expression $R_\alpha(\rho)$, we only need to focus on the quantity $\mathrm{tr}(\rho^\alpha)$ and derive the desired estimate via calculation after obtaining the estimate of $\mathrm{tr}(\rho^\alpha)$. Next, we construct the Fourier series approximation to $\mathrm{tr}(\rho^\alpha)$. Moreover, we display how the estimation error of $\mathrm{tr}(\rho^\alpha)$ propagates to the α-Rényi entropy.

To begin with, we write $\mathrm{tr}(\rho^\alpha)=\mathrm{tr}(\rho\cdot\rho^\beta)$, where $\beta=\alpha-1$. Recall the Taylor series of the power function x^β over the interval $x\in(0,2)$.

$$x^\beta=\sum_{k=0}^{\infty}\binom{\beta}{k}(x-1)^k, \tag{5.44}$$

where $\binom{\beta}{k}=\prod_{j=1}^{k}\frac{\beta-j+1}{j}=\frac{\beta(\beta-1)\ldots(\beta-k+1)}{k!}$ is the generalized binomial coefficient.

Clearly, truncating the infinity series of x^β would lead to the desired Taylor series. While, for different β, the truncation order K will be various. To get such an order, we need more information about the generalized binomial coefficients. Thus we show the bounds on the generalized binomial coefficient below.

Proposition 7. *For any constant $\beta\in(-1,0)\cup(0,+\infty)$, there exists a bound on the generalized binomial coefficient $\binom{\beta}{k}$.*

1. *For $\beta\in(-1,0)$ and any integer $k\geq 1$, $|\binom{\beta}{k}|\leq|\beta|$.*

2. *For $\beta\in(0,1]$ and any $k\geq 2$, $\left|\binom{\beta}{k}\right|\leq\left[1+\frac{\beta\ln\frac{(k+1)}{k^2}+\beta-1}{k}\right]^k$. Particularly, $|\binom{\beta}{k}|\leq\frac{1}{e}$, if $k\in\Omega(1)$.*

3. *For* $\beta \in (1,+\infty)$ *and* $k \geq \beta+1$, $\left|\binom{\beta}{k}\right| \leq \left[1+\frac{\beta \ln \frac{(\beta+1)^2}{k}+2}{k}\right]^k$. *Particularly,* $|\binom{\beta}{k}| \leq 1$, *if* $k \in \Omega((\beta+1)^2)$.

Moreover, for an integer K, the sum $\sum_{k=1}^{K}\left|\binom{\beta}{k}\right|$ *is bounded.*

(5.45)
$$\sum_{k=1}^{K}\left|\binom{\beta}{k}\right| \leq \begin{cases} O\left(K+e^2(\beta+1)^{2\beta}\cdot\left[\ln(\lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil+1)+1\right]-e^{\frac{2}{\beta}}(\beta+1)^2\right), & \text{if } \beta\in(1,+\infty), \\ O(K), & \text{if } \beta\in(0,1], \\ O\left(|\beta|K\right), & \text{if } \beta\in(-1,0). \end{cases}$$

Proof. Case 1: $\beta \in (1,+\infty)$.

We consider an integer k such that $k-1 \geq \beta$. We use $[\beta]$ to denote the maximal integer that is less than β.

$$\left|\binom{\beta}{k}\right| = \frac{\beta|\beta-1|\cdot|\beta-2|\cdot\ldots\cdot|\beta-k+1|}{k!} \tag{5.46}$$

$$= \frac{\beta}{k}\cdot\frac{|\beta-1|}{1}\cdot\frac{|\beta-2|}{2}\cdot\ldots\frac{|\beta-k+1|}{k-1}. \tag{5.47}$$

$$\leq \left[\frac{\frac{\beta}{k}+|\frac{\beta}{1}-1|+\ldots+|\frac{\beta}{[\beta]}-1|+\ldots+|\frac{\beta}{k-1}-1|}{k}\right]^k \tag{5.48}$$

$$= \left[\frac{\frac{\beta}{k}+(\frac{\beta}{1}-1)+\ldots+(\frac{\beta}{[\beta]}-1)+\ldots+(1-\frac{\beta}{k-1})}{k}\right]^k \tag{5.49}$$

$$= \left[\frac{\beta(1+\frac{1}{2}+\ldots+\frac{1}{[\beta]}+\frac{1}{k})-[\beta]+(k-1-[\beta])-\beta(\frac{1}{[\beta]+1}+\ldots+\frac{1}{k-1})}{k}\right]^k \tag{5.50}$$

$$= \left[\frac{2\beta(1+\frac{1}{2}+\ldots+\frac{1}{[\beta]})+\frac{\beta}{k}-[\beta]+(k-1-[\beta])-\beta(1+\frac{1}{2}+\ldots+\frac{1}{k-1})}{k}\right]^k. \tag{5.51}$$

In the first inequality, we use the inequality of arithmetic and geometric means.

Notice that sum $\sum_{l=1}^{k}\frac{1}{k}$ is a Harmonic number. By the properties of Harmonic numbers, we have

$$1+\frac{1}{2}+\ldots+\frac{1}{[\beta]} \leq \ln([\beta]+1)+1 \leq \ln(\beta+1)+1, \tag{5.52}$$

$$1+\frac{1}{2}+\ldots+\frac{1}{k-1} \geq \ln(k). \tag{5.53}$$

With these properties, we derive a new bound on the generalized binomial coefficient.

$$\left|\binom{\beta}{k}\right| \leq \left[\frac{2\beta(\ln(\beta+1)+1)+\frac{\beta}{k}-2[\beta]+k-1-\beta\ln(k)}{k}\right]^k \tag{5.54}$$

$$\leq \left[1+\frac{2\beta(\ln(\beta+1)+1)-2(\beta-1)-\beta\ln(k)}{k}\right]^k \tag{5.55}$$

$$= \left[1+\frac{\beta\ln\frac{(\beta+1)^2}{k}+2}{k}\right]^k. \tag{5.56}$$

Let $k \geq e^{\frac{2}{\beta}}(\beta+1)^2$, then we have $\beta\ln\frac{(\beta+1)^2}{k}+2 \leq 0$. Then $|\binom{\beta}{k}|$ will decrease exponentially fast as k increases. As a result, it suffices to bound the generalized binomial coefficient using $k = \lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil$, i.e., $|\binom{\beta}{k}| \leq 1$.

For an integer $K \geq \lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil$, we can derive a bound on the sum $\sum_{k=1}^K |\binom{\beta}{k}|$. Note that for $k \geq \lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil$, the generalized coefficient $\binom{\beta}{k}$ has a bound 1.

$$\sum_{k=1}^K \left|\binom{\beta}{k}\right| \leq \sum_{k=1}^{\lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil} \left[1+\frac{\beta\ln\frac{(\beta+1)^2}{k}+2}{k}\right]^k + K - \lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil \tag{5.57}$$

$$\leq \sum_{k=1}^{\lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil} \exp\left(\beta\ln\frac{(\beta+1)^2}{k}+2\right) + K - \lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil \tag{5.58}$$

$$= \sum_{k=1}^{\lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil} e^2\cdot\left[\frac{(\beta+1)^2}{k}\right]^\beta + K - \lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil. \tag{5.59}$$

In the second inequality, we use the inequality $\exp(x) \geq 1+x$ for all $x \geq 0$.

Next, we give a bound.

$$\sum_{k=1}^{\lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil} e^2\cdot\left[\frac{(\beta+1)^2}{k}\right]^\beta = e^2\cdot(\beta+1)^{2\beta}\sum_{k=1}^{\lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil} k^{-\beta} \tag{5.60}$$

$$< e^2\cdot(\beta+1)^{2\beta}\sum_{k=1}^{\lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil} \frac{1}{k} \tag{5.61}$$

$$\leq e^2\cdot(\beta+1)^{2\beta}\left[\ln(\lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil+1)+1\right]. \tag{5.62}$$

In the first inequality, we use the fact that $k^{-\beta} < k^{-1}$ as $\beta > 1$.

Consequently, we readily obtain a bound for the sum.

$$\sum_{k=1}^K \left|\binom{\beta}{k}\right| \leq e^2\cdot(\beta+1)^{2\beta}\cdot\left[\ln(\lceil e^{\frac{2}{\beta}}(\beta+1)^2\rceil+1)+1\right] + K - e^{\frac{2}{\beta}}(\beta+1)^2. \tag{5.63}$$

Case 2: $\beta \in (0,1]$.

Recall the form of the generalized binomial coefficient.

$$\left|\binom{\beta}{k}\right| = \frac{\beta(1-\beta)\ldots(k-1-\beta)}{k!} \tag{5.64}$$

$$= \frac{\beta}{k}\cdot(1-\beta)\ldots(1-\frac{\beta}{k-1}) \tag{5.65}$$

$$\leq \left[\frac{\frac{\beta}{k}+(1-\beta)+\ldots+(1-\frac{\beta}{k-1})}{k}\right]^k \tag{5.66}$$

$$= \left[\frac{\frac{\beta}{k}+(k-1)-\beta(1+\frac{1}{2}+\ldots+\frac{1}{k-1})}{k}\right]^k \tag{5.67}$$

$$\leq \left[\frac{\frac{\beta}{k}+(k-1)-\beta\ln(k)}{k}\right]^k \tag{5.68}$$

$$= \left[1+\frac{\frac{\beta}{k}-1-\beta\ln(k)}{k}\right]^k \tag{5.69}$$

$$\leq \left[1+\frac{\beta[\ln(k+1)+1-\ln(k)]-\beta\ln(k)-1}{k}\right]^k \tag{5.70}$$

$$= \left[1+\frac{\beta\ln\frac{(k+1)}{k^2}+\beta-1}{k}\right]^k. \tag{5.71}$$

The first inequality is due to the inequality of arithmetic and geometric means. The second and third are the results of applying the properties of the Harmonic series.

Notice that for any integer $k \geq 4$, the numerator $\beta\ln\frac{(k+1)}{k^2}+\beta-1 < \beta-1 < -1$. Hence, we can derive a bound by setting $k \geq 4$.

$$\left|\binom{\beta}{k}\right| \leq \left[1+\frac{\beta\ln\frac{(k+1)}{k^2}+\beta-1}{k}\right]^k \leq \frac{1}{e}. \tag{5.72}$$

Immediately, the bound on the sum is given as $O(K)$.

Case 3: $\beta \in (-1,0)$.

For any integer k, the bound on the generalized coefficient is given as follows.

$$\left|\binom{\beta}{k}\right| = \frac{(-\beta)(1-\beta)\ldots(k-1-\beta)}{k!} \tag{5.73}$$

$$\leq \frac{(-\beta)(2)\ldots(k)}{k!} \tag{5.74}$$

$$= -\beta. \tag{5.75}$$

Again, the bound on the sum is $O(|\beta|K)$. ■

With the bounds on $\binom{\beta}{k}$ in Proposition 7, we are able to develop the Taylor series approximation to $\mathrm{tr}(\rho^\alpha)$.

Proposition 8. *For any constant $\alpha \in (0,1)\cup(1,+\infty)$ and $\xi \in (0,1)$, there exists an integer K such that, for any quantum state ρ with eigenvalue lower bound Λ,*

$$\left|\mathrm{tr}(\rho^\alpha) - 1 - \sum_{k=1}^{K}\binom{\beta}{k}\mathrm{tr}\left(\rho(\rho-I)^k\right)\right| \leq \xi. \tag{5.76}$$

Particular, the choice of integer K is shown below.

$$K \in \begin{cases} \max\{\Omega(\alpha^2), \Omega(\log(\Lambda\xi)/\log(1-\Lambda))\} & \text{if } \alpha \in (2,+\infty), \\ \Omega(\log(\Lambda\xi)/\log(1-\Lambda)) & \text{if } \alpha \in (0,1)\cup(1,2]. \end{cases} \tag{5.77}$$

Proof. Let $\rho = \sum_j p_j \left|\mathbf{e}_j\right\rangle\left\langle\mathbf{e}_j\right|$ be the Schmidt decomposition of ρ, and expand ρ^β into the Taylor series.

$$\left\|\rho^\alpha - \rho - \sum_{k=1}^{K}\binom{\beta}{k}\rho(\rho - I_{supp})^k\right\| = \left\|\sum_{k=K+1}^{\infty}\binom{\beta}{k}\sum_j p_j(p_j-1)^k \left|\mathbf{e}_j\right\rangle\left\langle\mathbf{e}_j\right|\right\| \tag{5.78}$$

$$\leq \sum_{k=K+1}^{\infty}\left|\binom{\beta}{k}\right|(1-\Lambda)^k \tag{5.79}$$

$$\leq \frac{(1-\Lambda)^{K+1}}{\Lambda}. \tag{5.80}$$

By Proposition 7, we can find an integer K such that $|\binom{\beta}{k}| \leq 1$ for all $k \geq K$.

Furthermore, letting $K \geq \log(\Lambda\xi)/\log(1-\xi)$, we can verify that $(1-\Lambda)^{K+1}/\Lambda \leq \xi$. ■

Now we can give a Fourier series approximation of $R_\alpha(\rho)$.

Lemma 8. *Consider a quantum state $\rho \in \mathbb{C}^{2^n\times 2^n}$. Let $\Lambda \in (0,1)$ be a lower bound on all non-zero eigenvalues. For any $\alpha \in (0,1)\cup(1,+\infty)$, there exists an estimate $R_\alpha(\rho)_{est}$ of $R_\alpha(\rho)$ up to precision ϵ. To be specific,*

$$R_\alpha(\rho)_{est} = \frac{1}{1-\alpha}\log F_\alpha(\rho), \tag{5.81}$$

and $F_\alpha(\rho)$ satisfies $|F_\alpha(\rho) - \mathrm{tr}(\rho^\alpha)| \leq \xi$, where

$$F_\alpha(\rho) = 1 + \sum_{l=0}^{\lfloor L\rfloor}\sum_{s=D_l}^{U_l}\left(\sum_{k=1}^{K}(-1)^k b_l^{(k)}\binom{\alpha-1}{k}\right)2^{-l}\binom{l}{s}\mathrm{tr}(\rho\cdot\cos(\rho t(s,l))). \tag{5.82}$$

In particular, the relation between ϵ and ξ is given in Eq. (5.106)*, and definition of all $b_l^{(k)}$ are given in Eq.* (5.17)*. And the parameters of $F_\alpha(\rho)$ are given as follows. Coefficient $t(s,l)=\frac{(2s-l)\pi}{2}$, and $U_l=\min\{l,\lceil\frac{l}{2}\rceil+M_l\}$ and $D_l=\max\{0,\lfloor\frac{l}{2}\rfloor-M_l\}$. Moreover,*

(5.83)

$$K=\Theta\left(\frac{\log(\Lambda\xi)}{\log(1-\Lambda)}+\alpha^2\right),\quad L=\ln\left(\frac{4\sum_{k=1}^K|\binom{\alpha-1}{k}|}{\xi}\right)\frac{1}{\Lambda^2},\quad M_l=\left\lceil\sqrt{\ln\left(\frac{4\sum_{k=1}^K|\binom{\alpha-1}{k}|}{\xi}\right)\frac{l}{2}}\right\rceil.$$

And the overall weights of $F_\alpha(\rho)$ are bounded by $\sum_{k=1}^K|\binom{\alpha-1}{k}|$, and bounds on $\sum_{k=1}^K|\binom{\alpha-1}{k}|$ are given in Table 5.1.

α	Bound on $\sum_{k=1}^K\lvert\binom{\alpha-1}{k}\rvert$
$(0,1)$	$O(\lvert\alpha-1\rvert K)$
$(1,2]$	$O(K)$
$(2,+\infty)$	$O\left(K+e^2(\alpha)^{2\alpha-2}\cdot\left[\ln(\lceil e^{\frac{2}{\alpha-1}}(\alpha)^2\rceil+1)+1\right]-e^{\frac{2}{\alpha-1}}(\alpha)^2\right)$

Table 5.1: Upper bound on the overall weights.

Proof. We focus on constructing a Fourier series approximation to ρ^β, where $\beta=\alpha-1$.

By Lemma 8, we can find a Taylor series in Eq. (5.76) that approximates to ρ^β with the error $(1-\Lambda)^{K+1}/\Lambda$. Setting $K=\Theta\left(\frac{\log(\Lambda\xi)}{\log(1-\Lambda)}+\alpha^2\right)$, the approximation error is suppressed to $\xi/4$.

We show the process of constructing the Fourier series in three steps.

Step 1. For simplicity, we only consider the eigenvalue λ instead of state ρ. Notice that

$$\lambda-1=\frac{\arcsin(\sin(\frac{(\lambda-1)\pi}{2}))}{\frac{\pi}{2}}=-\frac{\arcsin(\cos(\lambda\pi/2))}{\pi/2}. \tag{5.84}$$

Take the relation in Eq. (5.84) into the series in Eq. (5.76), then we can find an approximation to $\mathrm{tr}(\rho^\alpha)$.

$$\mathrm{tr}(\rho^\alpha)=\mathrm{tr}(\rho\cdot\rho^\beta)\approx 1+\sum_{k=1}^K\binom{\beta}{k}(-1)^k\,\mathrm{tr}\left(\rho\left(\frac{\arcsin(\cos(\rho\pi/2))}{\pi/2}\right)^k\right). \tag{5.85}$$

Next, expand the function $(\arcsin(y)/\pi/2)^k$ into the Taylor series, which is given in the following formula.

$$(\arcsin(y)/\pi/2)^k=\sum_{l=0}^\infty b_l^{(k)}y^l. \tag{5.86}$$

The coefficients $b_l^{(k)}$ can be efficiently calculated.

Then, $\mathrm{tr}(\rho^\alpha)$ could be written as follows.

$$\mathrm{tr}(\rho^\alpha) \approx 1 + \sum_{k=1}^{K} \binom{\beta}{k} (-1)^k \sum_{l=0}^{\infty} b_l^{(k)} \mathrm{tr}\left(\rho \cdot \cos(\rho\pi/2)^l\right) \tag{5.87}$$

$$= 1 + \sum_{k=1}^{K} \binom{\beta}{k} (-1)^k \sum_{l=0}^{\infty} b_l^{(k)} \mathrm{tr}\left(\rho \cdot \left(\frac{e^{-i\rho\pi/2} + e^{i\rho\pi/2}}{2}\right)^l\right) \tag{5.88}$$

$$= 1 + \sum_{k=1}^{K} \binom{\beta}{k} (-1)^k \sum_{l=0}^{\infty} b_l^{(k)} \sum_{s=0}^{l} \frac{\binom{l}{s}}{2^l} \mathrm{tr}\left(\rho \cdot e^{i\rho(2s-l)\pi/2}\right). \tag{5.89}$$

Consequently, we could truncate the above series to obtain an approximation of $\mathrm{tr}(\rho^\alpha)$ at order L.

$$\mathrm{tr}(\rho^\alpha) \approx 1 + \sum_{k=1}^{K} \binom{\beta}{k} (-1)^k \sum_{l=0}^{\lfloor L \rfloor} b_l^{(k)} \sum_{s=0}^{l} \frac{\binom{l}{s}}{2^l} \mathrm{tr}\left(\rho \cdot \cos\left(\rho \frac{(2s-l)\pi}{2}\right)\right). \tag{5.90}$$

By setting $L = \ln\left(\frac{4\sum_{k=1}^{K} |\binom{\beta}{k}|}{\xi}\right) \frac{1}{\Lambda^2}$, we would obtain a series approximating to $\mathrm{tr}(\rho^\alpha)$ up to precision $\xi/2$. This is because, for any non-zero eigenvalue λ, we have

$$\left| \sum_{l=\lceil L \rceil}^{\infty} b_l^{(k)} \cos^l(\lambda\pi/2) \right| \leq \sum_{l=\lceil L \rceil}^{\infty} b_l^{(k)} \left| \cos^l(\lambda\pi/2) \right| \tag{5.91}$$

$$\leq \cos^L(\lambda\pi/2) \sum_{l=\lceil L \rceil}^{\infty} b_l^{(k)} \tag{5.92}$$

$$\leq \cos^L(\lambda\pi/2) \tag{5.93}$$

$$= \sin^L((1-\lambda)\pi/2) \tag{5.94}$$

$$\leq (1-\lambda^2)^L \tag{5.95}$$

$$\leq e^{-\lambda^2 L} \tag{5.96}$$

$$\leq e^{-\Lambda^2 L} \tag{5.97}$$

$$\leq \frac{\xi}{4\sum_{k=1}^{K} |\binom{\beta}{k}|}. \tag{5.98}$$

Here, we have used facts that $b_l^{(k)} \geq 0$, $\sum_{l=1}^{\infty} b_l^{(k)} = 1$, $\sin((1-\delta)\pi/2) \leq 1-\delta^2$ for all $\delta \in (0,1)$, $(1-\delta) \leq e^{-\delta}$, and $\lambda \in [\Lambda, 1]$.

Step 3. Now, we use the equality $\cos(z) = \frac{e^{iz}+e^{-iz}}{2}$ to rewrite the series in Eq. (5.90).

$$\sum_{k=1}^{K} (-1)^k \binom{\beta}{k} \sum_{l=0}^{\lfloor L \rfloor} b_l^{(k)} \left[\frac{e^{i\lambda\pi/2} + e^{-i\lambda\pi/2}}{2}\right]^l = \sum_{k=1}^{K} (-1)^k \binom{\beta}{k} \sum_{l=0}^{\lfloor L \rfloor} b_l^{(k)} 2^{-l} \sum_{s=0}^{l} \binom{l}{s} e^{i(2s-l)\lambda\pi/2}. \tag{5.99}$$

Particularly, this series could be further truncated by using the property of binomial distribution. By Chernoff's inequality, we have

$$\sum_{s=\lceil l/2\rceil+M_l}^{l} 2^{-l}\binom{l}{s} \le e^{-\frac{2M_l^2}{l}}. \tag{5.100}$$

Setting $M_l = \left\lceil \sqrt{\ln\left(\frac{4\sum_{k=1}^{K}|\binom{\beta}{k}|}{\epsilon}\right)\frac{l}{2}} \right\rceil$, suppose that $M_l \le \lfloor l/2\rfloor$, then we could find that

$$\sum_{s=0}^{\lfloor l/2\rfloor-M_l} 2^{-l}\binom{l}{s} = \sum_{s=\lceil l/2\rceil+M_l}^{l} 2^{-l}\binom{l}{s} \le e^{-\frac{2M_l^2}{l}} \le \frac{\xi}{4\sum_{k=1}^{K}|\binom{\beta}{k}|}. \tag{5.101}$$

Eventually, we have obtained the desired series $F_\alpha(\rho)$, shown below, up to precision ξ.

$$F_\alpha(\rho) = 1 + \sum_{l=0}^{\lfloor L\rfloor}\sum_{s=\lfloor l/2\rfloor-M_l}^{\lceil l/2\rceil+M_l}\left(\sum_{k=1}^{K}(-1)^k b_l^{(k)}\binom{\beta}{k}\right)2^{-l}\binom{l}{s}e^{i(2s-l)\lambda\pi/2}. \tag{5.102}$$

The claimed result follows immediately from the formula of entropy.

In addition, the sum of all coefficients is bounded. This is because

$$\sum_{l=0}^{\lfloor L\rfloor}\sum_{s=\lfloor l/2\rfloor-M_l}^{\lceil l/2\rceil+M_l}\sum_{k=1}^{K}\left|(-1)^k b_l^{(k)}\binom{\beta}{k}2^{-l}\binom{l}{s}\right| \le \sum_{l=0}^{\lfloor L\rfloor}\left|\sum_{k=1}^{K} b_l^{(k)}\binom{\beta}{k}\right| \le \sum_{k=1}^{K}\left|\binom{\beta}{k}\right|. \tag{5.103}$$

Here we have used facts that $2^{-l}\sum_{s=0}^{l}\binom{l}{s} = 1$, $\sum_{l=0}^{\infty} b_l^{(k)} = \left(\frac{\arcsin(1)}{\pi/2}\right)^k = 1$.

By Proposition 7, we can find the upper bound on the sum of generalized coefficients as claimed.

■

5.1.3 Approximation error analysis

Here, we discuss the relation between ξ and ϵ. Let $\widehat{\mathrm{tr}(\rho^\alpha)}$ be an estimate of $\mathrm{tr}(\rho^\alpha)$ up to error ξ, i.e., $|\widehat{\mathrm{tr}(\rho^\alpha)} - \mathrm{tr}(\rho^\alpha)| \le \xi$. Then the difference between the corresponding logarithms is given below.

$$\left|\frac{1}{1-\alpha}\log\widehat{\mathrm{tr}(\rho^\alpha)} - \frac{1}{1-\alpha}\log\mathrm{tr}(\rho^\alpha)\right| = \frac{1}{|1-\alpha|}\left|\log\frac{\widehat{\mathrm{tr}(\rho^\alpha)}}{\mathrm{tr}(\rho^\alpha)}\right| = \frac{1}{|1-\alpha|}\left|\log\left(1+\frac{\widehat{\mathrm{tr}(\rho^\alpha)}-\mathrm{tr}(\rho^\alpha)}{\mathrm{tr}(\rho^\alpha)}\right)\right|. \tag{5.104}$$

Notice that $|\log(1+x)| \le 2|x|$ for any $x \in [-0.5, 1]$. Then we can assume that $\frac{\widehat{\mathrm{tr}(\rho^\alpha)}-\mathrm{tr}(\rho^\alpha)}{\mathrm{tr}(\rho^\alpha)}$ falls in the interval $[-0.5, 1]$ and hence we have

$$\left|\frac{1}{1-\alpha}\log\mathrm{tr}(\rho^\alpha) - \frac{1}{1-\alpha}\log\widehat{\mathrm{tr}(\rho^\alpha)}\right| \le \frac{2}{|1-\alpha|}\left|\frac{\widehat{\mathrm{tr}(\rho^\alpha)}-\mathrm{tr}(\rho^\alpha)}{\mathrm{tr}(\rho^\alpha)}\right| \le \frac{2\xi}{|1-\alpha||\mathrm{tr}(\rho^\alpha)|}. \tag{5.105}$$

Moreover, since $\mathrm{tr}(\rho^\alpha) \geq [\mathrm{tr}(\rho^2)]^{\alpha-1}$ for all $\alpha \in (0,1) \cup (2,+\infty)$, and $\mathrm{tr}(\rho^\alpha) \geq \mathrm{tr}(\rho^2)$ for all $\alpha \in (1,2]$, we can determine ξ upon receiving ϵ. Explicitly, ξ is given by

$$\xi = \begin{cases} \frac{|1-\alpha|[\mathrm{tr}(\rho^2)]^{\alpha-1}}{2}\epsilon, & \forall \alpha \in (0,1) \cup (2,+\infty), \\ \frac{|1-\alpha|\mathrm{tr}(\rho^2)}{2}\epsilon, & \forall \alpha \in (1,2]. \end{cases} \tag{5.106}$$

Remark 8 The Swap test [47] can evaluate the term $\mathrm{tr}(\rho^2)$ efficiently. As a result, the 2-Rényi entropy can be obtained via Swap test as well. If let r_ρ be the rank of state ρ, we can substitute $\mathrm{tr}(\rho^2)$ with $1/r_\rho$ due to the fact that $\mathrm{tr}(\rho^2) \geq 1/r_\rho$. In these cases, the estimation accuracy ξ could be polynomially small if ρ is low-rank or $\mathrm{tr}(\rho^2)$ is polynomially small.

Now, we have provided estimates for the von Neumann and α-Rényi entropy in Lemma 7 & 8. Especially, these estimates can be obtained by evaluating the Fourier series in Eq. (5.15) & (5.82) on quantum computers. To achieve this purpose, we proceed to devise quantum circuits to estimate quantity $\mathrm{tr}(\rho\cos(\rho \cdot t(s,l)))$.

5.2 Quantum circuit construction

In this section, we first show a scheme to estimate the term $\mathrm{tr}(\rho\cos(\rho t))$. We also demonstrate the validity and estimate the cost of primitive single/two-qubit gates. Then we discuss compressing the circuit width. In the end, we discuss a crucial subroutine that can simulate the exponentiation of the Swap operator.

5.2.1 Circuit scheme

For simplicity, we first consider estimating $\mathrm{tr}(\rho\cos(\rho t))$ with small t. Based on the circuit of iterative quantum phase estimation [40, 62, 87], the circuit for this purpose is depicted in Figure 5.1. Please note that we denote the top first qubit of the circuit as the measure register. The first state ρ is prepared in the main register, and other copies are prepared on the ancillary registers.

In Figure 5.1, qubit $|0\rangle\langle 0|$ and two copies of state ρ are input into the circuit. Then two Hadamarad gates are applied to the measure register, sandwiching a controlled operation operation c-$e^{-i\mathcal{S}t}$, i.e., the exponentiation of the Swap operator, where $\mathcal{S}$ denotes the Swap operator. In the end of the circuit, the measurement occurs on the measure register along the eigenbasis of the Pauli matrix Z. Particularly, the measurement outcomes lead to the value of $\mathrm{tr}(\rho\cos(\rho t))$. More precisely, $\mathrm{tr}(\rho\cos(\rho t)) \approx \Pr[0] - \Pr[1]$, where $\Pr[0/1]$ denotes

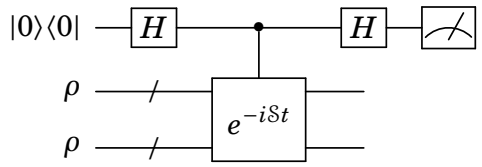


Figure 5.1: For a short time t, we first prepare a ground state $|0\rangle\langle 0|$ in the measure register, and prepare states ρ in the main register the ancillary register, respectively. Subsequently, perform the controlled unitary operator $e^{-i\mathcal{S}t}$ on state $\rho\otimes\rho$. At the end of the circuit, we measure along the eigenbasis of Pauli Z, which would immediately lead to an estimate for $\mathrm{tr}(\rho\cos(\rho t))$ up to precision $O(t^2)$.

the probability of observing outcome 0/1, respectively. And the estimation accuracy is shown in the result below.

Proposition 9. *Let V denote the unitary corresponding to the circuit in Figure 5.1. For any quantum state ρ and a small parameter $t\in(-1,1)$, the average of measurement results is close to* $\mathrm{tr}(\rho\cos(\rho t))$. *Explicitly,*

$$\left|\mathrm{tr}(\rho\cos(\rho t))-\mathrm{tr}\left(\left(V(|0\rangle\langle 0|\otimes\rho^{\otimes 2})V^\dagger\right)Z_0\right)\right|\le 2t^2. \tag{5.107}$$

Here, Z_0 equals to $Z\otimes I\otimes I$, which indicates measuring the measure register along Pauli Z's eigenbasis.

Proof. To demonstrate the validity, we focus on the state of the measure register and the main register. Notice that, after the controlled operation, the initial state evolves into the following form.

$$\frac{1}{2}\left[|0\rangle\langle 0|\otimes\rho^{\otimes 2}+|1\rangle\langle 1|\otimes e^{-i\mathcal{S}t}\rho^{\otimes 2}e^{i\mathcal{S}t}\right]+\frac{1}{2}\left[|0\rangle\langle 1|\otimes\rho^{\otimes 2}e^{i\mathcal{S}t}+|1\rangle\langle 0|\otimes e^{-i\mathcal{S}t}\rho^{\otimes 2}\right]. \tag{5.108}$$

Next, trace out the appended copy of ρ. For instance, a part of Eq. (5.108) is traced as follows.

(5.109)
$$|1\rangle\langle 0|\otimes\mathrm{tr}_{anc}\left(e^{-i\mathcal{S}t}\rho^{\otimes 2}\right)=|1\rangle\langle 0|\otimes\mathrm{tr}_{anc}\left((\cos(t)I-i\sin(t)\mathcal{S})\,\rho^{\otimes 2}\right)=|1\rangle\langle 0|\otimes(\cos(t)I-i\sin(t)\rho)\rho.$$

Notation tr_{anc} means tracing out the appended copy of ρ. Here, we have used facts that $e^{-i\mathcal{S}t}=\cos(t)I-i\sin(t)\mathcal{S}$ and $\mathrm{tr}_{anc}(\mathcal{S}\rho\otimes\sigma)=\rho\sigma$. A similar result could be derived for $|0\rangle\langle 1|\otimes\rho^{\otimes 2}e^{i\mathcal{S}t}$.

We bound the difference between $(\cos(t)I - i\sin(t)\rho)\rho$ and $e^{-i\rho t}\rho$.

$$
\begin{aligned}
(5.110) \qquad difference &= \left\| \left[e^{-i\rho t} - (\cos(t)I - i\sin(t)\rho) \right] \rho \right\|_{tr} \\
&= \left\| (1-\cos(t))\rho - i(t-\sin(t))\rho^2 + \sum_{k\geq 2} \frac{(-i\rho t)^k}{k!}\rho \right\|_{tr} \\
&\leq 2\sin^2(t/2) + |t - \sin(t)| + \frac{\sqrt{2}t^2}{2} \\
&\leq 2t^2,
\end{aligned}
$$

where we have used inequalities $|x - \sin(x)| \leq x^2/2$ and $|\sin(x)| \leq x$, and the inequality $\|\sum_{k\geq 2}(-i\rho t)^k/k!\|_{tr} \leq \sqrt{2}t^2/2$ (The proof can be found in Appendix A.1). Again, a similar result can be found for $\rho^{\otimes 2}e^{i\mathcal{S}t}$ and $\rho e^{i\rho t}$.

Note that the measurement outcome is

$$
(5.111) \qquad \Pr[0] - \Pr[1] = \frac{\mathrm{tr}\left(e^{-i\mathcal{S}t}\rho^{\otimes 2}\right) + \mathrm{tr}\left(\rho^{\otimes 2}e^{i\mathcal{S}t}\right)}{2} = \frac{\mathrm{tr}\left(\mathrm{tr}_{anc}(e^{-i\mathcal{S}t}\rho^{\otimes 2})\right) + \mathrm{tr}\left(\mathrm{tr}_{anc}(\rho^{\otimes 2}e^{i\mathcal{S}t})\right)}{2}.
$$

Recall we have shown that $\left\|\mathrm{tr}_{anc}(e^{-i\mathcal{S}t}\rho^{\otimes 2}) - e^{-i\rho t}\rho\right\|_{tr} \leq 2t^2$ and $\left\|\mathrm{tr}_{anc}(\rho^{\otimes 2}e^{i\mathcal{S}t}) - \rho e^{i\rho t}\right\|_{tr} \leq 2t^2$. Besides, $\mathrm{tr}(\rho\cos(\rho t)) = [\mathrm{tr}(e^{-i\rho t}\rho) + \mathrm{tr}(\rho e^{i\rho t})]/2$. Immediately, we derive the result

$$
(5.112) \qquad \left|\Pr[0] - \Pr[1] - \mathrm{tr}(\rho\cos(\rho t))\right| \leq 2t^2.
$$

■

Now we have shown that the circuit in Figure 5.1 can be used to estimate the term $\mathrm{tr}(\rho\cos(\rho t))$, especially when t is small. Regarding a large t, we use a circuit similar to

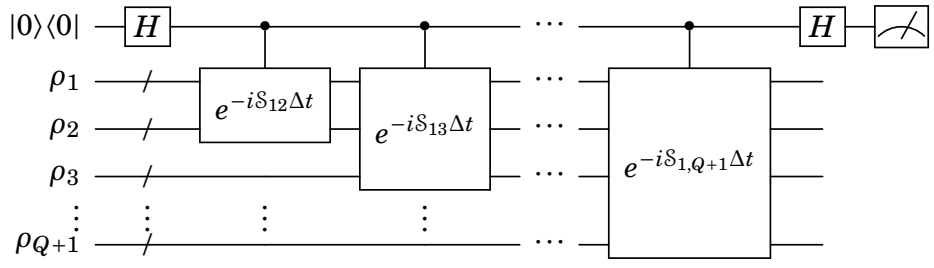


Figure 5.2: For general time t, the circuit could be inductively constructed. The operator $e^{-i\mathcal{S}\Delta t}$ is sequentially applied on the main register and different ancillary registers, conditional on the measure register. Here, we append Q ancillary states and use Q times of $e^{-i\mathcal{S}\Delta t}$. For clear, we label states on different register by $1, 2, 3, Q+1$, and the script of the swap operator indicates the registers that swap operator acts on.

that in Figure 5.1. Particularly, we divide the parameter t into several small pieces Δt and run c-$e^{-i\mathcal{S}\Delta t}$ sequentially. The corresponding circuit is depicted in Figure 5.2, where we run Q controlled operations, and the parameter Δt is small.

Based on the result in Proposition 9, we can readily derive the result for estimating $\mathrm{tr}(\rho\cos(\rho t))$ with a large $t \in \mathbb{R}$.

Proposition 10. *For any quantum state $\rho \in \mathbb{C}^{2^n \times 2^n}$ and $t \in \mathbb{R}$, there is a quantum circuit that can estimate the quantity $\mathrm{tr}(\rho\cos(\rho t))$ with precision ϵ. The number of needed copies of state is $O(t^2/\epsilon)$.*

Proof. Suppose we divide the parameter t into Q pieces and the parameter in the Swap operator becomes $\Delta t = t/Q$. In the circuit, we need Q ancillary register and the state in each ancillary register is ρ. For each interval of length Δt, the resulting error in the current state of the measure register and main register is at most $2\Delta t^2$ by Proposition 9. Easily, we can deduce the accumulating error, i.e., $Q \times 2\Delta t^2 \leq 2t^2/Q$. To suppress the overall accumulating errors, we set $Q = \lceil \frac{2t^2}{\epsilon} \rceil$. Consequently, the final error is at most ϵ.

Moreover, the number of used copies is $(Q+1) = O(t^2/\epsilon)$. ■

Proposition 10 has shown that we can use the circuit in Figure 5.2 to estimate the Fourier series. While there are some remaining issues. One is that the large circuit width may be a huge burden in practice. Another is to simulate the exponentiation of the Swap operator. The solutions to overcome these issues are discussed in the following sections.

5.2.2 Circuit width circumvent

In Figure 5.2, there are $Q+1$ copies of ρ prepared at the beginning, while the interaction only occurs between two copies at a time. For instance, copy ρ_1 only interacts with the ancillary register being state ρ_2. Once the controlled operation c-$e^{-i\mathcal{S}_{12}\Delta t}$ is completed, the occupied ancillary register is relieved. At that time, copy ρ_3 is employed for the next interaction. Hence, interactions occur alternatively between the measure register and main register and different ancillary registers.

Notice that the relieved ancillary register will no longer affect the state of the rest registers. Thus, we can measure the relieved register and reuse it for preparing a new state by the qubit reset technique. Although measuring the ancillary register will change the state in other registers, on average, their state is the reduced version of the whole system, which allows us to focus on the reduced state. Using qubit reset means that we can measure subsets of the qubits and reinitialize them [45]. In the past

decade, many experimental methods have been developed to actively reset the qubits on superconducting qubits [45, 54, 59, 101, 121, 125]. Recently, the qubit reset also applies to design quantum algorithms [52, 73, 96, 120, 159] with reduced circuit width. Particularly, [159] uses qubit reset to devise quantum circuits for estimating $\mathrm{tr}(\rho^k)$ with $k \in \mathbb{N}$, which can contribute to the problem of quantum Rényi entropy [123] estimation.

Here, we use the qubit reset to compression the width of the circuit in Figure 5.2. In this case, we only need one ancillary register. Specifically, we prepare the state $\rho_1 \otimes \rho_2$ on the main register and ancillary register in the beginning. Then the interaction occurs between them. Once the interaction ends, the ancillary register is measured. Subsequently, the ancillary register is readily reset to state ρ_3. And then, the interaction occurs again. The same procedure of measuring and the reset repeats Q times in all. The corresponding circuit using qubit reset can be found in Figure 5.3.

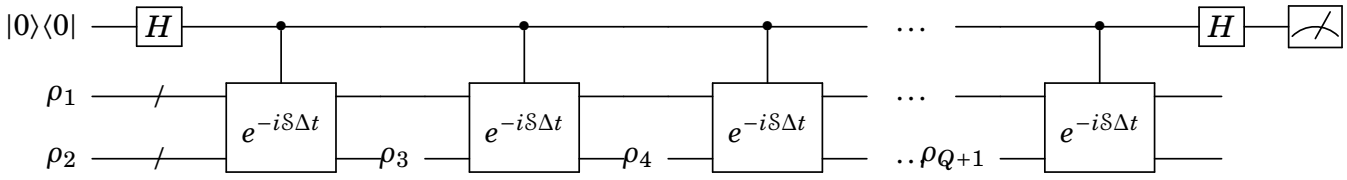


Figure 5.3: A quantum circuit for estimating $\mathrm{tr}(\rho\cos(\rho t))$ using qubit reset. The break and a state ρ in the wire means implementing qubit reset.

5.2.3 A subroutine

In this section, we devise a circuit to simulate the exponentiation of the Swap operator by the technique for simulating a linear combination of unitaries in [16]. Note that the unitary $e^{-i\mathbb{S}\Delta t}$ can be written as a linear combination of unitaries, i.e., $e^{-i\mathbb{S}\Delta t} = \cos(\Delta t)I_{2n} - i\sin(\Delta t)\mathbb{S}$. The index of identity means the number of qubits that the identity acts on. To break down the exponentiation of the Swap operator, we need to build up the module W first, which is shown in Figure 5.4. The first two qubits in state $|00\rangle$ are newly added ancillary qubits.

First, the R_1 gate is a rotation gate, the oc-R_2 gate means that act the rotation gate R_2 on the target qubit when the control qubit is in the state $|0\rangle$. The two single qubit rotations R_1 and R_2 are defined as follows:

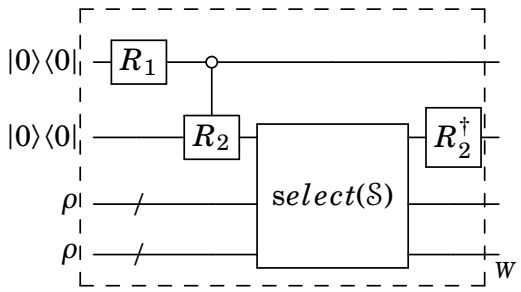


Figure 5.4: Quantum circuit for implementing the module W.

$$R_1|0\rangle = \frac{\alpha}{2}|0\rangle + \sqrt{1-\frac{\alpha^2}{4}}\,|1\rangle, \tag{5.113}$$

$$R_2|0\rangle = \sqrt{\frac{\cos(\Delta t)}{\alpha}}\,|0\rangle + \sqrt{\frac{|\sin(\Delta t)|}{\alpha}}\,|1\rangle, \tag{5.114}$$

where $\alpha = \cos(\Delta t) + |\sin(\Delta t)| < 2$. Here, we assume time Δt is small enough so that $\cos(\Delta t) > 0$.

Second, the select($\mathbb{S}$) gate implements the operation $(-i \cdot sgn)\mathbb{S}$ on $\rho \otimes \rho$ conditionally, which is defined as follows:

$$select(\mathbb{S}) = |0\rangle\langle 0| \otimes I_{2n} + |1\rangle\langle 1| \otimes (-i \cdot sgn)\mathbb{S}, \tag{5.115}$$

where sgn denotes the sign of Δt. The detailed structure can be found in Appendix A.5.

Third, we define a circuit module W as shown in Figure 5.4. At this stage, the module can be written as:

$$W = R_2^\dagger select(\mathbb{S})(oc{-}R_2)R_1. \tag{5.116}$$

Let $P = |00\rangle\langle 00|$ be the operator that projects onto the subspace spanned by $|00\rangle$, where $|00\rangle$ are the ancillary qubits in Figure 5.4. Then, define a unitary operator

$$A = -W(I_2 - 2P)W^\dagger(I_2 - 2P)W. \tag{5.117}$$

Here, the notation $I_2 - 2P$ denotes the operator that reflects along the vectors that are orthogonal to the ancillary qubits $|00\rangle$. As a result, the unitary A can simulate $e^{-i\mathbb{S}\Delta t}$.

Proposition 11. *For arbitrary parameter $\Delta t \in (-1, 1)$, define two rotations R_1 and R_2 as in Eqs.* (5.113)-(5.114)*. Define a circuit module W as in Figure 5.4 and a unitary*

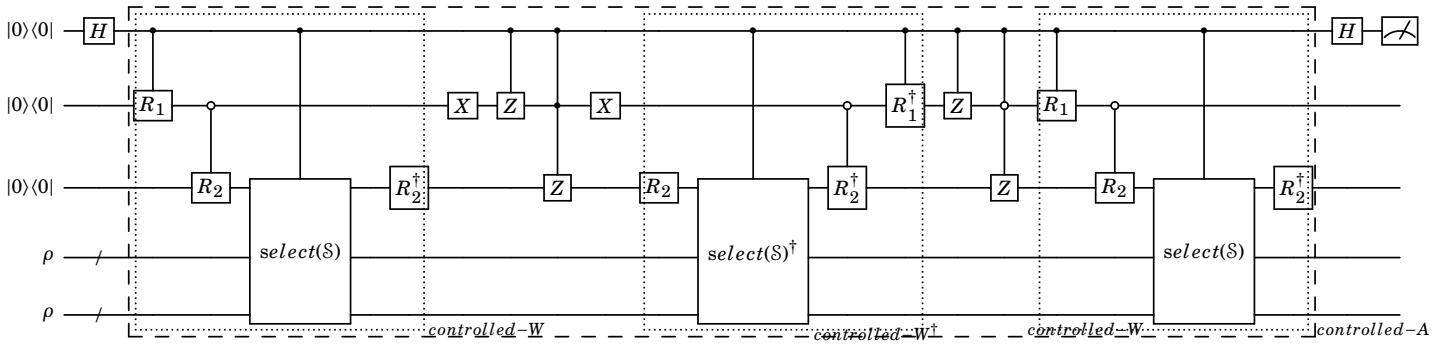


Figure 5.5: This figure depicts the resulting circuit by substituting c-$e^{-i\mathcal{S}\Delta t}$ with the circuit of controlled-A (dashed box) in Figure 5.1. The dotted circuit is the controlled-W circuit, in which the c-R_1 and oc-R_2 are the 1-controlled R_1 gate (apply R_1 on the target qubit if the control qubit in state $|1\rangle$) and 0-controlled R_2 gate (apply R_2 on the target qubit if the control qubit in state $|0\rangle$), respectively. The definitions of R_1 & R_2 can be found in Eqs. (5.113)-(5.114). The circuits between dotted boxes are known as reflectors. Denote that all elements in the circuit can be broken down into single/two-qubits gates, please refer to Appendix A.5 for details.

$A = -W(I_2 - 2P)W^\dagger(I_2 - 2P)W$, where $P = |00\rangle\langle 00|$, and I_2 denotes the identity acting on $|00\rangle$. Then the unitary $e^{-i\mathcal{S}\Delta t}$ can be simulated in the sense that

$$P \otimes e^{-i\mathcal{S}\Delta t} = PAP. \tag{5.118}$$

Proof. First, we can easily show that

$$PAP = 3PWP - 4PWPW^\dagger PWP. \tag{5.119}$$

Particularly, an important property of W is

$$\langle 00|W|00\rangle = \frac{1}{2}(\cos(\Delta t)I_{2n} - i\sin(\Delta t)\mathcal{S}) = \frac{1}{2}e^{-i\mathcal{S}\Delta t}. \tag{5.120}$$

Then,

$$PWP = \frac{1}{2}P \otimes e^{-iS\Delta t}, \tag{5.121}$$

$$PWPW^\dagger PWP = \frac{1}{4}PWP. \tag{5.122}$$

Last, the result immediately follows.

■

Using the circuit of A to substitute $e^{-i\mathcal{S}\Delta t}$ would lead to the desired quantum circuits for estimating the Fourier series. We provide one example in Figure 5.5, where the circuit of A is used once. We also estimate the number of needed primitive gates and qubits in the following.

Proposition 12. *For any quantum state $\rho \in \mathbb{C}^{2^n \times 2^n}$, and time $t \in \mathbb{R}$, there is a quantum circuit to estimate the quantity* $\mathrm{tr}(\rho\cos(\rho t))$ *up to precision ϵ. The total amount of single/two-qubit gates is $O(nt^2/\epsilon)$.*

Proof. The validity follows immediately from Proposition 10 and Proposition 11 and the gate decomposition of controlled $e^{-i\mathcal{S}\Delta t}$ in Appendix A.5. ■

By Proposition 12, we can estimate $\mathrm{tr}(\rho\cos(\rho t))$ using $O(t^2/\epsilon)$ primitive gates. In consequence, we can estimate the Fourier series by primitive gates as well.

5.3 Quantum entropy estimation

In this section, we present the quantum algorithms for estimating the von Neumann and α-Rényi entropy. The key idea is to run the devised quantum circuits to evaluate the Fourier series in Lemma 7 (for von Neumann entropy) and Lemma 8 (for Rényi entropy). To be more specific, we construct an unbiased estimator by evaluating the Fourier series via the circuits and classical post-processing the measurement outcomes. In particular, in post-processing, we use the importance sampling technique. Moreover, we analyze the correctness and cost of our algorithms.

Quantum algorithm for von Neumann entropy estimation The workflow for estimating $S(\rho)$ is depicted in Algorithm 6. First, we set a constant Λ as a lower bound on all non-zero eigenvalues of the input state ρ. Then, upon receiving the required precision ϵ and failure probability δ, we determine the Fourier series according to Lemma 7. After that, we construct an unbiased estimator by the importance sampling technique. Specifically, we randomly select each term of the Fourier series, where the probability is proportional to the corresponding weight. The average of the selected terms in expectation is proportional to the target quantity. And the proportional factor is easy to calculate. Finally, we evaluate all selected terms via quantum circuits and post-processing the measurement outcomes to reveal the desired estimates.

Algorithm 6 Quantum algorithm for von Neumann entropy estimation

Input: Constants $\epsilon, \delta, \Lambda \in (0,1)$, and copies of state $\rho \in \mathbb{C}^{2^n \times 2^n}$.
Output: Estimate of the von Neumann entropy $S(\rho)$.

1: Compute coefficients L, K, M_l, and $b_l^{(k)}$ as given in Lemma 7.
2: Compute $\|\mathbf{f}\|_{\ell_1}$, as given in Eq. (5.124).
3: Set estimation error $\varepsilon = \epsilon / \|\mathbf{f}\|_{\ell_1}$.
4: Set integer $N = \sum_{l=0}^{\lfloor L \rfloor} (2M_l + 1)$.
5: Define a distribution as in Eq. (5.125).
6: Set integer $B = \#Samples$ as in Eq. (5.127).
7: Sample B pairs of (s_1, l_1), ..., (s_B, l_B).
8: Set $j = 1$ and $sum = 0$.
9: **while** $j \leq B$ **do**
10: Set $Q = \lceil (2s_j - l_j)^2 \pi^2 / 4\varepsilon \rceil$.
11: Prepare $Q + 1$ copies of ρ.
12: Estimate $\mathrm{tr}(\rho \cos(\rho(2s_j - lj))\pi/2)$ with precision ε and probability $1 - \delta/2N$ via quantum circuits, given in Sec. 5.2.
13: Store the obtained estimate est_j.
14: Update $sum \leftarrow sum + est_j$ and $j \leftarrow j + 1$.
15: **end while**
16: **return** $\|\mathbf{f}\|_{\ell_1} \times sum/B$.

Theorem 2. *Consider a quantum state $\rho \in \mathbb{C}^{2^n \times 2^n}$. Let Λ be the lower bound on all non-zero eigenvalues of ρ. Suppose we have free access to copies of ρ, then Algorithm 6 outputs an estimate of the von Neumann entropy $S(\rho)$ up to precision ϵ, succeeding with probability at least $1 - \delta$. In addition, the total amount of the needed copies of ρ and single/two-qubit gates are, in the worst case, $\widetilde{O}\left(1/\epsilon^5 \Lambda^2\right)$ and $\widetilde{O}(n/\epsilon^3 \Lambda^2)$, respectively.*[1]

Proof. Correctness analysis.

We rephrase the Fourier series in Lemma 7 as follows.

$$F(\rho) = \sum_{l=0}^{\lfloor L \rfloor} \sum_{s=D_l}^{U_l} f(s,l) \mathrm{tr}\left(\rho \cos\left(\rho \frac{(2s-l)\pi}{2}\right)\right). \tag{5.123}$$

Coefficients $f(s,l)$ are given by

$$f(s,l) = \left(\sum_{k=1}^{K} \frac{b_l^{(k)}}{k}\right) \frac{\binom{l}{s}}{2^l}, \quad \forall s, l. \tag{5.124}$$

Let $\mathbf{f}$ be a vector that consists of $f(s,l)$. Then the ℓ_1-norm of $\mathbf{f}$ is bounded by $O(\log(K))$ i.e., $\|\mathbf{f}\|_{\ell_1} \in O(\log(K))$ (cf. Lemma 7).

[1] The notation $\widetilde{O}$ hides the logarithmic factors.

Next, define an importance sampling as follows:

$$\mathbf{R} = \mathrm{tr}\left(\rho \cos\left(\rho \frac{(2s-l)\pi}{2}\right)\right) \quad \text{with prob. } \frac{f(s,l)}{\|\mathbf{f}\|_{\ell_1}}. \tag{5.125}$$

The random variable $\mathbf{R}$ indicates that each Fourier term associated with (s,l) is sampled with probability proportional to its weight $f(s,l)$. Then, the Fourier series could be rewritten as an expectation of $\mathbf{R}$.

$$F(\rho) = \|\mathbf{f}\|_{\ell_1} \cdot \mathbf{E}[\mathbf{R}]. \tag{5.126}$$

Hence, we derive an unbiased estimator $F(\rho)$ for $S(\rho)$.

Cost analysis.

Note that the sample mean could estimate the expectation, and the estimation accuracy replies on the number of samples. Specifically, by Chebyshev's inequality, $O(\mathbf{V}ar/\epsilon^2)$ samples are sufficient to derive an estimate of the expectation with precision ϵ and high probability, where $\mathbf{V}ar$ denotes the variance, and ϵ is the precision. Meanwhile, the probability could be boosted to $1-\delta$ at the cost of an additional multiplicative factor $O(\log(1/\delta))$ according to Chernoff bounds. Alternatively, by Hoeffding's inequality, we only need $O(\log(1/\delta)/\epsilon^2)$ samples to derive an estimate with precision ϵ and probability larger than $1-\delta$.

Regarding random variable $\mathbf{R}$, the variance is less than 1. To estimate $F(\rho)$ with precision ϵ, we set the estimation precision for $\mathbf{E}[\mathbf{R}]$ as $\varepsilon = \epsilon/\|\mathbf{f}\|_{\ell_1}$ and set failure probability as $\delta/2$. As a result, the number of needed samples is

$$\#\text{Samples} = O\left(\frac{1}{\varepsilon^2}\log\left(\frac{2}{\delta}\right)\right) = O\left(\frac{\|\mathbf{f}\|_{\ell_1}^2}{\epsilon^2}\log\left(\frac{2}{\delta}\right)\right). \tag{5.127}$$

It means that there are at most $\#\mathrm{S}amples$ terms needing to estimate via quantum circuits.

On the other hand, notice that the Fourier series $F(\rho)$ consists of $N = \sum_{l=0}^{L}(2M_l+1)$ terms in all. Hence, it suffices to estimate each term with probability $1-\delta/2N$. In this way, the overall failure probability is at most δ by union bound.

When estimating the Fourier series, we need to measure the resultant state after evolving the input state ρ by the circuits (e.g., please refer to Figures 5.2 & 5.3 & 5.5). Note that the measurement outcome is evaluated nondeterministically, which implies that the estimation could fail. To suppress the failure probability to δ, we suffice to evaluate each term with a failure probability at most $\delta/2N$. In consequence, for each

term, the number of needed measurements is

$$\#\mathrm{M}easurements = O\left(\frac{\log(2N/\delta)}{\epsilon^2}\right). \tag{5.128}$$

Immediately, the total number of measurements for estimating the Fourier series is at most

$$C_m = \#\mathrm{S}ample \times \#\mathrm{M}easurements = O\left(\frac{\|\mathbf{f}\|_{\ell_1}^4}{\epsilon^4}\log\left(\frac{2}{\delta}\right)\log\left(\frac{N}{\delta}\right)\right) = \widetilde{O}\left(\frac{1}{\epsilon^4}\right). \tag{5.129}$$

Now, we consider the cost on the number of state ρ. Recall that, when estimating $\mathrm{tr}(\rho\cos(\rho t))$, performing once measurement costs $\widetilde{O}(t^2/\epsilon)$ copies of ρ by Proposition 10. For the Fourier series $F(\rho)$, the largest time is $O(M_L) = O\left(\ln\left(\frac{\ln(K)}{\epsilon}\right)\frac{1}{\Lambda}\right)$. At the same time, we have to measure C_m times in the entropy estimation. Then the number of overall copies is at most

$$C_\rho = C_m \times \widetilde{O}(M_L^2/\varepsilon) = C_m \times \widetilde{O}\left(\ln^2\left(\frac{\ln(K)}{\epsilon}\right)\frac{1}{\varepsilon\Lambda^2}\right) = \widetilde{O}\left(\frac{1}{\epsilon^5\Lambda^2}\right). \tag{5.130}$$

At last, consider the costs on the number of quantum gates. By Proposition 12, for $\mathrm{tr}(\rho\cos(\rho t))$, it costs $O(nt^2/\epsilon)$ primitive single/two-qubit gates to construct the circuit. Meanwhile, we consider the largest evolution time $O(M_L)$. Then the total amount of gate for the entropy estimation, in the worst case, is

$$C_g = \#\mathrm{S}ample \times \widetilde{O}(nM_L^2/\varepsilon) = \widetilde{O}\left(\frac{n}{\epsilon^3\Lambda^2}\right). \tag{5.131}$$

■

Quantum algorithm for quantum Rényi entropy estimation The quantum algorithm for the Rényi entropy is similar to that of the von Neumann entropy estimation. The main difference is that we evaluate the Fourier series approximation of $\mathrm{tr}(\rho^\alpha)$. Now, we show the workflow in Algorithm 7, which derives an unbiased estimator for $\mathrm{tr}(\rho^\alpha)$ and an estimate for $R_\alpha(\rho)$. For clarity, we use ϵ and ξ to denote the estimation precision of $R_\alpha(\rho)$ and $\mathrm{tr}(\rho^\alpha)$, respectively.

In the workflow, first, upon receiving the inputs, we find a Fourier series that can approximate the trace of the state's power function $\mathrm{tr}(\rho^\alpha)$ by Lemma 8. Then we proceed to evaluate the Fourier series by the quantum circuits and the classical post-processing. Note that we also employ the importance sampling to construct an unbiased estimator. Afterwards, we could readily derive the estimate of $R_\alpha(\rho)$.

Algorithm 7 Quantum algorithm for α-Rényi entropy estimation

Input: Constants $\epsilon, \delta, \Lambda \in (0,1)$, and copies of state $\rho \in \mathbb{C}^{2^n \times 2^n}$, and $\alpha \in (0,1) \cup (1,\infty)$.
Output: Estimate of the α-Rényi entropy $R_\alpha(\rho)$.

1: Set estimation precision ξ for estimating $\mathrm{tr}(\rho^\alpha)$, as given in Eq. (5.106).
2: Compute coefficients L, K, M_l, and $b_l^{(k)}$ as given in Lemma 8.
3: Compute coefficients $f(s,l) = \left(\sum_{k=1}^{K} b_l^{(k)} (-1)^k \binom{\beta}{k}\right) \frac{\binom{l}{s}}{2^l}, \quad \forall s,l.$
4: Compute the ℓ_1 norm of $\mathbf{f}$, which is the vector consisting of all $f(s,l)$.
5: Set a parameter $\varepsilon = \xi / \|\mathbf{f}\|_{\ell_1}$.
6: Set integer $N = \sum_{l=0}^{\lfloor L \rfloor} (2M_l + 1)$.
7: Define a distribution: $\mathbf{R} = \mathrm{tr}\left(\rho \cos\left(\rho \frac{(2s-l)\pi}{2}\right)\right)$ with prob. $\frac{|f(s,l)|}{\|\mathbf{f}\|_{\ell_1}}$.
8: Set integer $B = \#Samples = O\left(\frac{\|\mathbf{f}\|_{\ell_1}^2}{\xi^2} \log\left(\frac{2}{\delta}\right)\right)$.
9: Sample B pairs of (s_1, l_1), ..., (s_B, l_B) via the distribution.
10: Set $j = 1$ and $sum = 0$.
11: **while** $j \le B$ **do**
12: Set $Q = \lceil (2s_j - l_j)^2 \pi^2 / 4\varepsilon \rceil$.
13: Prepare $Q+1$ copies of ρ.
14: Estimate $\mathrm{tr}(\rho \cos(\rho(2s_j - lj))\pi/2)$ with precision ε and probability $1 - \delta/2N$ via quantum circuits, given in Sec. 5.2.
15: Store the obtained estimate est_j.
16: Update $sum \leftarrow sum + est_j$ and $j \leftarrow j+1$.
17: **end while**
18: **return** $\log(1 + \|\mathbf{f}\|_{\ell_1} \times sum/B)/(1-\alpha)$.

Theorem 3. *Consider a quantum state $\rho \in \mathbb{C}^{2^n \times 2^n}$. Let Λ be the lower bound on all non-zero eigenvalues of ρ. Suppose we have access to copies of ρ, then Algorithm 7 outputs an estimate of α-Rényi entropy $R_\alpha(\rho)$ up to precision ϵ, succeeding with probability at least $1-\delta$. Furthermore, the total amount of the needed copies of ρ and single/two-qubit gates, in the worst case, are shown in the Table 5.2.*

α	Copy cost	Gate cost
$(0,1) \cup (2,+\infty)$	$\widetilde{O}\left(\frac{\left(\sum_{k=1}^{K} \lvert\binom{\alpha-1}{k}\rvert\right)^5}{\lvert 1-\alpha\rvert^5 [\mathrm{tr}(\rho^2)]^{5(\alpha-1)} (\epsilon)^5 \Lambda^2}\right)$	$\widetilde{O}\left(\frac{n\left(\sum_{k=1}^{K} \lvert\binom{\alpha-1}{k}\rvert\right)^3}{\lvert 1-\alpha\rvert^3 [\mathrm{tr}(\rho^2)]^{3(\alpha-1)} (\epsilon)^3 \Lambda^2}\right)$
$(1,2]$	$\widetilde{O}\left(\frac{\left(\sum_{k=1}^{K} \lvert\binom{\alpha-1}{k}\rvert\right)^5}{\lvert 1-\alpha\rvert^5 [\mathrm{tr}(\rho^2)]^5 (\epsilon)^5 \Lambda^2}\right)$	$\widetilde{O}\left(\frac{3\left(\sum_{k=1}^{K} \lvert\binom{\alpha-1}{k}\rvert\right)^3}{\lvert 1-\alpha\rvert^3 [\mathrm{tr}(\rho^2)]^3 (\epsilon)^3 \Lambda^2}\right)$

Table 5.2: Cost estimation of Algorithm 7

Proof. Correctness analysis.

Recall the relation between $R_\alpha(\rho)$ and $\mathrm{tr}(\rho^\alpha)$. Thus, we focus on analyzing the estimation of $\mathrm{tr}(\rho^\alpha)$. For this purpose, we write the Fourier series $F_\alpha(\rho)$ as follows.

$$F_\alpha(\rho) = 1 + \sum_{l=0}^{\lfloor L \rfloor} \sum_{s=D_l}^{U_l} f(s,l)\mathrm{tr}\left(\rho \cos\left(\rho \frac{(2s-l)\pi}{2}\right)\right). \tag{5.132}$$

Coefficients $f(s,l)$ are given by

$$f(s,l) = \left(\sum_{k=1}^{K} b_l^{(k)} (-1)^k \binom{\beta}{k}\right) \frac{\binom{l}{s}}{2^l}, \quad \forall s,l. \tag{5.133}$$

where $\beta = \alpha - 1$. Let $\mathbf{f}$ be a vector that consists of $f(s,l)$. Then the ℓ_1-norm of $\mathbf{f}$ is bounded, i.e., $\|\mathbf{f}\|_{\ell_1} \in O\left(\sum_{k=1}^{K} |\binom{\beta}{k}|\right)$. In addition, the bound on $\sum_{k=1}^{K} |\binom{\beta}{k}|$ can be found in Lemma 8.

Next, define an importance sampling as follows:

$$\mathbf{R} = \mathrm{tr}\left(\rho \cos\left(\rho \frac{(2s-l)\pi}{2}\right)\right) \quad \text{with prob. } \frac{|f(s,l)|}{\|\mathbf{f}\|_{\ell_1}}. \tag{5.134}$$

The random variable $\mathbf{R}$ indicates that each Fourier term associated with (s,l) is sampled with probability proportional to its weight $f(s,l)$. Then, the Fourier series could be rewritten as an expectation of $\mathbf{R}$.

$$F(\rho) = 1 + \|\mathbf{f}\|_{\ell_1} \cdot \mathbf{E}[\mathbf{R}]. \tag{5.135}$$

Cost analysis.

Note that the sample mean could estimate the expectation. The estimation accuracy replies on the number of samples. By Chebyshev's inequality, $O(\mathbf{V}ar/\epsilon^2)$ samples are sufficient to derive an estimate of the expectation with precision ϵ and high probability, where $\mathbf{V}ar$ denotes the variance, and ϵ is the precision. Meanwhile, the probability could be boosted to $1-\delta$ at the cost of an additional multiplicative factor $O(\log(1/\delta))$ according to Chernoff bounds. Alternatively, by Hoeffding's inequality, we only need $O(\log(1/\delta)/\epsilon^2)$ samples to derive an estimate with precision ϵ and probability larger than $1-\delta$.

Regarding random variable $\mathbf{R}$, the variance is less than 1. We set the precision as $\varepsilon = \xi/\|\mathbf{f}\|_{\ell_1}$ and failure probability $\delta/2$. Then the number of required samples is

$$\#\mathrm{Samples} = O\left(\frac{1}{\varepsilon^2}\log\left(\frac{2}{\delta}\right)\right) = O\left(\frac{\|\mathbf{f}\|_{\ell_1}^2}{\xi^2}\log\left(\frac{2}{\delta}\right)\right). \tag{5.136}$$

It means that there are at most $\#\mathrm{S}amples$ terms needing to estimate via quantum circuits.

On the other hand, notice that the Fourier series $F(\rho)$ consists of $N = \sum_{l=0}^{L}(2M_l + 1)$ terms in all. Hence, it suffices to estimate each term with probability $1 - \delta/2N$. In this way, the overall failure probability is at most δ by union bound.

When estimating the Fourier series, we need to measure the resultant state after evolving the input state ρ by the circuits (e.g., please refer to Figure 5.2). Note that the measurement outcome is evaluated nondeterministically, which implies that the estimation could fail. To suppress the failure probability to δ, we suffice to evaluate each term with a failure probability at most $\delta/2N$. In consequence, for each term, the number of needed measurements is

$$\#\mathrm{M}easurements = O\left(\frac{\log(2N/\delta)}{\varepsilon^2}\right). \tag{5.137}$$

Immediately, the total number of measurements is at most

$$C_m = \#\mathrm{S}ample \times \#\mathrm{M}easurements = O\left(\frac{\|\mathbf{f}\|_{\ell_1}^4}{\xi^4}\log\left(\frac{2}{\delta}\right)\log\left(\frac{N}{\delta}\right)\right). \tag{5.138}$$

Now, we consider the number of copies of state ρ. Note that running the circuit once costs $\widetilde{O}(t^2/\epsilon)$ by Proposition 10. Here, the largest time is $O(M_L) = O\left(\ln\left(\frac{4\sum_{k=1}^{K}|\binom{\beta}{k}|}{\xi}\right)\frac{1}{\Lambda}\right)$. As shown above, we have to run circuits C_m times in the entropy estimation, then the number of overall copies is at most

$$C_\rho = C_m \times \widetilde{O}\left(\ln^2\left(\frac{\sum_{k=1}^{K}|\binom{\beta}{k}|}{\epsilon}\right)\frac{1}{\varepsilon\Lambda^2}\right) = \widetilde{O}\left(\frac{\|\mathbf{f}\|_{\ell_1}^5}{\xi^5\Lambda^2}\right). \tag{5.139}$$

By Proposition 12, for $\mathrm{tr}(\rho\cos(\rho t))$, the number of primitive single/two-qubit gates scales $O(nt^2/\epsilon)$. Here, the overall gate counts for the entropy estimation, in the worst case, is

$$C_g = \#\mathrm{S}ample \times O(nM_L^2/\varepsilon) = \widetilde{O}\left(\frac{n\|\mathbf{f}\|_{\ell_1}^3}{\xi^3\Lambda^2}\right). \tag{5.140}$$

Finally, using the relation between ξ and ϵ in Eq. (5.106), we can finish the proof for the claimed. ■

Remark 9 From results in Theorems 2 & 3, we can easily see that the costs of our algorithms scale polynomially in $1/\Lambda$, where Λ is the lower bound on all non-zero eigenvalues. If Λ is polynomially tiny, i.e., $\Lambda = \Omega(1/poly(n))$, the number of needed quantum states and primitive gates could be polynomially large. Hence, our algorithms can apply to large states once the states satisfy the aforementioned assumption. For example, it may happen when a large state has a low rank.

5.4 Numerical results

In this section, we conduct the numerical simulation to demonstrate the effectiveness and correctness of our algorithms. Specifically, we estimate $S(\rho)$ and $R_2(\rho)$ for several randomly generated single-qubit state ρ. And, we estimate the entropy of a single-qubit state under the effect of depolarizing noises and amplitude damping noises. All simulation experiments are operated on *Paddle Quantum* platform.

5.4.1 Effectiveness and correctness

To begin with, we generate a single-qubit mixed state at random and denote it by $\rho = \begin{pmatrix} 0.48786 & 0.0094 \\ 0.0094 & 0.51214 \end{pmatrix}$. As the eigenvalues of ρ are larger than 0.35, we set the eigenvalue lower bound as $\Lambda = 0.35$ (as long as smaller than the minimum non-zero eigenvalue of the state ρ). In the experiment, we set the error tolerance $\epsilon = 0.2, 0.4$, respectively. The results are depicted in Figure 5.6.

In Figure 5.6, the coloured curves represent the estimates of the entropy with the error tolerance $0.2, 0.4$. The shadowed areas represent standard deviation. Clearly, both blue and orange curves fluctuate around the black dashed line, but the blue curve ($\epsilon = 0.2$) is closer. In addition to that, the shadowed areas converge as the number of sampled points increases, meaning the estimation gets precise. Hence, we could conclude that our method could estimate the entropy precisely with a large number of sampled points and small error tolerance ϵ.

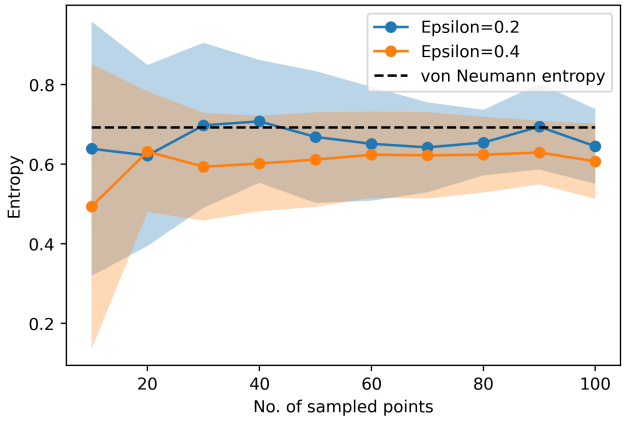
Next, we show the effectiveness with more quantum states. We randomly generate four more single-qubit mixed states to match the lower bound Λ equal to 0.35. All selected density matrices are shown below.

$$\rho_1 = \begin{pmatrix} 0.37336237 & -0.02597119 \\ -0.02597119 & 0.62663763 \end{pmatrix}, \rho_2 = \begin{pmatrix} 0.42050704 & -0.08174482 \\ -0.08174482 & 0.57949296 \end{pmatrix}, \tag{5.141}$$

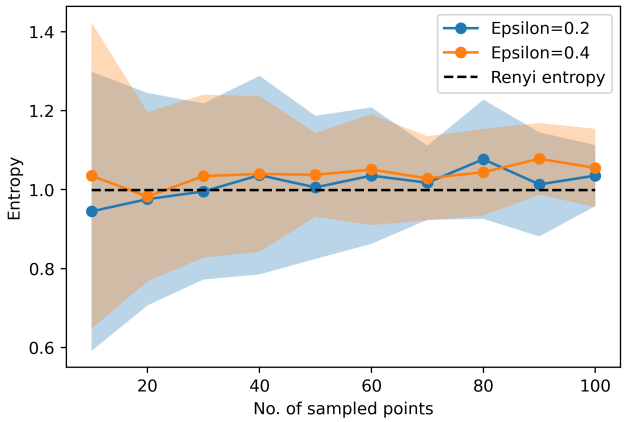
$$\rho_3 = \begin{pmatrix} 0.58221067 & -0.04587666 \\ -0.04587666 & 0.41778933 \end{pmatrix}, \rho_4 = \begin{pmatrix} 0.42932114 & -0.02696812 \\ -0.02696812 & 0.57067886 \end{pmatrix}. \tag{5.142}$$

We set the number of sampled points as 100 and error tolerance $\epsilon = 0.2$. The corresponding results are illustrated in Figure 5.7.

In Figure 5.7, the blue bar is the true quantum entropy corresponding to $S(\rho)$ and $R_2(\rho)$. The red bar is the value calculated by the Fourier series (cf. Lemmas 7 & 8). The green bar denotes the estimate of the entropy by our approach. Clearly, the red and



(a) Estimated von Neumann entropy.



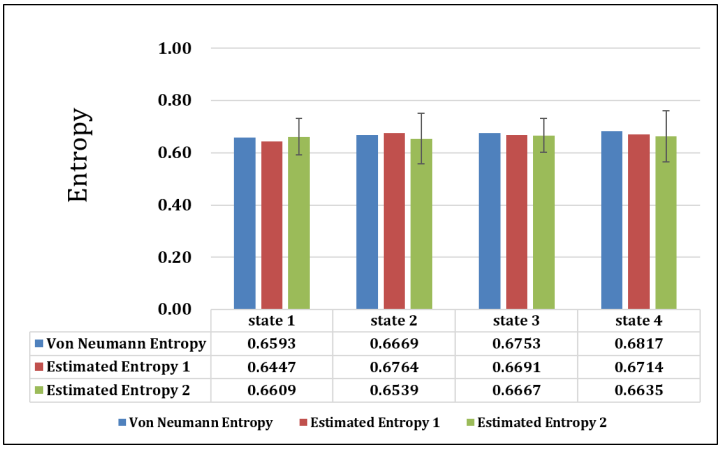
(b) Estimated 2-Rényi entropy.

Figure 5.6: In (a) and (b), the black dashed line represents the actual entropy of quantum state ρ. The blue and orange curves are average entropy over 20 repeats for ϵ equal to 0.2 and 0.4, respectively. The shadowed area stands for standard deviation.

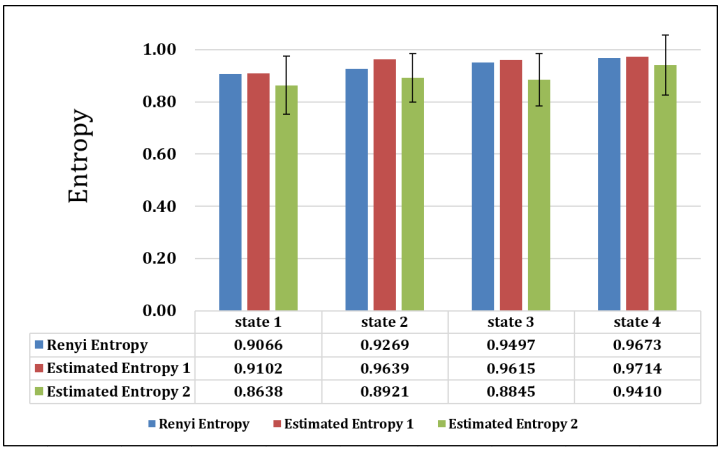
green bars are very close to the blue bar for different states. Thus, the experimental results show that our approach could find high-precision estimates for generated states, implying the validity of the Fourier series and our algorithms.

5.4.2 Robustness

We also study the performance of our approach under the effect of noises. Specifically, we consider a single qubit quantum state ρ and single-qubit amplitude damping channel



(a) Estimated von Neumann entropy.



(b) Estimated Rényi entropy.

Figure 5.7: The results for 4 randomly generated states. In (a) and (b), the blue bar is the real quantum entropy, the Estimated Entropy 1 stands for the entropy corresponding to the Fourier series approximation, and the Estimated Entropy 2 is the average entropy (100 sample points, repeat 20 times) calculated by our approach. In addition, the error bar represents the standard deviation.

$\mathcal{N}_p^{amp}(\rho)$ and depolarizing channel $\mathcal{N}_p^{depl}(\rho)$. The noisy quantum channels are given by

$$\mathcal{N}_p^{amp}(\rho) := D_0\rho D_0^\dagger + D_1\rho D_1^\dagger, \tag{5.143}$$

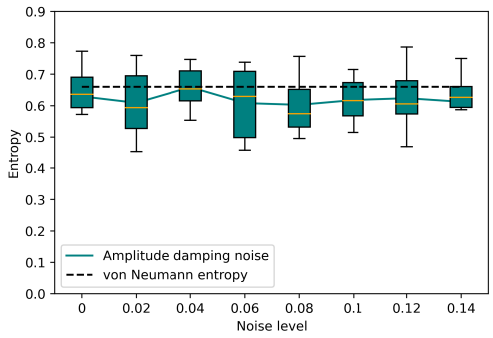
$$\mathcal{N}_p^{depl}(\rho) := E_0\rho E_0^\dagger + E_1\rho E_1^\dagger + E_2\rho E_2^\dagger + E_3\rho E_3^\dagger, \tag{5.144}$$

where

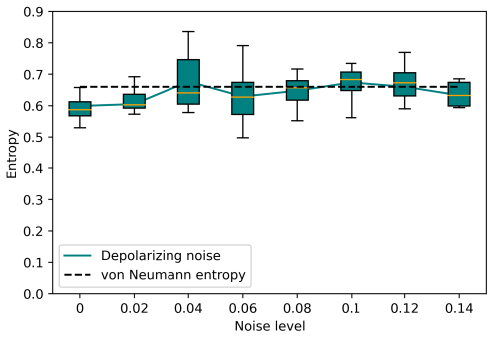
$$D_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-p} \end{bmatrix}, D_1 = \begin{bmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{bmatrix}, E_0 = \sqrt{1-p}\,I, E_1 = \sqrt{\frac{p}{3}}X, E_2 = \sqrt{\frac{p}{3}}Y, E_3 = \sqrt{\frac{p}{3}}Z.$$

Here, the parameter $p \in [0,1]$ is the noise level. Notation I stands for identity operator, and X, Y, Z are Pauli matrices.

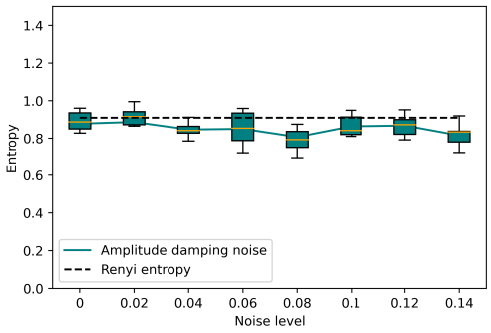
In the experiment, the input the quantum state is $\rho = \begin{pmatrix} 0.5398 & -0.1217 \\ -0.1217 & 0.4602 \end{pmatrix}$. We set the noise level p as 0, 0.02, 0.04, 0.06, 0.08, 0.10, 0.12, 0.15, respectively. And we set the error tolerance $\epsilon = 0.2$ and eigenvalue lower bound $\Lambda = 0.35$ and take 100 sample points. The results are shown in Figure 5.8.



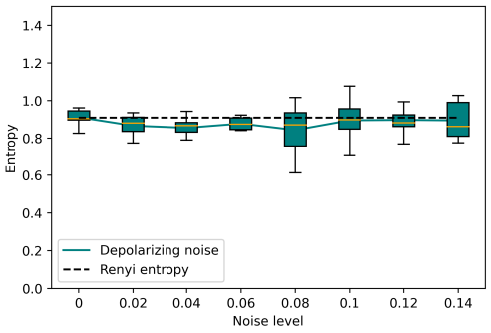
(a) $S(\rho)$ with amplitude damping noises.



(b) $S(\rho)$ with depolarizing noises.



(c) $R_2(\rho)$ with amplitude damping noises.



(d) $R_2(\rho)$ with depolarizing noises.

Figure 5.8: Figures (a) and (b) represent results for von Neumann entropy, and (c) and (d) represent the results for 2-Rényi entropy. The green curves link the average estimated entropy at different noise levels. The black dashed line represents the actual von Neumann entropy of quantum state ρ.

In Figure 5.8, (a) and (b) ((c) and (d)) are the box plots of $S(\rho)$ ($R_2(\rho)$) corresponding to amplitude damping noise and depolarizing noise, respectively. The black dashed line represents the true value of the entropy. The green curve represents the estimates of the entropy. As shown, all plots fluctuate around the black dashed line. Hence, we can

confidently announce that our algorithm is robust to intermediate amplitude damping and depolarizing noises.

In addition, the green curve in (b) and (d) is closer to the black dashed line than (a) and (c). Thus it implies that our algorithm performs better under depolarizing noises than amplitude damping noises for these chosen states.

5.5 Comparison to literature

The following discusses the differences between our results and the previous related works [2, 36, 56, 66, 100, 159].

First, algorithms of [36, 56, 66, 100] require an oracle for preparing the purification of the input state. In other words, one has to find such quantum circuits for preparing the purification before performing the entropy estimation task. However, implementing the oracle for an unknown quantum state via quantum circuits is not easy in general. In contrast, our algorithm no longer require the oracle for implementing the circuits and only require access to the copies of the states, making our algorithms more practical than that of [36, 56, 66, 100].

Second, compared to [2, 56, 66, 100], our algorithms could be more resource-efficient when the large state's minimal nonzero eigenvalue is polynomially small. The costs in previous works [56, 66, 100] are characterized in terms of the times of querying the oracle, depending on the dimension of the state. Meanwhile, the copy cost of [2] is exponentially large. In comparison, by Theorem 2 & 3, the copy costs of our algorithms scale polynomially with respect to the number of qubits n when parameter $\Lambda = \Omega(1/poly(n))$.

Third, the approach in [159] can estimate the quantum Rényi entropy $R_\alpha(\alpha)$ when the parameter α is integer. In comparison, our approach is more general, i.e., our approach can apply to the case when α is an integer or non-integer.

5.6 Applications

Given the importance of quantum entropies, our algorithms will have various applications in science and engineering. Here, we discuss the applications in low-rank quantum states, quantum Gibbs state preparation and entanglement entropy estimation.

State preparation quality assessment. The class of low-rank quantum states is particularly significant in physical experiments, forming a realistic model of quantum states

prepared in the lab [27]. For instance, one important task is to prepare pure quantum states. In general, the prepared state tends to have rapidly decaying eigenvalues such that a low-rank state can well approximate it. If the low-rank state's quantum entropy (disorder) can be quantified, we can assess the quality of the state preparation. Note that the generated state has a small number of significant eigenvalues, and the rests are close to zero. In this case, the generated state's eigenvalues with exponential scaling in the qubit counts can be ignored. Thus it is reasonable to assume that the minimal non-zero eigenvalue of the low-rank state is polynomially small in the worst case. At that time, the low-rank states satisfy the condition of applying our algorithms. Hence our algorithms can be applied to assess the quality of state preparation in the lab.

Gibbs state preparation. Quantum Gibbs states or thermal states are of significant importance to quantum simulation [35], quantum machine learning [18, 85], and quantum optimizaiton [133], etc. Several recent works [36, 148] have proposed to use variational quantum algorithms to prepare the quantum Gibbs states. The core idea of these works is to train a parameterized quantum circuit (PQC) to generate a parameterized state $\rho(\boldsymbol{\theta})$ matching the global minimum of the system's free energy. Specifically, the task is to minimize the loss function $L(\boldsymbol{\theta}) = \mathrm{tr}(H\rho(\boldsymbol{\theta})) - \beta^{-1}S(\rho(\boldsymbol{\theta}))$ by a gradient-descent method, where the notation H denotes the system's Hamiltonian, and β denotes the inverse temperature. One feasible scheme for estimating the gradient is the difference method, which demands efficient loss evaluation. As the loss evaluation involves the von Neumann entropy estimation, our algorithm could be employed for the gradient estimation. Besides, the PQC can be considered as a method to generate quantum state, which guarantees that we have the access to the target state freely.

Entanglement entropy estimation Quantum entanglement is a fundamental concept in quantum physics. It also is a central resource in quantum information many quantum information applications such as teleportation, super-dense coding, and quantum key distribution [46]. Thus, developing methods for entanglement quantification will be of great importance to the study in these fields. Notice that the entropy of entanglement [14] is the ideal entanglement measure for quantifying bipartite entanglement since the rate of the transformation between any bipartite pure state $|\psi\rangle_{AB}$ and two-qubit singlet is given by its entropy of entanglement, i.e., von Neumann entropy of the subsystem $S(\mathrm{tr}_B(|\psi\rangle\langle\psi|_{AB}))$. Hence, our approach can be employed to measure the entanglement via the von Neumann entropy estimation.

CHAPTER 6

CONCLUSION AND FUTURE WORK

In this dissertation, we developed an understanding of quantum Hamiltonian learning concerning the implementation in practice. Especially, we consider two closely related problems and provide concrete algorithms as solutions.

Specifically, in chapter 3, we proposed a hybrid quantum-classical Hamiltonian learning algorithm that employs a gradient-descent method to find the desired interaction coefficients. We achieved this purpose by unifying the variational quantum algorithms (VQAs) with the strategy proposed in [5]. To this end, we developed several subroutines: log-partition function estimation, stochastic variational quantum eigensolver (SVQE), and gradient estimation.

In SVQE, we proposed a method to learn the full/partial spectrum of the Hamiltonian and used the importance sampling to circumvent the resources in the loss evaluation. In the log-partition function, we proposed a method that combines the parameterized quantum circuits and convex optimization to find the global minimum of the free energy as well as compute the log-partition function. In gradient estimation, we presented a procedure to compute the gradient of the objective function costing polynomially many resources.

We conducted numerical experiments to demonstrate the effectiveness of our approach with randomly generated Hamiltonians and selected many-body Hamiltonians. In consequence, we showed that learning the full spectrum of Hamiltonians in the learning process could produce high-precision estimates of the desired interaction coefficients. Moreover, we also showed that partially learning several smallest eigenvalues of Ising

Hamiltonians could derive estimates up to a precision of 0.05.

In chapter 4, we provided hardware-efficient variational algorithms for quantum Gibbs state preparation with NISQ devices. We designed loss functions to approximate the free energy of a given Hamiltonian by utilizing the truncated Taylor series of the von Neumann entropy. By minimizing the loss functions, the parameterized quantum circuits could be trained to learn the Gibbs state via variational algorithms since the Gibbs state minimizes free energy. In particular, we showed that both the loss functions and their gradients could be evaluated on NISQ devices, thus allowing us to implement the hybrid quantum-classical optimization via either gradient-based or gradient-free optimization methods.

Moreover, we showed that our method could efficiently prepare the Gibbs states via analytical evidence and numerical experiments. We further showed that our variational algorithms work efficiently for many-body models, including the Ising chain and spin chain. In particular, we showed that the preparation of the Ising Gibbs state could be done efficiently and accurately via shallow parameterized quantum circuits with only one parameter and one additional qubit.

In chapter 5, we provided quantum algorithms based on the Fourier series for estimating the von Neumann entropy and α-Rényi entropy using independent copies of the state. The key of our algorithms is the quantum circuits that can efficiently evaluate the terms of the Fourier series and classically post-processing the measurement outcomes. Especially, quantum circuits are composed of primitive single/two-qubit gates. The design of quantum circuits synthesizes several quantum tools, such as iterative quantum phase estimation, the exponentiation of quantum state, qubit reset, and the linear combination of unitaries.

The notable property of our algorithm is that the circuits does not use oracles, which makes our circuit more friendly to NISQ devices than the oracle-based quantum algorithms. Besides, our circuits are only determined by parameters Λ and ϵ, allowing the same quantum circuits to estimate multiple different quantum states. Furthermore, when the input state's minimal non-zero eigenvalues are polynomially scaling in the qubit counts, i.e., $\Lambda = \Omega(1/poly(n))$, our algorithms only use polynomially many copies and thus have potential applications to large states processing. Given these merits, our algorithm may be expected to complete the entropy estimation task on the near and intermediate future quantum computing devices.

We believe our approaches for Hamiltonian learning, quantum Gibbs state preparation, and quantum entropy estimation would shed light on near-term quantum applica-

tions. Our SVQE algorithm might enrich the VQE family in the fields of molecules and materials. As many problems in computer science can be framed as partition function problems (e.g., counting colouring), our method may also contribute to these fields. Given that many problems of practical interest are related to Gibbs state and quantum entropy, it is natural to devise more quantum algorithms that are friendly to NISQ devices to solve problems related to quantum entropies. These problems could be in the areas of optimization (combinatorial optimization problems [133], semi-definite programs [24]), quantum machine learning (Hamiltonian learning [147]), many-body physics, quantum chemistry, etc. We also hope our results will advance the applications of NISQ devices in the study of condensed matter physics, high-energy physics, and gravity and black holes theory [11, 41].

Beyond the potential applications mentioned above, other problems regarding our algorithms have not been resolved in this dissertation. The most important is the barren plateau problem. Despite numerical results imply that our algorithms are successful in small problems, solving large problems will always be the ultimate goal. However, recent researches found out that VQA for large-scaled problems may encounter the so-called *barren plateau phenomena* [33, 102, 105, 145], where the cost landscape is too flat, leading to find the optimal parameters in an unacceptable time. A theoretical analysis of barren plateaus is provided in Appendix A.6.

In the meantime, exponential computational resources may be incurred, which is not suitable for demonstrating the superiority of quantum computing over classical computing. For example, suppose a gradient-descent method is employed in VQA, which exploits a PQC with deep depth. In that case, the gradient will vanish exponentially fast since the magnitude of the gradient decreases with exponential scaling in the qubit counts [105].

Even though it is quite challenging to solve the barren plateaus, recent progress in VQA shows the hope of success in the future. For instance, [97] shows that the tensor network-based machine learning model does not exhibit barren plateaus for local cost function. The absence of barren plateau has been shown in the quantum convolution neural network [115]. For quantum generative models, [84] discusses the usage of Rényi divergence of order two can avoid the barren plateau in thermal state learning and Hamiltonian learning tasks. These works imply that certain parameterized quantum circuits and cost function may escape from the barren plateaus. Moreover, there are strategies [61] focus on finding clever initial parameters to tackle the barren plateau. However, the aforementioned strategies cannot solve the barren plateaus in complete.

Hence, much more effort must be paid in future work.

Overall speaking, we provide concrete algorithms to solve Hamiltonian learning problems, which will be effective upon more robust quantum computers are available. We hope this dissertation could provide insights into the Hamiltonian learning problems and the application of the variational quantum algorithm. We wish our methods could find further applications in quantum physics and chemistry and beyond quantum computing.

APPENDIX A

APPENDIX

A.1 Supplementary proofs

Proofs for Eqs. (3.7)-(3.8) Consider a Hamiltonian $H \in \mathbb{C}^{N\times N}$ and a constant $\beta > 0$, then the system's free energy is given by $F(\rho) = \mathrm{tr}(H\rho) - \beta^{-1}S(\rho)$. Recall the fact [111] that

$$S(\rho) \leq -\sum_{j=1}^{N} \rho_{jj} \log \rho_{jj}, \tag{A.1}$$

where ρ_{jj} are the diagonal elements of quantum state ρ. Using this fact, for any state ρ, we can find a lower bound on free energy in the sense that

$$F(\rho) \geq \mathrm{tr}(H\rho) + \beta^{-1} \sum_{j=1}^{N} \rho_{jj} \log \rho_{jj}. \tag{A.2}$$

On the other hand, let U be a unitary such that $H = U\Lambda U^{\dagger}$, where $\Lambda = diag(\lambda_1, ..., \lambda_N)$ is a diagonal matrix. Let $\widetilde{\rho} = diag(\rho_{11}, ..., \rho_{NN})$ be the diagonal matrix consisting of ρ's diagonal elements and let $\sigma = U^{\dagger}\widetilde{\rho}U$. It is easy to verify that $\mathrm{tr}(H\rho) = \mathrm{tr}(\Lambda\sigma)$. Furthermore, taking this relation into Eq. (A.2)'s right hand side, we can find that

$$F(\rho) \geq \mathrm{tr}(\Lambda\sigma) - \beta^{-1}S(\sigma). \tag{A.3}$$

Notice that Eq. (A.3)'s right-hand side is equal to $F(\widetilde{\rho})$, then we have

$$F(\rho) \geq F(\widetilde{\rho}). \tag{A.4}$$

The inequality in Eq. (A.4) shows that free energy's global optimum is commuting with the Hamiltonian H.

According to the above discussion, we can rewrite the optimization program of finding free energy's minimal value as follows

$$\min_{\rho} F(\rho) = \min_{\mathbf{p}} \sum_{j=1}^{N} \lambda_j p_j + \beta^{-1} \sum_{j=1}^{N} p_j \log p_j, \tag{A.5}$$

where $\mathbf{p}$ represents an arbitrary probability distribution. Eq. (A.5)'s right-hand side can be solved using the Lagrange multiplier method, and the optimum is given below:

$$\mathbf{p}^* = \frac{1}{Z}(e^{-\beta\lambda_1}, ..., e^{-\beta\lambda_N}), \tag{A.6}$$

with $Z := \sum_{j=1}^{N} e^{-\beta\lambda_j}$.

Finally, the equalities in Eqs. (9)-(10) can be proved by taking $\mathbf{p}^*$ into Eq. (A.5)'s right-hand side and computing the minimal value.

Proof for Proposition 1

Lemma 9. *For any parameterized Hamiltonian $H(\boldsymbol{\nu}) = \sum_{\ell=1}^{m} \nu_\ell E_\ell$ with $E_\ell \in \{X, Y, Z, I\}^{\otimes n}$, we have*

$$\| H(\boldsymbol{\nu}) \| \leq \sqrt{m} \cdot \| \boldsymbol{\nu} \|_2 . \tag{A.7}$$

where $\| \cdot \|$ denotes the spectral norm and $\| \cdot \|_2$ is the ℓ_2-norm.

Proof. Let U be the unitary that diagonalizes the Hamiltonian $H(\boldsymbol{\nu})$, and then we can use the following form to represent $H(\boldsymbol{\nu})$.

$$H(\boldsymbol{\nu}) = \sum_{j=1}^{N} \lambda_j \cdot U \left|\psi_j\right\rangle\!\left\langle\psi_j\right| U^\dagger, \tag{A.8}$$

where $|\psi_1\rangle, ..., |\psi_N\rangle$ are the computational basis.

Typically, each eigenvalue is represented as follows:

$$\lambda_j = \langle\psi_j| U^\dagger H(\boldsymbol{\nu}) U |\psi_j\rangle \tag{A.9}$$

$$= \sum_{\ell=1}^{m} \nu_\ell \langle\psi_j| U^\dagger E_\ell U |\psi_j\rangle \tag{A.10}$$

Then, applying the Cauchy-Schwarz inequality leads to an upper bound on each eigenvalue:

$$(\lambda_j)^2 \leq \sum_{\ell=1}^{m} (\nu_\ell)^2 \cdot \sum_{\ell=1}^{m} (\langle\psi_j| U^\dagger E_\ell U |\psi_j\rangle)^2. \tag{A.11}$$

Meanwhile, recalling that all E_ℓ are Pauli matrix tensor product, we can obtain an upper bound below:

$$(\lambda_j)^2 \leq m \sum_{\ell=1}^{m} (v_\ell)^2. \tag{A.12}$$

Ranging j in $\{1,...,N\}$ in Eq. (A.12), the maximal eigenvalue is upper bounded by $\sqrt{m}\|\boldsymbol{v}\|_2$, validating the claim. ■

Proof. Since the expression $\sum_{j=1}^{N} p_j \lambda_j$ is regarded as an expectation, then we can estimate it by the sample mean with high accuracy and probability. To be specific, let X denote a random variable that takes value λ_j with probability p_j. Then, this expression can be written as

$$\mathbf{E}[X] = \sum_{j=1}^{N} p_j \lambda_j. \tag{A.13}$$

Furthermore, recall Chebyshev's inequality, then we have

$$\Pr\left(|\bar{X} - \mathbf{E}[X]| \leq \epsilon\right) \geq 1 - \frac{\mathbf{Var}[X]}{T\epsilon^2}. \tag{A.14}$$

where $\bar{X} = \frac{1}{T}(X_1 + X_2 + ... + X_T)$ and $\mathbf{Var}[X]$ is the variance of X. Technically, we can set large T to increase the probability. Here, we only need to choose T such that

$$\frac{\mathbf{Var}[X]}{T\epsilon^2} = \frac{2}{3}. \tag{A.15}$$

Note that the second moment $\mathbf{E}[X^2]$ bounds the variance $\mathbf{Var}[X]$. Meanwhile, the second moment of X is bounded by the squared spectral norm of H, shown below.

$$\mathbf{E}[X^2] = \sum_{j=1}^{N} p_j (\lambda_j)^2 \tag{A.16}$$

$$\leq \sum_{j=1}^{N} p_j \|H(\boldsymbol{v})\|^2 \tag{A.17}$$

$$= \|H(\boldsymbol{v})\|^2. \tag{A.18}$$

The inequality is due to the fact that each eigenvalue is less than the spectral norm. Apply Lemma 9, then we will obtain an bound on T:

$$T = \frac{3\mathbf{Var}[X]}{2\epsilon^2} \leq \frac{3\mathbf{E}[X^2]}{2\epsilon^2} \leq \frac{3m\|\boldsymbol{v}\|_2^2}{2\epsilon^2}. \tag{A.19}$$

Lastly, according to the Chernoff bound, we can boost the probability to $1-\eta$ for any $\eta > 0$ by repeatedly computing the sample mean $O(\log(1/\eta))$ times and taking the median of all sample means. ■

Proof for Proposition 2

Lemma 10. *Consider a parameterized Hamiltonian $H(\boldsymbol{\nu}) = \sum_{\ell=1}^{m} \nu_\ell E_\ell$ with $E_\ell \in \{X, Y, Z, I\}^{\otimes n}$. For any unitary U and state $|\psi\rangle$, estimating the value $\langle\psi|U^\dagger H(\boldsymbol{\nu})U|\psi\rangle$ up to precision ϵ with probability at least $1-\eta$ requires a sample complexity of*

$$O\left(\frac{m\|\boldsymbol{\nu}\|_1^2 \log(m/\eta)}{\epsilon^2}\right). \tag{A.20}$$

Proof. First, we rewrite the value $\langle\psi|U^\dagger H(\boldsymbol{\nu})U|\psi\rangle$ as follows:

$$\langle\psi|U^\dagger H(\boldsymbol{\nu})U|\psi\rangle = \sum_{\ell=1}^{m} \nu_\ell \langle\psi|U^\dagger E_\ell U|\psi\rangle. \tag{A.21}$$

Second, we count the required number of measurements to estimate the value $\langle\psi|U^\dagger E_\ell U|\psi\rangle$ up to precision $\epsilon/\|\boldsymbol{\nu}\|_1$ with probability at least $1-\eta/m$, where $\|\cdot\|_1$ denotes the ℓ_1-norm. Since the Pauli operator, E_ℓ, has eigenvalues ± 1, we can partition E_ℓ's eigenvectors into two sets, corresponding to positive and negative eigenvalues, respectively. For convenience, we call the measurement outcome corresponding to eigenvalue 1 as the positive measurement outcome and the rest as the negative measurement outcome. We define a random variable X in the sense that

$$X = \begin{cases} 1, & \Pr[\text{Positive measurement outcome}] \\ -1, & \Pr[\text{Negative measurement outcome}] \end{cases} \tag{A.22}$$

It is easy to verify that $\mathbf{E}[X] = \langle\psi|U^\dagger E_\ell U|\psi\rangle$. Thus, an approach to compute value $\langle\psi|U^\dagger E_\ell U|\psi\rangle$ is computing an estimate for the expectation $\mathbf{E}[X]$. Meanwhile, consider that $\mathbf{E}[X^2] \leq 1$, then the required number of samples is $O(\|\boldsymbol{\nu}\|_1^2 \log(m/\eta)/\epsilon^2)$.

Lastly, for $\langle\psi|U^\dagger H(\boldsymbol{\nu})U|\psi\rangle$, the estimate's maximal error is $\|\boldsymbol{\nu}\|_1 \cdot \epsilon/\|\boldsymbol{\nu}\|_1 = \epsilon$. By union bound, the overall failure probability is less than $m \cdot \eta/m = \eta$. Thus, the claim is proved. ■

Proof. Let Y denote a random variable that takes value $\langle\psi_j|U^\dagger(\boldsymbol{\theta})H(\boldsymbol{\nu})U(\boldsymbol{\theta})|\psi_j\rangle$ with probability q_j, then the objective function $M(\boldsymbol{\theta})$ can be rewritten as

$$\mathbf{E}[Y] = M(\boldsymbol{\theta}). \tag{A.23}$$

By Chebyshev's inequality, the expectation can be computed by taking enough samples of Y and averaging them. Note that the variance of Y determines the number of samples, and the absolute value Y is less than the spectral norm $\|H(\boldsymbol{\nu})\|$, i.e., $|Y| \leq \|H(\boldsymbol{\nu})\|$. Along

with Lemma 9, it is easy to see that the required number of Y's samples for obtaining an estimate with error $\epsilon/2$ and probability larger than 2/3 is $T = O(m\|\boldsymbol{\nu}\|_2^2/\epsilon^2)$. Furthermore, by Chernoff bounds, the probability can be improved to $1-\eta/2$ at an additional cost of multiplicative factor of $D = O(\log(1/\eta))$.

On the other hand, each sample Y's value has to be determined by performing the measurement. Since $|\psi_j\rangle$ is a computational basis, hence Y can take at most 2^n different values. To ensure the probability for estimating $\mathbf{E}[Y]$ larger than $1-\eta$, the probability of each estimate $\langle\psi_j|U^\dagger(\boldsymbol{\theta})H(\boldsymbol{\nu})U(\boldsymbol{\theta})|\psi_j\rangle$ only needs to be at least $1-\eta/2^{n+1}$. By union bound, the overall failure probability is at most $\eta/2+\eta\cdot\frac{TD}{2^{n+1}}<\eta$ (For large Hamiltonians, the number of samples TD can be significantly less than dimension 2^n). Besides, according to Lemma 10, $\langle\psi_j|U^\dagger(\boldsymbol{\theta})H(\boldsymbol{\nu})U(\boldsymbol{\theta})|\psi_j\rangle$'s estimate within accuracy $\epsilon/2$ and probability $1-\eta/2^{n+1}$ requires a sample complexity of $O(m\|\boldsymbol{\nu}\|_1^2(n+\log(m/\eta))/\epsilon^2)$. Thus, the overall number of measurements is the product of the number of samples $TD = O(m\|\boldsymbol{\nu}\|_2^2\log(1/\eta)/\epsilon^2)$ and each sample's sample complexity $O(m\|\boldsymbol{\nu}\|_1^2(n+\log(m/\eta))/\epsilon^2)$. In other words, the objective function $M(\boldsymbol{\theta})$'s estimate within error ϵ and probability $1-\eta$ requires a sample complexity of

$$O\left(TD\cdot\frac{m\|\boldsymbol{\nu}\|_1^2(n+\log(m/\eta))}{\epsilon^2}\right)=O\left(\frac{m^2\|\boldsymbol{\nu}\|_1^2\|\boldsymbol{\nu}\|_2^2\log(1/\eta)(n+\log(m/\eta))}{\epsilon^4}\right).$$

■

Proof for Proposition 3

Lemma 11. *Let $\widehat{\boldsymbol{\lambda}}=(\widehat{\lambda}_1,...,\widehat{\lambda}_N)$ denote the estimated eigenvalues from SVQE and define a function $G(\mathbf{p})$ as follows:*

$$G(\mathbf{p})=\sum_{j=1}^N p_j\widehat{\lambda}_j+\beta^{-1}\sum_{j=1}^N p_j\log p_j. \tag{A.24}$$

Let $\widehat{\mathbf{p}}^*$ be the global optimal point of $G(\mathbf{p})$, that is, for any probability distribution $\mathbf{p}$, we have $G(\widehat{\mathbf{p}}^*)\le G(\mathbf{p})$. Meanwhile, suppose $\mathbf{p}^*$ is the global optimal point of $C(\mathbf{p})$. Then, we have

$$|G(\widehat{\mathbf{p}}^*)-C(\mathbf{p}^*)|\le\max\left\{\mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda}-\lambda|],\mathbf{E}_{\mathbf{p}^*}[|\widehat{\lambda}-\lambda|]\right\}, \tag{A.25}$$

where

$$\mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda}-\lambda|]=\sum_{j=1}^N\widehat{p}_j^*|\widehat{\lambda}_j-\lambda_j|, \tag{A.26}$$

$$\mathbf{E}_{\mathbf{p}^*}[|\widehat{\lambda}-\lambda|]=\sum_{j=1}^N p_j^*|\widehat{\lambda}_j-\lambda_j|. \tag{A.27}$$

Proof. Since functions $C(\mathbf{p})$ and $G(\mathbf{p})$ reach their global minimums at points $\mathbf{p}^*$ and $\widehat{\mathbf{p}}^*$ respectively, then we have

$$C(\widehat{\mathbf{p}}^*) \geq C(\mathbf{p}^*), \tag{A.28}$$

$$G(\widehat{\mathbf{p}}^*) \leq G(\mathbf{p}^*). \tag{A.29}$$

Besides, we also have another relation:

$$|C(\mathbf{p}) - G(\mathbf{p})| = \sum_{j=1}^{N} p_j |(\widehat{\lambda}_j - \lambda_j)|, \tag{A.30}$$

where $\|\cdot\|_\infty$ denotes the maximum norm.

Combining the above inequalities, we have the following result:

(A.31)
$$C(\mathbf{p}^*) \leq C(\widehat{\mathbf{p}}^*) \leq G(\widehat{\mathbf{p}}^*) + \mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda} - \lambda|] \leq G(\mathbf{p}^*) + \mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda} - \lambda|] \leq C(\mathbf{p}^*) + \mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda} - \lambda|] + \mathbf{E}_{\mathbf{p}^*}[|\widehat{\lambda} - \lambda|].$$

Then the inequality in Eq. (A.25) is proved. ■

Proof. Recalling the expressions of $C(\mathbf{p}^*)$ and $G(\widehat{\mathbf{p}}^*)$ in Eqs. (12) & (A.24), it is easy to verify the following inequalities:

$$F(\rho_\beta(\boldsymbol{v})) = C(\mathbf{p}^*), \tag{A.32}$$

$$F(\rho_\beta^*) = G(\widehat{\mathbf{p}}^*). \tag{A.33}$$

where F denotes the free energy, i.e., $F(\rho) = \mathrm{tr}(H\rho) - \beta^{-1}S(\rho)$.

Using the result in Lemma 11, we will obtain the following inequality.

$$|F(\rho_\beta^*) - F(\rho_\beta(\boldsymbol{v}))| = |G(\widehat{\mathbf{p}}^*) - C(\mathbf{p}^*)| \leq \max\left\{\mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda} - \lambda|], \mathbf{E}_{\mathbf{p}^*}[|\widehat{\lambda} - \lambda|]\right\}. \tag{A.34}$$

In the meanwhile, a property of the free energy says that

$$F(\rho_\beta^*) = F(\rho_\beta(\boldsymbol{v})) + \beta^{-1}S(\rho_\beta^* \| \rho_\beta(\boldsymbol{v})). \tag{A.35}$$

where $S(\rho_\beta^* \| \rho_\beta(\boldsymbol{v}))$ is the relative entropy. Rewriting the above equation as follows:

$$F(\rho_\beta^*) - F(\rho_\beta(\boldsymbol{v})) = \beta^{-1}S(\rho_\beta^* \| \rho_\beta(\boldsymbol{v})). \tag{A.36}$$

Combining the relations in Eqs. (A.34) and (A.36), we obtain the following inequality:

$$S(\rho_\beta^* \| \rho_\beta(\boldsymbol{v})) \leq \beta \max\left\{\mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda} - \lambda|], \mathbf{E}_{\mathbf{p}^*}[|\widehat{\lambda} - \lambda|]\right\}. \tag{A.37}$$

Lastly, according to Pinsker's inequality, the above inequality immediately leads to a bound on the trace distance between ρ_β and ρ_β^* in the sense that

$$D(\rho_\beta^*, \rho_\beta(\boldsymbol{v})) \leq \sqrt{2S(\rho_\beta^* \| \rho_\beta(\boldsymbol{v}))} \leq \sqrt{2\beta \max\left\{\mathbf{E}_{\widehat{\mathbf{p}}^*}[|\widehat{\lambda} - \lambda|], \mathbf{E}_{\mathbf{p}^*}[|\widehat{\lambda} - \lambda|]\right\}}. \tag{A.38}$$

The the claimed is proved. ■

Proof for Proposition 4

Proof. Let Z_ℓ denote the random variable that takes value $\langle\psi_j|U^\dagger(\boldsymbol{\theta})E_\ell U(\boldsymbol{\theta})|\psi_j\rangle$ with probability $\widehat{p}_j^*$, for all $\ell = 1,...,m$. Then we have

$$\mathbf{E}[Z_\ell] = \sum_{j=1}^{N} \widehat{p}_j^* \cdot \langle\psi_j|U^\dagger(\boldsymbol{\theta})E_\ell U(\boldsymbol{\theta})|\psi_j\rangle. \tag{A.39}$$

Thus partial derivative can be computed in the following way

$$\frac{\partial L(\boldsymbol{v})}{\partial v_\ell} \approx -\beta\mathbf{E}[Z_\ell] + \beta e_\ell. \tag{A.40}$$

It implies that the estimate's error can be set as ϵ/β to ensure the gradient's maximal error less than ϵ.

Next, we determine the number of samples such that the overall failure probability for estimating the gradient is less than δ. Since the gradient has m partial derivatives, corresping to $\mathbf{E}[Z_\ell]$, thus it suffices to estimate each with probability larger than $1-\delta/m$. Meanwhile, each mean $\mathbf{E}[Z_\ell]$ can be computed by sampling. Notice that all $|Z_\ell| \leq 1$, by Chebyshev's inequality, then it suffices to take $K = O(\beta^2/\epsilon^2)$ samples to compute an estimate for each $\mathbf{E}[Z_\ell]$ with precision $\epsilon/2\beta$ and probability larger than 2/3. Furthermore, by Chernoff bounds, the probability can be improved to $1-\eta/2m$ at an additional cost of multiplicative factor of $D = O(\log(2m/\eta))$. It is worth pointing out that, for each variable Z_ℓ, the samples are taken according to the same probability distribution $\widehat{\mathbf{p}}^*$, thus it is natural to use the sampled states $|\psi_{t_j^s}\rangle$ (cf. Algorithm 4) to compute all means $\mathbf{E}[Z_\ell]$. Then the total number of samples is $KD = O(\beta^2\log(m/\eta)/\epsilon^2)$.

On the other hand, each value $\langle\psi_j|U^\dagger(\boldsymbol{\theta})E_\ell U(\boldsymbol{\theta})|\psi_j\rangle$ in Eq. (A.39) has to be computed by performing the measurement. Note that there are 2^n values $\langle\psi_j|U^\dagger(\boldsymbol{\theta})E_\ell U(\boldsymbol{\theta})|\psi_j\rangle$ in all. To ensure the mean estimate's failure probability less than $\eta/2m$, it suffices to suppress each value's failure probability to $\eta/2^{n+1}m$. Following the same discussion in Lemma 10, the estimate for value $\langle\psi_j|U^\dagger(\boldsymbol{\theta})E_\ell U(\boldsymbol{\theta})|\psi_j\rangle$ can be computed up to precision $\epsilon/2\beta$ using $O(\beta^2\log(2^{n+1}m/\eta)/\epsilon^2)$ measurements.

Regarding the failure probability, by union bound, the overall failure probability is at most $m\cdot(\eta/2m + KD\cdot\eta/2^{n+1}m)$, where KD is the number of samples $KD = O(\beta^2\log(m/\eta)/\epsilon^2)$. Especially, for larger Hamiltonians, the number of measurements is usually less than the dimension 2^n. Thus, the overall failre probability is less than η.

Lastly, the total number of measurements is given below:

$$m\cdot KD\cdot O(\beta^2\log(2^{n+1}m/\eta)/\epsilon^2) = O(m\beta^4\log(m/\eta)\log(2^{n+1}m/\eta)/\epsilon^4). \tag{A.41}$$

■

Proof for Eq. (5.110)

Lemma 12. *For any quantum state ρ, time $t > 0$, and integer $K > 1$, the trace norm of $\sum_{k\geq K} \frac{(-i\rho t)^k}{k!}$ is bounded by $\sqrt{2}t^K/K!$.*

Proof. Suppose ρ has a spectral decomposition

$$\rho = \sum_j \lambda_j \left|\mathbf{e}_j\right\rangle\!\left\langle\mathbf{e}_j\right|. \tag{A.42}$$

As the trace norm is the sum of all singular values, we could deduce that

$$\left\| \sum_{k\geq K} \frac{(-i\rho t)^k}{k!} \right\|_{tr} = \left\| \sum_j \sum_{k\geq K} \frac{(-i\lambda_j t)^k}{k!} \left|\mathbf{e}_j\right\rangle\!\left\langle\mathbf{e}_j\right| \right\|_{tr} \tag{A.43}$$

$$= \sum_j \left| \sum_{k\geq K} \frac{(-i\lambda_j t)^k}{k!} \right| \tag{A.44}$$

$$\leq \sum_j \frac{\sqrt{2}(\lambda_j t)^K}{K!} \tag{A.45}$$

$$\leq \sum_j \frac{\sqrt{2}\lambda_j (t)^K}{K!} \tag{A.46}$$

$$\leq \frac{\sqrt{2}t^K}{K!}. \tag{A.47}$$

Here, we have used facts that $\left|\sum_{k\geq K} \frac{(-i\lambda_j t)^k}{k!}\right| \leq \frac{\sqrt{2}(\lambda_j t)^K}{K!}$ for all λ_j, and $\sum_j \lambda_j = 1$.

To complete the proof, we show that, for any $x \in \mathbb{R}$ and integer $K > 1$,

$$\left| e^{-ix} - \sum_{k=0}^{K-1} \frac{(-ix)^k}{k!} \right| \leq \frac{\sqrt{2}|x|^K}{K!}. \tag{A.48}$$

Given arbitrary integer $K > 0$, we use Euler's formula and expand the triangle functions to the Taylor series. The error is given by

$$e^{ix} - \sum_{k=1}^{K} \frac{(ix)^k}{k!} = \cos(x) + i\sin(x) - \left[\sum_p (i)^{2p+1} \frac{x^{2p+1}}{(2p+1)!} + \sum_p (i)^{2p} \frac{x^{2p}}{(2p)!} \right] \tag{A.49}$$

$$= \cos(x) + i\sin(x) - \left[i\sum_p (-1)^p \frac{x^{2p+1}}{(2p+1)!} + \sum_p (-1)^p \frac{x^{2p}}{(2p)!} \right] \tag{A.50}$$

$$= \left[\cos(x) - \sum_p (-1)^p \frac{x^{2p}}{(2p)!} \right] + i \left[\sin(x) - \sum_p (-1)^p \frac{x^{2p+1}}{(2p+1)!} \right], \tag{A.51}$$

where $1 \leq p \leq [K/2]$.

Notice that $\sum_p (-1)^p \frac{x^{2p+1}}{(2p+1)!}$ and $\sum_p (-1)^p \frac{x^{2p}}{(2p)!}$ are truncated Taylor series of $\sin(x)$ and $\cos(x)$ up to order K, respectively. By the Taylor's theorem, we have

$$\cos(x) - \sum_p (-1)^p \frac{x^{2p}}{(2p)!} = \frac{\cos^{(K+1)}(\zeta)}{(K+1)!} x^{K+1}, \tag{A.52}$$

$$\sin(x) - \sum_p (-1)^p \frac{x^{2p+1}}{(2p+1)!} = \frac{\sin^{(K+1)}(\xi)}{(K+1)!} x^{K+1}. \tag{A.53}$$

where ζ and ξ are some values between 0 and x. Immediately, we have

$$\left| \cos(x) - \sum_p (-1)^p \frac{x^{2p}}{(2p)!} \right| \leq \frac{|x|^{K+1}}{(K+1)!}, \tag{A.54}$$

$$\left| \sin(x) - \sum_p (-1)^p \frac{x^{2p+1}}{(2p+1)!} \right| \leq \frac{|x|^{K+1}}{(K+1)!}. \tag{A.55}$$

Finally, the result immediately follows, and the proof is finished. ■

A.2 Variational algorithm for Gibbs state preparation with higher-order truncations

Here we present a variational algorithm for preparing the Gibbs state with K-truncated free energy. To illustrate our algorithm, we give some notations first. We let $A_{t_j} B_{t_j}$ denote the registers that store the states for estimating $\mathrm{tr}(\rho^t)$, where $t_j \in \Sigma_t$ and Σ_t includes all the indices of these registers.

A.3 Estimation of the higher-order gradients

Lemma S13. *Given a parameterized density operator $\rho(\boldsymbol{\theta})$, we have the following equality,*

$$\partial_{\theta_m} \mathrm{tr}(\rho(\boldsymbol{\theta})^3) = 3\partial_{\theta_m,1} \mathrm{tr}(\rho_1(\boldsymbol{\theta}) \otimes \rho_2(\boldsymbol{\theta}) \otimes \rho_3(\boldsymbol{\theta}) \cdot S_1 S_2), \tag{A.56}$$

where $\partial_{\theta_m,1}$ means the derivative is computed with respective to θ_m of the state stored in 1-th register, $\rho_j(\boldsymbol{\theta})$ is the state stored in j-th register, and $S_1 = \mathrm{SWAP}_{12} \otimes I_3$ and $S_2 = I_1 \otimes \mathrm{SWAP}_{23}$, and the SWAP_{ij} is the operator that swaps the state stored in i-th and j-th register.

Proof. To prove the claim, we need the following result, which we give the proof later.

$$\mathrm{tr}(\rho_1 \otimes \rho_2 \otimes \rho_3 \cdot S_1 S_2) = \overline{\mathrm{tr}(\rho_2 \otimes \rho_1 \otimes \rho_3 \cdot S_1 S_2)} = \overline{\mathrm{tr}(\rho_3 \otimes \rho_2 \otimes \rho_1 \cdot S_1 S_2)}. \tag{A.57}$$

Algorithm 8 Variational quantum Gibbs state preparation with truncation order K

1: Input: choose the ansatz of unitary $U(\boldsymbol{\theta})$, tolerance ε, truncation order K, and initial parameters of $\boldsymbol{\theta}$;
2: Compute coefficients C_0, C_1,..., C_K according to Eq. (4.6).
3: Prepare initial states $|00\rangle$ in registers AB and apply $U(\boldsymbol{\theta})$ to these states.
4: Measure and compute $\mathrm{tr}(H\rho_{B_1})$ and compute the loss function $L_1 = \mathrm{tr}(H\rho_{B_1})$;
5: Measure the overlap $\mathrm{tr}(\prod_{t_2\in\Sigma_2}\rho_{B_{t_2}})$ via Destructive Swap Test and compute the loss function $L_2 = -\beta^{-1}C_1\,\mathrm{tr}(\prod_{t_2\in\Sigma_2}\rho_{B_{t_2}})$;
6: Measure the overlap $\mathrm{tr}(\prod_{t_k\in\Sigma_k}\rho_{B_{t_k}})$ via higher order state overlap estimation and compute the loss function $L_k = -\beta^{-1}C_{k-1}\,\mathrm{tr}(\prod_{t_k\in\Sigma_k}\rho_{B_{t_k}})$ for each $k\in\{3,...,K+1\}$.
7: Perform optimization of $\mathcal{F}_K(\boldsymbol{\theta}) = \sum_{k=1}^{K+1}L_k - \beta^{-1}C_0$ and update parameters of $\boldsymbol{\theta}$;
8: Repeat 3-7 until the loss function $\mathcal{F}_K(\boldsymbol{\theta})$ converges with tolerance ε;
9: Output the state $\rho^{out} = \mathrm{tr}_A\,U(\boldsymbol{\theta})|00\rangle\langle 00|_{AB}U(\boldsymbol{\theta})^\dagger$.

Let $\rho_1(\boldsymbol{\theta}) = \rho_2(\boldsymbol{\theta}) = \rho_3(\boldsymbol{\theta}) = \rho(\boldsymbol{\theta})$, then the claimed is proved in the following,

$$\begin{aligned}
\frac{\partial}{\partial\theta_m}\mathrm{tr}(\rho(\boldsymbol{\theta})^3) &= \frac{\partial}{\partial\theta_m}\mathrm{tr}(\rho_1(\boldsymbol{\theta})\otimes\rho_2(\boldsymbol{\theta})\otimes\rho_3(\boldsymbol{\theta})\cdot S_1S_2) && \text{(A.58)}\\
&= \frac{\partial}{\partial\theta_{m,1}}\mathrm{tr}(\rho_1(\boldsymbol{\theta})\otimes\rho_2(\boldsymbol{\theta})\otimes\rho_3(\boldsymbol{\theta})\cdot S_1S_2)\\
&\quad+\frac{\partial}{\partial\theta_{m,2}}\mathrm{tr}(\rho_1(\boldsymbol{\theta})\otimes\rho_2(\boldsymbol{\theta})\otimes\rho_3(\boldsymbol{\theta})\cdot S_1S_2)\\
&\quad+\frac{\partial}{\partial\theta_{m,3}}\mathrm{tr}(\rho_1(\boldsymbol{\theta})\otimes\rho_2(\boldsymbol{\theta})\otimes\rho_3(\boldsymbol{\theta})\cdot S_1S_2) && \text{(A.59)}\\
&= \frac{\partial}{\partial\theta_{m,1}}\mathrm{tr}(\rho_1(\boldsymbol{\theta})\otimes\rho_2(\boldsymbol{\theta})\otimes\rho_3(\boldsymbol{\theta})\cdot S_1S_2)\\
&\quad+\frac{\partial}{\partial\theta_{m,2}}\mathrm{tr}(\rho_2(\boldsymbol{\theta})\otimes\rho_1(\boldsymbol{\theta})\otimes\rho_3(\boldsymbol{\theta})\cdot S_1S_2)\\
&\quad+\frac{\partial}{\partial\theta_{m,3}}\mathrm{tr}(\rho_3(\boldsymbol{\theta})\otimes\rho_2(\boldsymbol{\theta})\otimes\rho_1(\boldsymbol{\theta})\cdot S_1S_2) && \text{(A.60)}\\
&= 3\frac{\partial}{\partial\theta_{m,1}}\mathrm{tr}(\rho_1(\boldsymbol{\theta})\otimes\rho_2(\boldsymbol{\theta})\otimes\rho_3(\boldsymbol{\theta})\cdot S_1S_2), && \text{(A.61)}
\end{aligned}$$

where the equality is Eq. (A.59) is the result of chain rule, and we use the relation in Eq. (A.57) to derive the equality in Eq. (A.60).

Now we prove the equality in Eq. (A.57).

Let $\rho_1 = \sum_j p_j|\phi_j\rangle\langle\phi_j|$, $\rho_2 = \sum_l q_l|\psi_l\rangle\langle\psi_l|$, and $\rho_3 = \sum_k r_k|\xi_k\rangle\langle\xi_k|$. We have the fol-

lowing equalities.

$$\text{tr}(\rho_1 \otimes \rho_2 \otimes \rho_3 \cdot S_1 S_2) = \sum_{jlk} p_j q_l r_k \langle \psi_l | \phi_j \rangle \langle \xi_k | \psi_l \rangle \langle \phi_j | \xi_k \rangle, \tag{A.62}$$

$$\text{tr}(\rho_2 \otimes \rho_1 \otimes \rho_3 \cdot S_1 S_2) = \sum_{jlk} p_j q_l r_k \langle \phi_j | \psi_l \rangle \langle \xi_k | \phi_j \rangle \langle \psi_l | \xi_k \rangle, \tag{A.63}$$

$$\text{tr}(\rho_3 \otimes \rho_2 \otimes \rho_1 \cdot S_1 S_2) = \sum_{jlk} p_j q_l r_k \langle \psi_l | \xi_k \rangle \langle \phi_j | \psi_l \rangle \langle \xi_k | \phi_j \rangle. \tag{A.64}$$

Comparing Eqs. (A.62)-(A.64), the equality in Eq. (A.57) is proved. ■

A.4 Supplementary discussion for optimization

To simplify the notations, let L_1 denote $\text{tr}(H\rho(\boldsymbol{\theta}))$, L_2 denote $2\beta^{-1}\text{tr}(\rho(\boldsymbol{\theta})^2)$, and L_3 denote $-\frac{\beta^{-1}}{2}\text{tr}(\rho(\boldsymbol{\theta})^3)$. Using these notations, our loss function can be rewritten as $\mathcal{F}_2(\boldsymbol{\theta}) = L_1 + L_2 + L_3 - \frac{3\beta^{-1}}{2}$, and the gradient of $\mathcal{F}_2(\boldsymbol{\theta})$ can be rewritten as follow:

$$\nabla_{\boldsymbol{\theta}} \mathcal{F}_2(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} L_1 + \nabla_{\boldsymbol{\theta}} L_2 + \nabla_{\boldsymbol{\theta}} L_3. \tag{A.65}$$

Therefore, the gradient of $\mathcal{F}_K(\boldsymbol{\theta})$ can be estimated via computing the gradients of L_j, $j = 1,2,3$. Specifically, $\nabla_{\boldsymbol{\theta}} L_j$, $j = 2,3$, can be computed using the destructive SWAP test and higher order state overlap estimation, respectively. As for $\nabla_{\boldsymbol{\theta}} L_1$, it can be estimated by measurement directly.

Next, we show that the gradients of L_j's can be computed by shifting the parameters $\boldsymbol{\theta}$ of the circuit. The partial derivatives of each L_j have the following forms,

$$\frac{\partial L_1}{\partial \theta_m} = \frac{\partial}{\partial \theta_m} \text{tr}(U_N ... U_1 |0\rangle\langle 0| U_1^\dagger ... U_N^\dagger \cdot (I \otimes H)), \tag{A.66}$$

$$\frac{\partial L_2}{\partial \theta_m} = 2\beta^{-1} \frac{\partial}{\partial \theta_m} \text{tr}((U_N ... U_1 |0\rangle\langle 0| U_1^\dagger ... U_N^\dagger)^{\otimes 2} \cdot W_1), \tag{A.67}$$

$$\frac{\partial L_3}{\partial \theta_m} = -\frac{\beta^{-1}}{2} \frac{\partial}{\partial \theta_m} \text{tr}((U_N ... U_1 |0\rangle\langle 0| U_1^\dagger ... U_N^\dagger)^{\otimes 3} \cdot W_2), \tag{A.68}$$

where W_1 denotes the operator $SWAP_{B_2B_3} \otimes I_{A_2A_3}$, W_2 denotes the operator $(SWAP_{B_4B_5} \otimes I_{A_4A_5A_6B_6}) \cdot (SWAP_{B_5B_6} \otimes I_{A_4B_4A_5A_6})$, and the operator $SWAP_{B_jB_l}$ is a swap operator acting on registers B_j and B_l.

To further simplify notations, we absorb all gates before and after U_m into the density operator and measurement operator, respectively. To be more specific, let $\psi_{A_lB_l}$ denote the density operator $U_{m-1} ... U_1 |00\rangle\langle 00|_{A_lB_l} U_1^\dagger ... U_{m-1}^\dagger$ in register A_lB_l, for $l = 1, ..., 6$. And

we define observable operators K, O, G as follows

$$K = U_{m+1}^{\dagger}...U_N^{\dagger}(I_{A_1} \otimes H_{B_1})U_N...U_{m+1}. \tag{A.69}$$

$$O = (U_{m+1}^{\dagger}...U_N^{\dagger})^{\otimes 2} W_1 (U_N...U_{m+1})^{\otimes 2}, \tag{A.70}$$

$$G = (U_{m+1}...U_N^{\dagger})^{\otimes 3} W_2 (U_N...U_{m+1})^{\otimes 3}. \tag{A.71}$$

Then partial derivatives in Eqs. (A.66)-(A.68) can be rewritten as

$$\frac{\partial L_1}{\partial \theta_m} = \frac{\partial}{\partial \theta_m} \operatorname{tr}(U_m(\theta_m)\psi_{A_1B_1}U_m^{\dagger}(\theta_m)\cdot K), \tag{A.72}$$

$$\frac{\partial L_2}{\partial \theta_m} = 2\beta^{-1}\frac{\partial}{\partial \theta_m} \operatorname{tr}(U_m\psi_{A_2B_2}U_m^{\dagger}(\theta_m)\otimes U_m\psi_{A_3B_3}U_m^{\dagger}(\theta_m)\cdot O), \tag{A.73}$$

$$\frac{\partial L_3}{\partial \theta_m} = -\frac{\beta^{-1}}{2}\frac{\partial}{\partial \theta_m} \operatorname{tr}(U_m\psi_{A_4B_4}U_m^{\dagger}(\theta_m)\otimes U_m\psi_{A_5B_5}U_m^{\dagger}(\theta_m)\otimes U_m\psi_{A_6B_6}U_m^{\dagger}(\theta_m)\cdot G). \tag{A.74}$$

Now, we derive the analytical forms of the derivatives of each L_j, $j = 1,2,3$. Notice that the trainable unitary $U(\boldsymbol{\theta})$ is a sequence of unitaries $U_m(\theta_m)$ and each unitary $U_m(\theta_m) = e^{-i\theta_m H_m/2}$. The partial derivative of $U(\boldsymbol{\theta})$ can be explicitly given as follows,

$$\frac{\partial U(\boldsymbol{\theta})}{\partial \theta_m} = U_N(\theta_N)...\frac{\partial U_m(\theta_m)}{\partial \theta_m}...U_1(\theta_1), \tag{A.75}$$

$$= -\frac{i}{2}U_N(\theta_N)...H_mU_m...U_1(\theta_1). \tag{A.76}$$

Using the expression of $\partial_{\theta_m}U(\boldsymbol{\theta})$ in Eq. (A.76), and some facts, including an identity $i[H_m, M] = U_m(-\pi/2)MU_m^{\dagger}(-\pi/2) - U_m(\pi/2)MU_m^{\dagger}(\pi/2)$, which holds true for arbitrary matrix M, the symmetry of the operator O, and an equality $\partial_{\theta_m}\operatorname{tr}(\rho(\boldsymbol{\theta})^3) = 3\partial_{\theta_{m1}}\operatorname{tr}(\rho_1(\boldsymbol{\theta})\otimes \rho_2(\boldsymbol{\theta})\otimes\rho_3(\boldsymbol{\theta})\cdot S_1S_2)$ (c.f. Lemma S13 in Appendix A.3), where $S_1 = \mathrm{SW}AP_{12}\otimes I_3$ and $S_2 = I_1\otimes \mathrm{SW}AP_{23}$, the gradients of each L_j can be estimated using the following formulas,

$$\frac{\partial L_1}{\partial \theta_m} = \frac{1}{2}(\langle K\rangle_{\theta_m+\frac{\pi}{2}} - \langle K\rangle_{\theta_m-\frac{\pi}{2}}), \tag{A.77}$$

$$\frac{\partial L_2}{\partial \theta_m} = 2\beta^{-1}\left(\langle O\rangle_{\theta_m+\frac{\pi}{2},\theta_m} - \langle O\rangle_{\theta_m-\frac{\pi}{2},\theta_m}\right), \tag{A.78}$$

$$\frac{\partial L_3}{\partial \theta_m} = -\frac{3\beta^{-1}}{4}\left(\langle G\rangle_{\theta_m+\frac{\pi}{2},\theta_m,\theta_m} - \langle G\rangle_{\theta_m-\frac{\pi}{2},\theta_m,\theta_m}\right), \tag{A.79}$$

where the notation $\langle X\rangle_{a,b}$ is defined below,

$$\langle K\rangle_{\theta_\alpha} = \operatorname{tr}\left(U_\alpha\psi_{A_1B_1}U_\alpha^{\dagger}\cdot K\right), \tag{A.80}$$

$$\langle O\rangle_{\theta_\alpha,\theta_\beta} = \operatorname{tr}\left(U_\alpha\psi_{A_2B_2}U_\alpha^{\dagger}\otimes U_\beta\psi_{A_3B_3}U_\beta^{\dagger}\cdot O\right), \tag{A.81}$$

$$\langle G\rangle_{\theta_\alpha,\theta_\beta,\theta_\gamma} = \operatorname{tr}\left(U_\alpha\psi_{A_4B_4}U_\alpha^{\dagger}\otimes U_\beta\psi_{A_5B_5}U_\beta^{\dagger}\otimes U_\gamma\psi_{A_6B_6}U_\gamma^{\dagger}\cdot G\right). \tag{A.82}$$

A.5 Gate decomposition

This section decomposes the quantum gates in Figure 5.5 into primitive single/two-qubit gates. We primarily consider decomposing controlled gates c-s$elect(\mathcal{S})$, and anti-controlled gates oc-R_2.

Circuit of oc-R_2. Here, we let $R_2 = R_y(\theta_2)$. By the relation that $XR_y(-\theta)X = R_y(\theta)$, we can decompose oc-$R_y(\theta_2)$ as shown in Figure A.1.

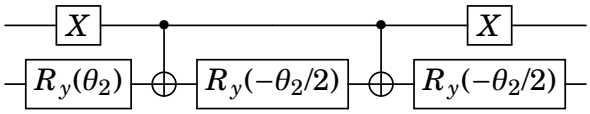


Figure A.1: Quantum circuit for anti-controlled rotation oc-R_2.

Circuit of $c-select(\mathcal{S})$**.** Notice that s$elect(\mathcal{S})$ is a product of two operations: $P_1 = (|0\rangle\langle 0| + (-isgn(t)|1\rangle\langle 1|) \otimes I_{2n}$ and $P_2 = |0\rangle\langle 0| \otimes I_{2n} + |1\rangle\langle 1| \otimes \mathcal{S}$. Hence, the c-s$elect(\mathcal{S})$ is to separately apply c-P_1/P_2. One key component of c-P_1 is the controlled phase gate c-S, which consists of T gate, CNOT, and R_z. The decomposition is depicted in Figure A.2.

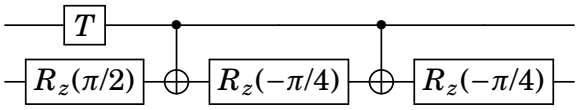


Figure A.2: Quantum circuit for controlled phase gate c-S.

As for the c-P_2 gate, we first append one more measure register $|0\rangle$ and then use Toffoli and CNOT gates. Note that the decomposition of Toffoli gate can be found in [111]. In consequence, our circuit is composed of primitive single/two-qubit gates.

Ultimately, combining the circuits of c-P_1/P_2 leads to the circuit for c-s$elect(\mathcal{S})$, which is depicted in Figure A.3.

A.6 Barren plateaus

Barren plateaus phenomena [105] mean that the search direction is exponentially small in the landscape of the loss function. In the next, we explain how gradients vanish exponentially fast with scaling in the number of qubits.

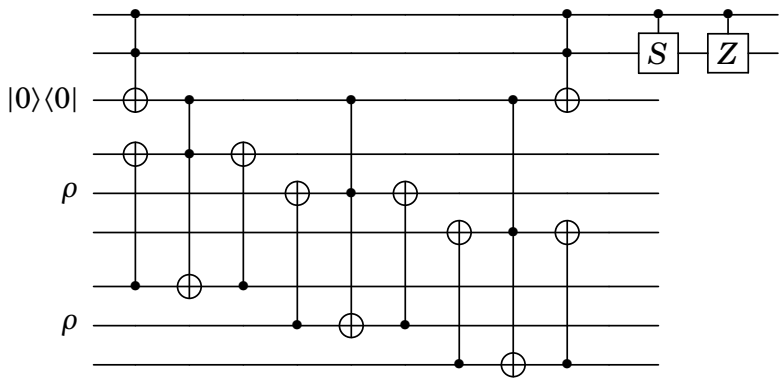


Figure A.3: Quantum circuit for implementing controlled s$elect(\mathbb{S})$. Here we take three-qubit state ρ as example. The circuit appends one qubit $|0\rangle$. The decomposition of the c-S is given in Figure A.2. Particularly, the c-Z gate is applied only when $t > 0$.

Note that for a fixed circuit template, the randomness nature arises from randomly choosing parameters. We, therefore, focus on computing the expectation and the variance of the gradient.

Expectation deduction. We focus on unitaries $U_{1:k}$ and $U_{k+1:L}$ and their randomness, respectively.

For remark, the choices of $U_{1:k}$ and $U_{k+1:L}$ are independent of each other. Hence, we write the probability distribution of $U(\boldsymbol{\theta})$ in terms of the independence.

$$\mu(U) = \mu(U_{1:k}) \times \mu(U_{k+1:L}). \tag{A.83}$$

Here μ denotes the Haar measure.

Then the expectation of loss $L(\hat{\boldsymbol{\theta}}_{k,+})$ can be written as

(A.84)

$$\mathbf{E}[L(\hat{\boldsymbol{\theta}}_{k,+})] = \int_D L(\hat{\boldsymbol{\theta}}_{k,+}) \mathrm{d}\mu(U)$$

(A.85)

$$= \mathrm{tr}\left[U_k^\dagger(\frac{\pi}{2}) \cdot \int_{D_{1:k}} U_{k+1:L}^\dagger H U_{k+1:L} \mathrm{d}\mu(U_{k+1:L}) \cdot U_k(\frac{\pi}{2}) \cdot \int_{D_{k+1:L}} U_{1:k} \left|0^n\right\rangle\!\left\langle 0^n\right| U_{1:k}^\dagger \mathrm{d}\mu(U_{1:k}) \right].$$

By the left- and right-invariant property of the Haar measure, we have

$$U_k^\dagger(\frac{\pi}{2}) \cdot \int_{D_{1:k}} U_{k+1:L}^\dagger H U_{k+1:L} \mathrm{d}\mu(U_{k+1:L}) \cdot U_k(\frac{\pi}{2}) = \int_{D_{1:k}} U_{k+1:L}^\dagger H U_{k+1:L} \mathrm{d}\mu(U_{k+1:L}). \tag{A.86}$$

Consequently, the expectation is simplified as

(A.87)
$$\mathbf{E}[L(\hat{\boldsymbol{\theta}}_{k,+})] = \mathrm{tr}\left[\int_{D_{1:k}} U_{k+1:L}^{\dagger} H U_{k+1:L} \mathrm{d}\mu(U_{k+1:L}) \cdot \int_{D_{k+1:L}} U_{1:k} \left|0^n\right\rangle\!\left\langle 0^n\right| U_{1:k}^{\dagger} \mathrm{d}\mu(U_{1:k})\right].$$

We can readily derive a similar result for $\mathbf{E}[L(\hat{\boldsymbol{\theta}}_{k,-})]$, i.e.,

(A.88)
$$\mathbf{E}[L(\hat{\boldsymbol{\theta}}_{k,-})] = \mathrm{tr}\left[\int_{D_{1:k}} U_{k+1:L}^{\dagger} H U_{k+1:L} \mathrm{d}\mu(U_{k+1:L}) \cdot \int_{D_{k+1:L}} U_{1:k} \left|0^n\right\rangle\!\left\langle 0^n\right| U_{1:k}^{\dagger} \mathrm{d}\mu(U_{1:k})\right].$$

Ultimately, the expectation of the gradient is zero, as shown above.

Remark 10 In the above deduction, we have assumed that quantum circuits $U_{k+1:L}$ can reach the invariant property of the Haar measure. We can also assume $U_{1:k}$ satisfy the invariant property as well. Particularly, the invariant property is shown below.

(A.89)
$$\int_D \mathrm{d}\mu(U) f(U) = \int \mathrm{d}\mu(U) f(VU) = \int \mathrm{d}\mu(U) f(UV),$$

for any $f(U)$ and $V \in D$, where the integration domain is implicitly denoted by D. However, it would cost exponentially many resources for quantum circuits to satisfy this property. To overcome this issue, we can assume the quantum circuits satisfy t-design, where only modest polynomial resources are required for a restricted class of $f(U)$.

Variance deduction. Now, we consider computing the variance of the gradient.

(A.90)
$$\mathbf{Var}\left[\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_k}\right] = \frac{1}{4}\mathbf{E}\left[\left(L(\hat{\boldsymbol{\theta}}_{k,+}) - L(\hat{\boldsymbol{\theta}}_{k,-})\right)^2\right] = \frac{1}{4}\mathbf{E}\left[L^2(\hat{\boldsymbol{\theta}}_{k,+}) - 2L(\hat{\boldsymbol{\theta}}_{k,+})L(\hat{\boldsymbol{\theta}}_{k,-}) + L^2(\hat{\boldsymbol{\theta}}_{k,-})\right].$$

Without loss of generality, we focus on computing $\mathbf{E}[L(\hat{\boldsymbol{\theta}}_{k,+})L(\hat{\boldsymbol{\theta}}_{k,-})]$.

Phase 1. We rewrite $L(\hat{\boldsymbol{\theta}}_{k,+})L(\hat{\boldsymbol{\theta}}_{k,-})$ and assume $U_{1:k}$ satisfies 2-design.

(A.91)
$$L(\hat{\boldsymbol{\theta}}_{k,+})L(\hat{\boldsymbol{\theta}}_{k,-}) = \langle 0^n | U^{\dagger}(\hat{\boldsymbol{\theta}}_{k,+}) H U(\hat{\boldsymbol{\theta}}_{k,+}) | 0^n \rangle \cdot \langle 0^n | U^{\dagger}(\hat{\boldsymbol{\theta}}_{k,-}) H U(\hat{\boldsymbol{\theta}}_{k,-}) | 0^n \rangle$$

(A.92)
$$= \mathrm{tr}\left[H U(\hat{\boldsymbol{\theta}}_{k,+}) \left|0^n\right\rangle\!\left\langle 0^n\right| U^{\dagger}(\hat{\boldsymbol{\theta}}_{k,-}) H U(\hat{\boldsymbol{\theta}}_{k,-}) \left|0^n\right\rangle\!\left\langle 0^n\right| U^{\dagger}(\hat{\boldsymbol{\theta}}_{k,+})\right]$$

(A.93)
$$= \mathrm{tr}\left[U_{1:k} A U_{1:k}^{\dagger} B U_{1:k} C U_{1:k}^{\dagger} D\right],$$

where

(A.94)
$$A = \left|0^n\right\rangle\!\left\langle 0^n\right|, \quad B = U_k^{\dagger}(\frac{-\pi}{2}) U_{k+1:L}^{\dagger} H U_{k+1:L} U_k(\frac{-\pi}{2}),$$

(A.95)
$$C = \left|0^n\right\rangle\!\left\langle 0^n\right|, \quad D = U_k^{\dagger}(\frac{\pi}{2}) U_{k+1:L}^{\dagger} H U_{k+1:L} U_k(\frac{\pi}{2}).$$

Then the expectation $\mathbf{E}[L(\hat{\boldsymbol{\theta}}_{k,+})L(\hat{\boldsymbol{\theta}}_{k,-})]$ is reduced to

$$\int \mathrm{d}\mu(U_{1:k}) \times \int \mathrm{d}\mu(U_{k+1:L}) \mathrm{tr}\left[U_{1:k} A U_{1:k}^\dagger B U_{1:k} C U_{1:k}^\dagger D\right]. \tag{A.96}$$

According to Lemma 2 of [33], if the circuits $U_{1:k}$ satisfy 2-design, the integration over the domain of $U_{1:k}$ is

$$\frac{\mathrm{tr}(A)\mathrm{tr}(C)\mathrm{tr}(BD) + \mathrm{tr}(AC)\mathrm{tr}(B)\mathrm{tr}(D)}{4^n - 1} - \frac{\mathrm{tr}(AC)\mathrm{tr}(BD) + \mathrm{tr}(A)\mathrm{tr}(B)\mathrm{tr}(C)\mathrm{tr}(D)}{2^n(4^n-1)}. \tag{A.97}$$

As a result, the expectation is

(A.98)

$$\begin{aligned}\mathbf{E}\left[L(\hat{\boldsymbol{\theta}}_{k,+})L(\hat{\boldsymbol{\theta}}_{k,-})\right] &= \int \mathrm{d}\mu(U_{k+1:L}) \left[\mathrm{tr}(U_{k+1:L}^\dagger H U_{k+1:L} H_k U_{k+1:L}^\dagger H U_{k+1:L} H_k) + [\mathrm{tr}(H)]^2\right] \\ &\quad \times \left(\frac{1}{4^n-1} - \frac{1}{2^n(4^n-1)}\right) \\ &= \frac{\int \mathrm{d}\mu(U_{k+1:L}) \mathrm{tr}(U_{k+1:L}^\dagger H U_{k+1:L} H_k U_{k+1:L}^\dagger H U_{k+1:L} H_k) + [\mathrm{tr}(H)]^2}{2^n(2^n+1)}. \end{aligned} \tag{A.99}$$

Moreover, the expectations of rests are given below,

(A.100)

$$\mathbf{E}\left[L(\hat{\boldsymbol{\theta}}_{k,+})L(\hat{\boldsymbol{\theta}}_{k,+})\right] = \mathbf{E}\left[L(\hat{\boldsymbol{\theta}}_{k,-})L(\hat{\boldsymbol{\theta}}_{k,-})\right] = \frac{\mathrm{tr}(H^2) + [\mathrm{tr}(H)]^2}{4^n-1} - \frac{\mathrm{tr}(H^2) + [\mathrm{tr}(H)]^2}{2^n(4^n-1)} = \frac{\mathrm{tr}(H^2) + [\mathrm{tr}(H)]^2}{2^n(2^n+1)}.$$

Eventually, the variance of the gradient is computed as follows.

$$\mathbf{Var}\left[\frac{\partial}{\partial \theta_k} L(\boldsymbol{\theta}_k)\right] =$$

(A.101)

$$\frac{1}{2}\left[\frac{\mathrm{tr}(H^2) + [\mathrm{tr}(H)]^2}{2^n(2^n+1)} - \frac{\int \mathrm{d}\mu(U_{k+1:L}) \mathrm{tr}(U_{k+1:L}^\dagger H U_{k+1:L} H_k U_{k+1:L}^\dagger H U_{k+1:L} H_k) + [\mathrm{tr}(H)]^2}{2^n(2^n+1)}\right]$$

(A.102)

$$= \frac{\mathrm{tr}(H^2) - \int \mathrm{d}\mu(U_{k+1:L}) \mathrm{tr}(U_{k+1:L}^\dagger H U_{k+1:L} H_k U_{k+1:L}^\dagger H U_{k+1:L} H_k)}{2^{n+1}(2^n+1)}.$$

Phase 2. We also assume the circuits $U_{k+1:L}$ satisfy 2-design.

$$\int \mathrm{d}\mu(U_{k+1:L}) \mathrm{tr}(U_{k+1:L} H_k U_{k+1:L}^\dagger H U_{k+1:L} H_k U_{k+1:L}^\dagger H) \tag{A.103}$$

$$= \frac{\mathrm{tr}(H_k)\mathrm{tr}(H_k)\mathrm{tr}(H^2) + \mathrm{tr}(H_k^2)[\mathrm{tr}(H)]^2}{4^n-1} - \frac{\mathrm{tr}(H_k^2)\mathrm{tr}(H^2) + [\mathrm{tr}(H_k)\mathrm{tr}(H)]^2}{2^n(4^n-1)} \tag{A.104}$$

$$= \frac{2^n[\mathrm{tr}(H)]^2}{4^n-1} - \frac{2^n \mathrm{tr}(H^2)}{2^n(4^n-1)} = \frac{2^n[\mathrm{tr}(H)]^2 - \mathrm{tr}(H^2)}{4^n-1}. \tag{A.105}$$

In this case, the variance is

$$\mathbf{Var}\left[\frac{\partial}{\partial\theta_k}L(\boldsymbol{\theta}_k)\right] = \frac{1}{2^{n+1}(2^n+1)}\left[\mathrm{tr}(H^2) - \frac{2^n[\mathrm{tr}(H)]^2 - \mathrm{tr}(H^2)}{4^n-1}\right] \tag{A.106}$$

$$= \frac{2^n\,\mathrm{tr}(H^2) - \mathrm{tr}(H)^2}{2(2^n+1)(4^n-1)}. \tag{A.107}$$

For many-body Hamiltonian, $tr(H) = 0$ since the Hamiltonians usually only consist of Pauli strings, which are traceless.

Remark 11 Let $H = \sum_l \mu_l H_l$, where H_l are different Pauli strings. Then the trace $\mathrm{tr}(H^2)$ has a bound.

$$\mathrm{tr}(H^2) = \mathrm{tr}\left(\sum_l \mu_l^2 H_l^2 + \sum_{k,l:k\neq l} \mu_l\mu_k H_l H_k\right) = 2^n \sum_l \mu_l^2. \tag{A.108}$$

As a result, the variance bound is given by $\frac{4^n\sum_l \mu_l^2}{2(2^n+1)(4^n-1)}$.

BIBLIOGRAPHY

[1] D. S. ABRAMS AND S. LLOYD, *Quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors*, Phys. Rev. Lett., 83 (1999), pp. 5162–5165.

[2] J. ACHARYA, I. ISSA, N. V. SHENDE, AND A. B. WAGNER, *Measuring quantum entropy*, 2019 IEEE International Symposium on Information Theory (ISIT), (2019), pp. 3012–3016.

[3] D. AHARONOV, I. ARAD, AND T. VIDICK, *Guest column: the quantum pcp conjecture*, Acm sigact news, 44 (2013), pp. 47–79.

[4] D. AHARONOV, W. VAN DAM, J. KEMPE, Z. LANDAU, S. LLOYD, AND O. REGEV, *Adiabatic Quantum Computation Is Equivalent to Standard Quantum Computation*, SIAM Review, 50 (2008), pp. 755–787.

[5] A. ANSHU, S. ARUNACHALAM, T. KUWAHARA, AND M. SOLEIMANIFAR, *Sample-efficient learning of quantum many-body systems*, (2020).

[6] A. ARRASMITH, L. CINCIO, R. D. SOMMA, AND P. J. COLES, *Operator Sampling for Shot-frugal Optimization in Variational Algorithms*, arXiv preprint arXiv:2004.06252, (2020), pp. 1–11.

[7] F. ARUTE, K. ARYA, R. BABBUSH, D. BACON, J. C. BARDIN, R. BARENDS, R. BISWAS, S. BOIXO, F. G. BRANDAO, D. A. BUELL, B. BURKETT, Y. CHEN, Z. CHEN, B. CHIARO, R. COLLINS, W. COURTNEY, A. DUNSWORTH, E. FARHI, B. FOXEN, A. FOWLER, C. GIDNEY, M. GIUSTINA, R. GRAFF, K. GUERIN, S. HABEGGER, M. P. HARRIGAN, M. J. HARTMANN, A. HO, M. HOFFMANN, T. HUANG, T. S. HUMBLE, S. V. ISAKOV, E. JEFFREY, Z. JIANG, D. KAFRI, K. KECHEDZHI, J. KELLY, P. V. KLIMOV, S. KNYSH, A. KOROTKOV, F. KOSTRITSA, D. LANDHUIS, M. LINDMARK, E. LUCERO, D. LYAKH, S. MANDRÀ, J. R. MCCLEAN, M. MCEWEN, A. MEGRANT, X. MI, K. MICHIELSEN,

M. MOHSENI, J. MUTUS, O. NAAMAN, M. NEELEY, C. NEILL, M. Y. NIU, E. OSTBY, A. PETUKHOV, J. C. PLATT, C. QUINTANA, E. G. RIEFFEL, P. ROUSHAN, N. C. RUBIN, D. SANK, K. J. SATZINGER, V. SMELYANSKIY, K. J. SUNG, M. D. TREVITHICK, A. VAINSENCHER, B. VILLALONGA, T. WHITE, Z. J. YAO, P. YEH, A. ZALCMAN, H. NEVEN, AND J. M. MARTINIS, *Quantum supremacy using a programmable superconducting processor*, Nature, 574 (2019), pp. 505–510.

[8] A. ASPURU-GUZIK, A. D. DUTOI, P. J. LOVE, AND M. HEAD-GORDON, *Simulated quantum computation of molecular energies*, Science, 309 (2005), pp. 1704–1707.

[9] S. AXLER, *Linear algebra done right*, Springer Science & Business Media, 1997.

[10] B. BEIN, *Entropy*, Best Practice & Research Clinical Anaesthesiology, 20 (2006), pp. 101–109.

[11] J. D. BEKENSTEIN, *Black holes and entropy*, in JACOB BEKENSTEIN: The Conservative Revolutionary, World Scientific, 2020, pp. 307–320.

[12] M. BENEDETTI, D. GARCIA-PINTOS, O. PERDOMO, V. LEYTON-ORTEGA, Y. NAM, AND A. PERDOMO-ORTIZ, *A generative modeling approach for benchmarking and training shallow quantum circuits*, npj Quantum Information, 5 (2019), p. 45.

[13] M. BENEDETTI, E. GRANT, L. WOSSNIG, AND S. SEVERINI, *Adversarial quantum circuit learning for pure state approximation*, New Journal of Physics, 21 (2019), p. 043023.

[14] C. H. BENNETT, H. J. BERNSTEIN, S. POPESCU, AND B. SCHUMACHER, *Concentrating partial entanglement by local operations*, Physical Review A, 53 (1996), pp. 2046–2052.

[15] C. H. BENNETT AND P. W. SHOR, *Quantum information theory*, IEEE transactions on information theory, 44 (1998), pp. 2724–2742.

[16] D. W. BERRY, A. M. CHILDS, AND R. KOTHARI, *Hamiltonian Simulation with Nearly Optimal Dependence on all Parameters*, in 2015 IEEE 56th Annual Symposium on Foundations of Computer Science, vol. 2015-Decem, IEEE, oct 2015, pp. 792–809.

[17] K. BHARTI, A. CERVERA-LIERTA, T. H. KYAW, T. HAUG, S. ALPERIN-LEA, A. ANAND, M. DEGROOTE, H. HEIMONEN, J. S. KOTTMANN, T. MENKE, W.-K. MOK, S. SIM, L.-C. KWEK, AND A. ASPURU-GUZIK, *Noisy intermediate-scale quantum (NISQ) algorithms*, arXiv:2101.08448, (2021), pp. 1–82.

[18] J. BIAMONTE, P. WITTEK, N. PANCOTTI, P. REBENTROST, N. WIEBE, AND S. LLOYD, *Quantum machine learning*, Nature, 549 (2017), pp. 195–202.

[19] E. BILGIN AND S. BOIXO, *Preparing Thermal States of Quantum Systems by Dimension Reduction*, Physical Review Letters, 105 (2010), p. 170405.

[20] M. BILKIS, M. CEREZO, G. VERDON, P. J. COLES, AND L. CINCIO, *A semi-agnostic ansatz with variable structure for quantum machine learning*, arXiv preprint arXiv:2103.06712, (2021).

[21] X. BONET-MONROIG, R. BABBUSH, AND T. E. O'BRIEN, *Nearly Optimal Measurement Scheduling for Partial Tomography of Quantum States*, arXiv preprint arXiv:1908.05628, (2019), pp. 1–9.

[22] S. BOYD, L. VANDENBERGHE, AND L. FAYBUSOVICH, *Convex optimization*, IEEE Transactions on Automatic Control, 11 (2006), p. 1859.

[23] F. G. BRANDÃO AND M. J. KASTORYANO, *Finite correlation length implies efficient preparation of quantum thermal states*, Communications in Mathematical Physics, 365 (2019), pp. 1–16.

[24] F. G. BRANDAO AND K. M. SVORE, *Quantum Speed-Ups for Solving Semidefinite Programs*, in 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS), IEEE, oct 2017, pp. 415–426.

[25] C. BRAVO-PRIETO, R. LAROSE, M. CEREZO, Y. SUBASI, L. CINCIO, AND P. J. COLES, *Variational Quantum Linear Solver: A Hybrid Algorithm for Linear Systems*, (2019), pp. 1–16.

[26] H. BUHRMAN, R. CLEVE, J. WATROUS, AND R. DE WOLF, *Quantum Fingerprinting*, Physical Review Letters, 87 (2001), p. 167902.

[27] C. BUTUCEA, M. GUTA, AND T. KYPRAIOS, *Spectral thresholding quantum tomography for low rank states*, New Journal of Physics, 17 (2015), p. 113050.

[28] E. CARLEN, *Trace inequalities and quantum entropy: an introductory course*, Entropy and the quantum, 529 (2010), pp. 73–140.

[29] M. C. CARO AND I. DATTA, *Pseudo-dimension of quantum circuits*, Quantum Machine Intelligence, 2 (2020), pp. 1–14.

[30] M. C. CARO, E. GIL-FUSTER, J. J. MEYER, J. EISERT, AND R. SWEKE, *Encoding-dependent generalization bounds for parametrized quantum circuits*, arXiv preprint arXiv:2106.03880, (2021).

[31] M. CEREZO, A. POREMBA, L. CINCIO, AND P. J. COLES, *Variational Quantum Fidelity Estimation*, Quantum, 4 (2020), p. 248.

[32] M. CEREZO, K. SHARMA, A. ARRASMITH, AND P. J. COLES, *Variational Quantum State Eigensolver*, (2020), pp. 1–14.

[33] M. CEREZO, A. SONE, T. VOLKOFF, L. CINCIO, AND P. J. COLES, *Cost function dependent barren plateaus in shallow parametrized quantum circuits*, Nature Communications, 12 (2021), p. 1791.

[34] C.-C. CHEN, M. WATABE, K. SHIBA, M. SOGABE, K. SAKAMOTO, AND T. SOGABE, *On the expressibility and overfitting of quantum circuit learning*, ACM Transactions on Quantum Computing, 2 (2021), pp. 1–24.

[35] A. M. CHILDS, D. MASLOV, Y. NAM, N. J. ROSS, AND Y. SU, *Toward the first quantum simulation with quantum speedup*, Proceedings of the National Academy of Sciences, 115 (2018), pp. 9456–9461.

[36] A. N. CHOWDHURY, G. H. LOW, AND N. WIEBE, *A Variational Quantum Algorithm for Preparing Quantum Gibbs States*, arXiv, (2020), pp. 1–13.

[37] L. CINCIO, Y. SUBASI, A. T. SORNBORGER, AND P. J. COLES, *Learning the quantum algorithm for state overlap*, New Journal of Physics, 20 (2018), p. 113022.

[38] B. COMMEAU, M. CEREZO, Z. HOLMES, L. CINCIO, P. J. COLES, AND A. SORNBORGER, *Variational Hamiltonian Diagonalization for Dynamical Quantum Simulation*, arXiv preprint arXiv:2009.02559, (2020), pp. 1–12.

[39] J. COTLER AND F. WILCZEK, *Quantum Overlapping Tomography*, Physical Review Letters, 124 (2020).

[40] M. DOBŠÍČEK, G. JOHANSSON, V. SHUMEIKO, AND G. WENDIN, *Arbitrary accuracy iterative quantum phase estimation algorithm using a single ancillary qubit: A two-qubit benchmark*, Physical Review A, 76 (2007), p. 030306.

[41] X. DONG, *The gravity dual of rényi entropy*, Nature communications, 7 (2016), pp. 1–6.

[42] Y. DU, T. HUANG, S. YOU, M.-H. HSIEH, AND D. TAO, *Quantum circuit architecture search: error mitigation and trainability enhancement for variational quantum solvers*, arXiv preprint arXiv:2010.10217, (2020).

[43] Y. DU AND D. TAO, *On exploring practical potentials of quantum auto-encoder with advantages*, arXiv preprint arXiv:2106.15432, (2021), pp. 1–25.

[44] Y. DU, Z. TU, X. YUAN, AND D. TAO, *An efficient measure for the expressivity of variational quantum algorithms*, arXiv preprint arXiv:2104.09961, (2021).

[45] D. J. EGGER, M. WERNINGHAUS, M. GANZHORN, G. SALIS, A. FUHRER, P. MUELLER, AND S. FILIPP, *Pulsed reset protocol for fixed-frequency superconducting qubits*, Physical Review Applied, 10 (2018), p. 044030.

[46] A. K. EKERT, *Quantum cryptography based on Bell's theorem*, Physical Review Letters, 67 (1991), pp. 661–663.

[47] A. K. EKERT, C. M. ALVES, D. K. OI, M. HORODECKI, P. HORODECKI, AND L. C. KWEK, *Direct estimations of linear and nonlinear functionals of a quantum state*, Physical review letters, 88 (2002), p. 217901.

[48] A. K. EKERT, C. M. ALVES, D. K. L. OI, M. HORODECKI, P. HORODECKI, AND L. C. KWEK, *Direct Estimations of Linear and Nonlinear Functionals of a Quantum State*, Physical Review Letters, 88 (2002), p. 217901.

[49] S. ENDO, Z. CAI, S. C. BENJAMIN, AND X. YUAN, *Hybrid Quantum-Classical Algorithms and Quantum Error Mitigation*, Journal of the Physical Society of Japan, 90 (2021), p. 032001.

[50] E. FARHI, J. GOLDSTONE, AND S. GUTMANN, *A quantum approximate optimization algorithm*, arXiv preprint arXiv:1411.4028, (2014).

[51] E. FARHI AND H. NEVEN, *Classification with Quantum Neural Networks on Near Term Processors*, arXiv:1802.06002, (2018), pp. 1–21.

[52] M. FOSS-FEIG, D. HAYES, J. M. DREILING, C. FIGGATT, J. P. GAEBLER, S. A. MOSES, J. M. PINO, AND A. C. POTTER, *Holographic quantum algorithms for simulating correlated spin systems*, Physical Review Research, 3 (2021), p. 033002.

[53] J. C. GARCIA-ESCARTIN AND P. CHAMORRO-POSADA, *swap test and Hong-Ou-Mandel effect are equivalent*, Physical Review A, 87 (2013), p. 052330.

[54] K. GEERLINGS, Z. LEGHTAS, I. M. POP, S. SHANKAR, L. FRUNZIO, R. J. SCHOELKOPF, M. MIRRAHIMI, AND M. H. DEVORET, *Demonstrating a driven reset protocol for a superconducting qubit*, Physical review letters, 110 (2013), p. 120501.

[55] A. GHEORGHIU AND M. J. HOBAN, *Estimating the entropy of shallow circuit outputs is hard*, (2020), pp. 1–31.

[56] A. GILYÉN AND T. LI, *Distributional property testing in a quantum world*, Leibniz International Proceedings in Informatics, LIPIcs, 151 (2019), pp. 1–18.

[57] L. A. GOLDBERG AND M. JERRUM, *Inapproximability of the tutte polynomial*, Information and Computation, 206 (2008), pp. 908–929.

[58] D. GOTTESMAN AND I. CHUANG, *Quantum Digital Signatures*, arXiv:quant-ph/0105032, (2001).

[59] L. C. GOVIA AND F. K. WILHELM, *Unitary-feedback-improved qubit initialization in the dispersive regime*, Physical Review Applied, 4 (2015), p. 054001.

[60] C. E. GRANADE, C. FERRIE, N. WIEBE, AND D. G. CORY, *Robust online Hamiltonian learning*, New Journal of Physics, 14 (2012), p. 103013.

[61] E. GRANT, L. WOSSNIG, M. OSTASZEWSKI, AND M. BENEDETTI, *An initialization strategy for addressing barren plateaus in parametrized quantum circuits*, Quantum, 3 (2019), p. 214.

[62] R. B. GRIFFITHS AND C.-S. NIU, *Semiclassical fourier transform for quantum computation*, Physical Review Letters, 76 (1996), p. 3228.

[63] D. GROSS, Y.-K. LIU, S. T. FLAMMIA, S. BECKER, AND J. EISERT, *Quantum state tomography via compressed sensing*, Physical review letters, 105 (2010), p. 150401.

[64] M. GRÖTSCHEL, L. LOVÁSZ, AND A. SCHRIJVER, *Geometric algorithms and combinatorial optimization*, Algorithms and Combinatorics, (1993).

[65] G. G. GUERRESCHI AND M. SMELYANSKIY, *Practical optimization for hybrid quantum-classical algorithms*, arXiv preprint arXiv:1701.01450, (2017), pp. 1–25.

[66] T. GUR, M.-H. HSIEH, AND S. SUBRAMANIAN, *Sublinear quantum algorithms for estimating von Neumann entropy*, (2021), pp. 1–40.

[67] C. GYURIK, D. VAN VREUMINGEN, AND V. DUNJKO, *Structural risk minimization for quantum linear classifiers*, arXiv preprint arXiv:2105.05566, (2021).

[68] O. HIGGOTT, D. WANG, AND S. BRIERLEY, *Variational quantum computation of excited states*, Quantum, 3 (2019), p. 156.

[69] Z. HOLMES, K. SHARMA, M. CEREZO, AND P. J. COLES, *Connecting ansatz expressibility to gradient magnitudes and barren plateaus*, arXiv preprint arXiv:2101.02138, (2021).

[70] R. HORODECKI, P. HORODECKI, M. HORODECKI, AND K. HORODECKI, *Quantum entanglement*, Reviews of Modern Physics, 81 (2009), pp. 865–942.

[71] H.-Y. HUANG, K. BHARTI, AND P. REBENTROST, *Near-term quantum algorithms for linear systems of equations*, arXiv, (2019), pp. 1–22.

[72] H. Y. HUANG, R. KUENG, AND J. PRESKILL, *Predicting many properties of a quantum system from very few measurements*, Nature Physics, (2020), pp. 1–40.

[73] W. HUGGINS, P. PATIL, B. MITCHELL, K. B. WHALEY, AND E. M. STOUDENMIRE, *Towards quantum machine learning with tensor networks*, Quantum Science and Technology, 4 (2019), p. 024001.

[74] R. ISLAM, R. MA, P. M. PREISS, M. ERIC TAI, A. LUKIN, M. RISPOLI, AND M. GREINER, *Measuring entanglement entropy in a quantum many-body system*, Nature, 528 (2015), pp. 77–83.

[75] E. T. JAYNES, *Information theory and statistical mechanics*, Physical review, 106 (1957), p. 620.

[76] H. JIANG, Y. T. LEE, Z. SONG, AND S. C.-W. WONG, *An Improved Cutting Plane Method for Convex Optimization, Convex-Concave Games and its Applications*, arXiv preprint arXiv:2004.04250, (2020).

[77] T. JONES, S. ENDO, S. MCARDLE, X. YUAN, AND S. C. BENJAMIN, *Variational quantum algorithms for discovering hamiltonian spectra*, Physical Review A, 99 (2019), p. 062304.

[78] A. T. KALAI AND S. VEMPALA, *Simulated Annealing for Convex Optimization*, Mathematics of Operations Research, 31 (2006), pp. 253–266.

[79] A. KANDALA, A. MEZZACAPO, K. TEMME, M. TAKITA, M. BRINK, J. M. CHOW, AND J. M. GAMBETTA, *Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets*, Nature, 549 (2017), pp. 242–246.

[80] D. B. KAPLAN, N. KLCO, AND A. ROGGERO, *Ground States via Spectral Combing on a Quantum Computer*, arXiv:1709.08250, (2017).

[81] N. KARMARKAR, *A new polynomial-time algorithm for linear programming*, in Proceedings of the sixteenth annual ACM symposium on Theory of computing, 1984, pp. 302–311.

[82] M. J. KASTORYANO AND F. G. BRANDAO, *Quantum gibbs samplers: the commuting case*, Communications in Mathematical Physics, 344 (2016), pp. 915–957.

[83] J. E. KELLEY, JR, *The cutting-plane method for solving convex programs*, Journal of the society for Industrial and Applied Mathematics, 8 (1960), pp. 703–712.

[84] M. KIEFEROVA, O. M. CARLOS, AND N. WIEBE, *Quantum generative training using rényi divergences*, arXiv preprint arXiv:2106.09567, (2021).

[85] M. KIEFEROVÁ AND N. WIEBE, *Tomography and generative training with quantum Boltzmann machines*, Physical Review A, 96 (2017), p. 062327.

[86] D. P. KINGMA AND J. BA, *Adam: A method for stochastic optimization*, arXiv preprint arXiv:1412.6980, (2014).

[87] A. Y. KITAEV, *Quantum measurements and the abelian stabilizer problem*, arXiv preprint quant-ph/9511026, (1995).

[88] E.-M. KONTOPOULOU, A. GRAMA, W. SZPANKOWSKI, AND P. DRINEAS, *Randomized Linear Algebra Approaches to Estimate the Von Neumann Entropy of Density Matrices*, in 2018 IEEE International Symposium on Information Theory (ISIT), vol. 2018-June, IEEE, jun 2018, pp. 2486–2490.

[89] T. D. LADD, F. JELEZKO, R. LAFLAMME, Y. NAKAMURA, C. MONROE, AND J. L. O'BRIEN, *Quantum computers*, Nature, 464 (2010), pp. 45–53.

[90] R. LAROSE, A. TIKKU, É. O,ÄôNEEL-JUDY, L. CINCIO, AND P. J. COLES, *Variational quantum state diagonalization*, npj Quantum Information, 5 (2019), pp. 1–10.

[91] Y. T. LEE, A. SIDFORD, AND S. S. VEMPALA, *Efficient Convex Optimization with Membership Oracles*, arXiv preprint arXiv:1706.07357, (2017).

[92] Y. T. LEE, A. SIDFORD, AND S. C.-W. WONG, *A Faster Cutting Plane Method and its Implications for Combinatorial and Convex Optimization*, in 2015 IEEE 56th Annual Symposium on Foundations of Computer Science, vol. 2015-Decem, IEEE, oct 2015, pp. 1049–1065.

[93] T. LI AND X. WU, *Quantum Query Complexity of Entropy Estimation*, IEEE Transactions on Information Theory, 65 (2019), pp. 2899–2921.

[94] L. LIN, *Lecture Notes on Quantum Algorithms for Scientific Computation*, (2022).

[95] N. M. LINKE, S. JOHRI, C. FIGGATT, K. A. LANDSMAN, A. Y. MATSUURA, AND C. MONROE, *Measuring the Rényi entropy of a two-site Fermi-Hubbard model on a trapped ion quantum computer*, Physical Review A, 98 (2018), p. 052334.

[96] J.-G. LIU, Y.-H. ZHANG, Y. WAN, AND L. WANG, *Variational quantum eigensolver with fewer qubits*, Physical Review Research, 1 (2019), p. 023025.

[97] Z. LIU, L.-W. YU, L. M. DUAN, AND D.-L. DENG, *The Presence and Absence of Barren Plateaus in Tensor-network Based Machine Learning*, (2021), pp. 1–28.

[98] S. LLOYD, M. MOHSENI, AND P. REBENTROST, *Quantum principal component analysis*, Nature Physics, 10 (2014), pp. 631–633.

[99] P. M. LONG AND R. A. SERVEDIO, *Restricted boltzmann machines are hard to approximately evaluate or simulate*, ICML, (2010).

[100] A. LUONGO AND C. SHAO, *Quantum algorithms for spectral sums*, (2020).

[101] P. MAGNARD, P. KURPIERS, B. ROYER, T. WALTER, J.-C. BESSE, S. GASPARINETTI, M. PECHAL, J. HEINSOO, S. STORZ, A. BLAIS, ET AL., *Fast and unconditional all-microwave reset of a superconducting qubit*, Physical review letters, 121 (2018), p. 060502.

[102] C. O. MARRERO, M. KIEFEROVÁ, AND N. WIEBE, *Entanglement Induced Barren Plateaus*, (2020), pp. 1–14.

[103] J. MARTYN AND B. SWINGLE, *Product spectrum ansatz and the simplicity of thermal states*, Physical Review A, 100 (2019), p. 032107.

[104] S. MCARDLE, T. JONES, S. ENDO, Y. LI, S. C. BENJAMIN, AND X. YUAN, *Variational ansatz-based quantum simulation of imaginary time evolution*, npj Quantum Information, 5 (2019), pp. 1–6.

[105] J. R. MCCLEAN, S. BOIXO, V. N. SMELYANSKIY, R. BABBUSH, AND H. NEVEN, *Barren plateaus in quantum neural network training landscapes*, Nature Communications, 9 (2018), p. 4812.

[106] J. R. MCCLEAN, J. ROMERO, R. BABBUSH, AND A. ASPURU-GUZIK, *The theory of variational hybrid quantum-classical algorithms*, New Journal of Physics, 18 (2016), p. 023023.

[107] K. MITARAI, M. NEGORO, M. KITAGAWA, AND K. FUJII, *Quantum circuit learning*, Physical Review A, 98 (2018), p. 032309.

[108] A. MONTANARO AND R. DE WOLF, *A survey of quantum property testing*, Theory of Computing, 2016 (2016), pp. 1–81.

[109] M. MOTTA, C. SUN, A. T. K. TAN, M. J. O'ROURKE, E. YE, A. J. MINNICH, F. G. S. L. BRANDÃO, AND G. K.-L. CHAN, *Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution*, Nature Physics, 16 (2020), pp. 205–210.

[110] K. M. NAKANISHI, K. MITARAI, AND K. FUJII, *Subspace-search variational quantum eigensolver for excited states*, Physical Review Research, 1 (2019), p. 033062.

[111] M. A. NIELSEN AND I. CHUANG, *Quantum computation and quantum information*, 2002.

[112] M. OSTASZEWSKI, E. GRANT, AND M. BENEDETTI, *Quantum circuit structure learning*, (2019), pp. 1–11.

[113] R. B. PATEL, J. HO, F. FERREYROL, T. C. RALPH, AND G. J. PRYDE, *A quantum Fredkin gate*, Science advances, 2 (2016), p. e1501531.

[114] A. PERUZZO, J. MCCLEAN, P. SHADBOLT, M.-H. YUNG, X.-Q. ZHOU, P. J. LOVE, A. ASPURU-GUZIK, AND J. L. O'BRIEN, *A variational eigenvalue solver on a photonic quantum processor*, Nature Communications, 5 (2014), p. 4213.

[115] A. PESAH, M. CEREZO, S. WANG, T. VOLKOFF, A. T. SORNBORGER, AND P. J. COLES, *Absence of barren plateaus in quantum convolutional neural networks*, 2020.

[116] D. PETZ, *Quasi-entropies for finite quantum systems*, Reports on Mathematical Physics, 23 (1986), pp. 57–65.

[117] D. POULIN AND P. WOCJAN, *Sampling from the thermal quantum gibbs state and evaluating partition functions with a quantum computer*, Physical review letters, 103 (2009), p. 220502.

[118] J. PRESKILL, *Quantum Computing in the NISQ era and beyond*, Quantum, 2 (2018), p. 79.

[119] A. G. RATTEW, S. HU, M. PISTOIA, R. CHEN, AND S. WOOD, *A domain-agnostic, noise-resistant, hardware-efficient evolutionary variational quantum eigensolver*, arXiv preprint arXiv:1910.09694, (2019).

[120] A. G. RATTEW, Y. SUN, P. MINSSEN, AND M. PISTOIA, *Quantum Simulation of Galton Machines Using Mid-Circuit Measurement and Reuse*, (2020).

[121] M. D. REED, B. R. JOHNSON, A. A. HOUCK, L. DICARLO, J. M. CHOW, D. I. SCHUSTER, L. FRUNZIO, AND R. J. SCHOELKOPF, *Fast reset and suppressing spontaneous emission of a superconducting qubit*, Applied Physics Letters, 96 (2010), p. 203110.

[122] F. REIF, *Fundamentals of statistical and thermal physics*, Waveland Press, 2009.

[123] A. RÉNYI, *On measures of entropy and information*, in Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics, University of California Press, 1961, pp. 547–561.

[124] A. RIERA, C. GOGOLIN, AND J. EISERT, *Thermalization in Nature and on a Quantum Computer*, Physical Review Letters, 108 (2012), p. 080402.

[125] D. RISTÈ, J. G. VAN LEEUWEN, H.-S. KU, K. W. LEHNERT, AND L. DICARLO, *Initialization by measurement of a superconducting quantum bit circuit*, Physical review letters, 109 (2012), p. 050507.

[126] A. W. ROBERTS, *Convex functions*, in Handbook of Convex Geometry, Elsevier, 1993, pp. 1081–1104.

[127] M. SCHULD, V. BERGHOLM, C. GOGOLIN, J. IZAAC, AND N. KILLORAN, *Evaluating analytic gradients on quantum hardware*, Physical Review A, 99 (2019), p. 032331.

[128] B. SCHUMACHER, *Quantum coding*, Physical Review A, 51 (1995), p. 2738.

[129] SETH LLOYD, *Universal Quantum Simulators*, Science, 273 (1996), pp. 1073–1078.

[130] C. E. SHANNON, *A mathematical theory of communication*, ACM SIGMOBILE mobile computing and communications review, 5 (2001), pp. 3–55.

[131] K. SHARMA, S. KHATRI, M. CEREZO, AND P. J. COLES, *Noise resilience of variational quantum compiling*, New Journal of Physics, 22 (2020), p. 043006.

[132] S. SIM, P. D. JOHNSON, AND A. ASPURU,ÄêGUZIK, *Expressibility and Entangling Capability of Parameterized Quantum Circuits for Hybrid Quantum,ÄêClassical Algorithms*, Advanced Quantum Technologies, 2 (2019), p. 1900070.

[133] R. D. SOMMA, S. BOIXO, H. BARNUM, AND E. KNILL, *Quantum simulations of classical annealing processes*, Physical review letters, 101 (2008), p. 130504.

[134] Y. SUBASI, L. CINCIO, AND P. J. COLES, *Entanglement spectroscopy with a depth-two quantum circuit*, Journal of Physics A: Mathematical and Theoretical, 52 (2019), p. 044001.

[135] S. SUBRAMANIAN AND M.-H. HSIEH, *Quantum algorithm for estimating α-renyi entropies of quantum states*, Physical Review A, 104 (2021), p. 022428.

[136] R. SWEKE, F. WILDE, J. MEYER, M. SCHULD, P. K. FÄHRMANN, B. MEYNARD-PIGANEAU, AND J. EISERT, *Stochastic gradient descent for hybrid quantum-classical optimization*, arXiv preprint arXiv:1910.01155, (2019), pp. 1–23.

[137] K. TEMME, T. J. OSBORNE, K. G. VOLLBRECHT, D. POULIN, AND F. VERSTRAETE, *Quantum metropolis sampling*, Nature, 471 (2011), pp. 87–90.

[138] B. M. TERHAL AND D. P. DIVINCENZO, *Problem of equilibration and the computation of correlation functions on a quantum computer*, Physical Review A, 61 (2000), p. 022301.

[139] A. VALENTI, E. VAN NIEUWENBURG, S. HUBER, AND E. GREPLOVA, *Hamiltonian learning for quantum error correction*, Physical Review Research, 1 (2019), p. 033092.

[140] J. VAN APELDOORN, A. GILYÉN, S. GRIBLING, AND R. DE WOLF, *Quantum sdp-solvers: Better upper and lower bounds*, in 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS), IEEE, 2017, pp. 403–414.

[141] G. VERDON, M. BROUGHTON, AND J. BIAMONTE, *A quantum algorithm to train neural networks using low-depth circuits*, arXiv:1712.05304, (2017).

[142] G. VERDON, J. MARKS, S. NANDA, S. LEICHENAUER, AND J. HIDARY, *Quantum hamiltonian-based models and the variational quantum thermalizer algorithm*, Bulletin of the American Physical Society, 65 (2020).

[143] J. VON NEUMANN, *Mathematische grundlagen der quantenmechanik springer*, 1932.

[144] J. WANG, S. PAESANI, R. SANTAGATI, S. KNAUER, A. A. GENTILE, N. WIEBE, M. PETRUZZELLA, J. L. O'BRIEN, J. G. RARITY, A. LAING, AND M. G. THOMPSON, *Experimental quantum Hamiltonian learning*, Nature Physics, 13 (2017), pp. 551–555.

[145] S. WANG, E. FONTANA, M. CEREZO, K. SHARMA, A. SONE, L. CINCIO, AND P. J. COLES, *Noise-Induced Barren Plateaus in Variational Quantum Algorithms*, 2 (2020), pp. 1–20.

[146] X. WANG, Z. SONG, AND Y. WANG, *Variational Quantum Singular Value Decomposition*, Quantum, 5 (2021), p. 483.

[147] Y. WANG, G. LI, AND X. WANG, *A Hybrid Quantum-Classical Hamiltonian Learning Algorithm*, (2021), pp. 1–18.

[148] ——, *Variational Quantum Gibbs State Preparation with a Truncated Taylor Series*, Physical Review Applied, 16 (2021), p. 054035.

[149] Y. WANG, B. ZHAO, AND X. WANG, *Quantum algorithms for estimating quantum entropies*, (2022), pp. 22–26.

[150] Z. WANG, S. HADFIELD, Z. JIANG, AND E. G. RIEFFEL, *Quantum approximate optimization algorithm for MaxCut: A fermionic view*, Physical Review A, 97 (2018), p. 022304.

[151] J. WATROUS, *Quantum Computational Complexity*, in Computational Complexity, vol. 9781461418, Springer New York, New York, NY, apr 2012, pp. 2361–2387.

[152] D. WECKER, M. B. HASTINGS, AND M. TROYER, *Training a quantum optimizer*, Physical Review A, 94 (2016), p. 022309.

[153] N. WIEBE, C. GRANADE, AND D. G. CORY, *Quantum bootstrapping via compressed quantum Hamiltonian learning*, New Journal of Physics, 17 (2015), p. 022005.

[154] N. WIEBE, C. GRANADE, C. FERRIE, AND D. CORY, *Quantum Hamiltonian learning using imperfect quantum resources*, Physical Review A - Atomic, Molecular, and Optical Physics, 89 (2014), pp. 1–18.

[155] N. WIEBE, C. GRANADE, C. FERRIE, AND D. G. CORY, *Hamiltonian Learning and Certification Using Quantum Resources*, Physical Review Letters, 112 (2014), p. 190501.

[156] N. WIEBE, A. KAPOOR, AND K. M. SVORE, *Quantum deep learning*, Quantum Information and Computation, 16 (2016), pp. 541–587.

[157] J. WU AND T. H. HSIEH, *Variational Thermal Quantum Simulation via Thermofield Double States*, Physical Review Letters, 123 (2019), p. 220502.

[158] X. XU, J. SUN, S. ENDO, Y. LI, S. C. BENJAMIN, AND X. YUAN, *Variational algorithms for linear algebra*, 2 (2019), pp. 1–10.

[159] J. YIRKA AND Y. SUBASI, *Qubit-efficient entanglement spectroscopy using qubit resets*, (2020).

[160] X. YUAN, S. ENDO, Q. ZHAO, Y. LI, AND S. C. BENJAMIN, *Theory of variational quantum simulation*, Quantum, 3 (2019), p. 191.

[161] M.-H. YUNG AND A. ASPURU-GUZIK, *A quantum-quantum Metropolis algorithm*, Proceedings of the National Academy of Sciences, 109 (2012), pp. 754–759.

[162] P. ZANARDI, *Quantum entanglement in fermionic lattices*, Physical Review A, 65 (2002), p. 042101.

[163] S.-X. ZHANG, C.-Y. HSIEH, S. ZHANG, AND H. YAO, *Differentiable quantum architecture search*, Quantum Science and Technology, 7 (2022), p. 045023.