ABSTRACT

Title of Dissertation: OPTIMIZATION PROBLEMS
IN QUANTUM MACHINE LEARNING

Xuchen You
Doctor of Philosophy, 2023

Dissertation Directed by: Professor Xiaodi Wu
Department of Computer Science

The variational algorithm is a paradigm for designing quantum procedures implementable on noisy intermediate-scale quantum (NISQ) machines. It is viewed as a promising candidate for demonstrating practical quantum advantage.

In this dissertation, we look into the optimization aspect of the variational quantum algorithms as an attempt to answer when and why a variational quantum algorithm works. We mainly focus on two instantiations of the family of variational algorithms, the Variational Quantum Eigensolvers (VQEs) and the Quantum Neural Networks (QNNs).

We first established that, for almost all QNN architecture designs, there exist hard problem instances leading to an optimization landscape swarmed by spurious local minima provided that the QNN is under-parameterized. This observation rules out the possibility of a universal good QNN design achieving exponential advantage against the classical neural networks on any dataset and calls for instance-dependent designs for variational circuits.

We then show that VQE training converges linearly when the number of parameters exceeds an over-parameterization threshold. By tying the threshold to instance-dependent quantities, we developed variants of VQE algorithms that allow the training and testing of shallower variational circuits, as depths are usually the implementation bottlenecks on NISQ machines.

For QNNs, by looking into its convergence, we show that the dynamics of QNN training are different from the dynamics of any kernel regression, therefore ruling out the popular conjecture that over-parameterized QNNs are equivalent to certain versions of neural tangent kernels like their classical counterparts. As a practical implication, our analysis showcases the measurement design as a way to accelerate the convergence of QNNs.

At the end of this dissertation, we consider the classical problem of optimization with partial information, the Multi-arm Bandits (MABs). We show that, when enhanced with quantum access to the arms, there is a quadratic speed-up against the classical algorithms, which can serve as the building block for quantum reinforcement learning algorithms.

Optimization Problems in Quantum Machine Learning

by

Xuchen You

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2023

Advisory Committee:
Professor Xiaodi Wu, Chair/Advisor
Professor Alexander Barg, Dean's Representative
Professor Furong Huang
Professor Kaiqing Zhang
Professor Tianyi Zhou

Acknowledgments

I could not have undertaken this journey without Xiaodi Wu being the best advisor one could hope for: My taste and approaches to research problems were shaped through countless weekly meetings with him. In additional, his mentorship (both in research and life), provided me with freedom to explore topics of my interest. I am immensely grateful for his unwavering support and belief in me.

I am also indebted to Furong Huang, from whom I gained the hands-on knowledge of machine learning during her group meetings and long discussions. She was incredibly understanding and supportive when I made the decision to shift my research direction. Moreover, I would like to express my gratitude to Liwei Wang. It was through his lectures, group meetings and reading groups that I was first introduced to the field of machine learning.

I am deeply grateful to Alexander Barg, Soheil Feizi, Furong Huang, Xiaodi Wu, Kaiqing Zhang, Tianyi Zhou for the insightful questions and thoughtful feedbacks during my preliminary examination and defense process.

This dissertation would not have been possible without my fantastic collaborators: Samyadeep Basu, Boyang Chen, Andrew Childs, Soheil Feizi, Furong Huang, Jialin Li, Tongyang Li, Daochen Wang, Xiaodi Wu. I would like to thank my dear friends, Shouvanik Chakrabarti, Zitan Chen, Seyed Esmaelli, Alejandro Flores-Velazco, Hongye Hu, Jiaqi Leng, Jialin Li, Jingling Li, Tongyang Li, Yi Mao, Yuxiang Peng, Nirat Saini, Jiahao Su, Pattara Sukprasert, Daochen Wang, Sheng

Yang, Shaopeng Zhu, as well as all lab mates, office mates, house mates and cohort members: it is simply impossible to list each and every one of their names. Special thanks to Shouvanik, Jialin and Zitan, for their support and reminding me that there is so much more to life than research. Thanks also go to the staffs of the CS department and QUICS, in particular Tom Hurst and Andrea Svejda for their generous help.

Finally I thank my parents and grandparents, my wife and my cats. This dissertation is dedicated to you.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Motivations

Quantum computing leverages the quantum properties of matter to accelerate classical computations: on the theory side, a broad line of algorithms based on the pioneering works of Shor and Grover have been proven to outperform their classical counterpart on practical tasks ranging from factoring to reinforcement learning (e.g. [2, 3, 4]); on the experimental side, there has been convincing preliminary results suggesting the quantum advantage in certain sampling tasks (e.g. [5, 6]) synthetically coined to favor existing quantum computers. With the recent progress of quantum computers, the community seeks for a practical task, where the quantum advantage can be firmly established.

The Variational Quantum Algorithm (VQA,[7]) is a family of algorithms involving quantum systems that are controlled by classical parameters. A VQA is composed of a variational quantum circuit (as a parameterized model) and a classical optimization routine (most commonly, variants of the gradient descent method). At each iteration, the variational circuit is evaluated and then the outcomes are passed to the classical optimizer to update the parameters.

By delegating precise control to reliable classical computers, VQAs can potentially enjoy the best of both worlds: utilizing the exponential expressive of the space of quantum states while (partially) retaining the reliability of a classical computer. Such a hybrid of classical

and quantum procedures is a promising candidate for demonstrating the quantum advantage on Noisy Intermediate-Scale Quantum (NISQ) computers [8] available in the near term that are limited in scales and lack error tolerance. In addition, the VQA is versatile as a paradigm: by choosing different objective functions, a VQA can realize functionalities including tasks in machine learning (e.g. [9, 10]), quantum chemistry (e.g. [11, 12]) and combinatorial optimization (e.g. [13]).

Apart from the number of qubits available, the bottleneck of the implementation of VQAs on a NISQ machine is the depth of the variational circuit due to the noisy nature of NISQ machines; the total cost of the quantum resources for running a variational algorithm is jointly characterized by the depth of the circuit and the number of evaluations of the circuit required for finding a good set of parameters. In practice, the depth of circuit is closely related to the number of parameters governing the variational circuit. More formally, let p denote the number of parameters in a variational circuit, let ϵ be the target optimality of the VQA solution, let T be the number of gradient steps ran on the circuit, and let δ be the failure rate due to the non-convex nature of the objective function. We aim at understanding the relationship among p, T, ϵ and δ by studying the optimization of a variational quantum algorithm from the theoretical perspective. As we will see later, a theoretical understanding of when and why a VQA works or fails leads to a principled way for designing VQAs with better performances.

1.2 Contributions

In this dissertation, we aim at developing a theory of VQAs using tools from the optimization literature, tying the quantities T, p, ϵ and δ mentioned above to the size of the problem (e.g. the

dimension d of the ambient space where the problem is embedded) and other instance-dependent quantities:

1. In Chapter 2, we prove a no-go theorem for quantum neural networks (QNNs, an instantiation of the VQA), stating that when the number of parameters p is of order $O(\log d)$, the QNN optimization landscape is swarmed by sub-optimal local minima. This establishes a lower bound on the number of parameters for VQA trainability under gradient descent.

2. In Chapter 3, we provide the first rigorous proof for the linear convergence of the Variational Quantum Eigensolvers (VQEs, another instantiation of the VQA) when the number of parameters p is $\Omega(\mathsf{poly}(d))$, exceeding the *trainability* threshold. We further tie the *trainability* threshold to specific VQE problem instances and the design of variational circuits.

3. In Chapter 4, we demonstrate that our analysis framework in Chapter 3 is applicable to general variational quantum algorithms. Specifically, we apply the framework to QNNs and observe that, contrary to the popular belief, a QNN, whether it is over-parameterized or not, does not follow the dynamics of the kernel regression of any kernel, therefore refuting the conjecture that over-parameterized QNNs are equivalent to quantum neural tangent kernels. Our analysis also show that the convergence of QNNs can be accelerated by simply scaling the readout measurements.

4. In Chapter 5, we demonstrate that our characterization in Chapter 3 leads to a principled way of designing VQEs. Specifically

    - We provide an sampling-based procedure for predicting and comparing the performances of variational circuits with different designs. Our procedure does not require repeated

training over different random initializations and sweeping over different number of parameters;

- We provide a two-stage routine that can be used to compress the VQE run-time circuit; such routine is especially useful when the goal is to repeatedly prepare a certain ground state for measuring its properties.
- We provide a preconditioning routine that allows the training of shallower circuits at the price of more readout measurements.

In addition, we extend our scope to optimization problems beyond the family of variational algorithms in Chapter 6 and provide a fully quantum algorithm for solving the exploration problem in multi-arm bandits, achieving a quadratic speedup against any classical algorithms.

1.3 Preliminaries

In this section we introduce necessary backgrounds on quantum information and linear algebra. For further readings, we refer the readers to the books by Nielsen and Chuang [14] and by Watrous [15].

1.3.1 Linear Algebra

Basic notions. We use $\mathbf{I}_{d\times d}$ to denote the identity matrix in $\mathbb{C}^{d\times d}$. The subscripts are omitted when there is no ambiguity on the dimension. Let $\dagger$ denote the conjugate transpose of complex matrices. A matrix $\mathbf{U} \in \mathbb{C}^{d\times d}$ is said to be a unitary if $\mathbf{U}\mathbf{U}^\dagger = \mathbf{I}_{d\times d}$. A matrix is said to be Hermitian if its conjugate transpose is itself: $\mathbf{A} = \mathbf{A}^\dagger$. Let $[\cdot,\cdot]$ denote the commutator of two matrices, such that $[\mathbf{A},\mathbf{B}] := \mathbf{A}\mathbf{B} - \mathbf{B}\mathbf{A}$.

Inner product and adjoints. The inner product between two Hermitians $\mathbf{A}$ and $\mathbf{B}$ is defined as $\langle \mathbf{A}, \mathbf{B} \rangle := \mathrm{tr}(\mathbf{AB})$. For a linear map $\Phi(\cdot)$, the adjoint of the linear map $\Phi^*(\cdot)$ is defined such that $\langle \mathbf{A}, \Phi(\mathbf{B}) \rangle = \langle \Phi^*(\mathbf{A}), \mathbf{B} \rangle$. The Frobenius norm of a Hermitian $\mathbf{A}$ is defined as $\|\mathbf{A}\|_F := \sqrt{\langle \mathbf{A}, \mathbf{A} \rangle}$. We additionally use $\|\cdot\|_{\mathsf{op}}$ and $\|\cdot\|_{\mathrm{tr}}$ to denote the operator norm (i.e. the largest eigenvalue in terms of the absolute values) and the trace norm of matrices; we use $\|\cdot\|_p$ to denote the p-norm of vectors, with the subscript omitted for $p = 2$.

Exponent of matrices. The exponent of a Hermitian $\mathbf{A}$ is defined using Taylor expansion $\exp(\mathbf{A}) := \sum_{k=0}^{\infty} \frac{\mathbf{A}^k}{k!}$, Let $\{E_j\}_{j=1}^d$ and $\{\mathbf{u}_j\}_{j=1}^d$ be the eigenvalues and eigenvectors of $\mathbf{A}$, We have

$$\exp(\mathbf{A}) := \sum_{k=0}^{\infty} \frac{\mathbf{A}^k}{k!} = \sum_{k=0}^{\infty} \sum_{j=1}^{d} \frac{1}{k!} E_j^k \mathbf{u}_j \mathbf{u}_j^\dagger = \sum_{j=1}^{d} e^{E_j} \mathbf{u}_j \mathbf{u}_j^\dagger, \tag{1.1}$$

For $\theta \in \mathbb{R}$ and Hermitian $\mathbf{H}$, $\exp(-i\theta\mathbf{H})$ is unitary, since:

$$\exp(-i\theta\mathbf{H})^\dagger = \Big(\sum_{k=0}^{\infty} \frac{\big(-i\theta\mathbf{H}\big)^k}{k!}\Big)^\dagger = \sum_{k=0}^{\infty} \frac{\big(i\theta\mathbf{H}\big)^k}{k!} = \exp(i\theta\mathbf{H}), \tag{1.2}$$

and

$$\exp(i\theta\mathbf{H}) \exp(-i\theta\mathbf{H}) = \big(\sum_{j=1}^{d} e^{i\theta E_j} \mathbf{u}_j \mathbf{u}_j^\dagger\big) = \big(\sum_{k=1}^{d} e^{-i\theta E_k} \mathbf{u}_k \mathbf{u}_k^\dagger\big) = \mathbf{I}_{d\times d}. \tag{1.3}$$

Kronecker product. For $\mathbf{A} \in \mathbb{C}^{d_1 \times d_2}$ and $\mathbf{B} \in \mathbb{C}^{d_2 \times d_2}$, the Kronecker product of $\mathbf{A}$ and $\mathbf{B}$ is defined as:

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} A_{11}\mathbf{B} & A_{12}\mathbf{B} & \cdots & A_{1d_1}\mathbf{B} \\ A_{21}\mathbf{B} & A_{22}\mathbf{B} & \cdots & A_{2d_1}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ A_{d_1 1}\mathbf{B} & A_{d_1 2}\mathbf{B} & \cdots & A_{d_1 d_1}\mathbf{B} \end{bmatrix} \tag{1.4}$$

where A_{ij} is the (i, j)-th element of matrix $\mathbf{A}$. As can be seen from the definition $\mathbf{A} \otimes \mathbf{B} \in \mathbb{C}^{d_1 d_2 \times d_1 d_2}$. We also use the symbol $\otimes$ for direct product of Hilbert spaces depending on the context. By definition, the trace of $\mathbf{A} \otimes \mathbf{B}$

$$\operatorname{tr}\big(\mathbf{A} \otimes \mathbf{B}\big) = \sum_{j=1}^{d_1} A_{jj} \operatorname{tr}(\mathbf{B}) = \operatorname{tr}(\mathbf{A}) \operatorname{tr}(\mathbf{B}). \tag{1.5}$$

1.3.2 Quantum Information

We now describe notions in quantum information using linear algebra. We start by formulating quantum mechanics in the language of density matrices.

Quatum states. A *quantum state* with dimension d is can be represented by a positive semidefinite (PSD) Hermitian $\boldsymbol{\rho} \in \mathbb{C}^{d \times d}$, the *density matrix*. For any PSD Hermitian $\boldsymbol{\rho}$ with $\operatorname{tr}(\boldsymbol{\rho}) = 1$, there is a correponding quantum state, and vice versa. Quantum states with rank-1 density matrices are referred to as *pure states*. For a pure state, its density matrix $\boldsymbol{\rho}$ allows a eigen-decompostion $\boldsymbol{\rho} = \mathbf{v}\mathbf{v}^\dagger$ with $\mathbf{v} \in \mathbb{C}^d$ being a ℓ_2-unit vector. $\mathbf{v}$ is referred to as the state vector representation of a pure quantum state.

The state space of a system composed of two subsystem with dimension d_1 and d_2 has a dimension of $d_1 d_2$. And a density matrix for a such composite system lies in $\mathbb{C}^{d_1 \times d_1} \otimes \mathbb{C}^{d_2 \times d_2}$. If a state can be expressed as a Kronecker product of density matrices, we say it is a *product state*. A simple example is $\boldsymbol{\rho}_{12} := \boldsymbol{\rho}_1 \otimes \boldsymbol{\rho}_2$ with $\boldsymbol{\rho}_1 \in \mathbb{C}^{d_1}$ and $\boldsymbol{\rho}_2 \in \mathbb{C}^{d_2}$.

Analogous to classical binary bits, the basic element for quantum computers are *qubits*. The state space of a single qubit is 2-dimensional, and the density matrix for a system composed of n qubits lies in $\otimes_{i=1}^{n} \mathbb{C}^{2\times 2}$.

Unitary gates. An operation over a quantum state is a linear map that is completely positive and preserves the trace (See [15] for a rigorous definition). Throughout this paper we focus on *unitary operators*. In the context of quantum circuit models, operations are also referred to as *gates*.

Under the density matrix representation, a unitary gate, denoted by $\mathbf{U} \in \mathbb{C}^{d\times d}$, transforms a state $\boldsymbol{\rho} \in \mathbb{C}^{d\times d}$ to $\boldsymbol{\rho}' = \mathbf{U}\boldsymbol{\rho}\mathbf{U}^\dagger$. The positive semidefiniteness and the trace are preserved. For a pure state $\boldsymbol{\rho} = \mathbf{v}\mathbf{v}^\dagger$, under the state vector representation, the unitary gate $\mathbf{U}$ transforms $\mathbf{v}$ to $\mathbf{v}' = \mathbf{U}\mathbf{v}$.

A set of unitary gates commonly used on a qubit are the *Pauli gates*:

$$\mathbf{X} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \ \mathbf{Y} = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \ \mathbf{Z} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}; \tag{1.6}$$

Note that the Pauli gates are both unitary and Hermitian.

We are especially interested in unitary gates parameterized by $\theta \in \mathbb{R}$ as $\exp(-i\theta\mathbf{H})$ for a Hermitian $\mathbf{H}$. We sometimes refer to $\mathbf{H}$ as the *Hamiltonian* that generates the parameterized

quantum gate. For any $\lambda \in \mathbb{R}$, shifting $\mathbf{H}$ by $\lambda \cdot \mathbf{I}$ does not change the parameterized gate. To see this, for any $\boldsymbol{\rho}$, consider $\tilde{\mathbf{H}}_\lambda := \mathbf{H} + \lambda \mathbf{I}$:

$$e^{-i\theta\tilde{\mathbf{H}}_\lambda}\boldsymbol{\rho}e^{i\theta\tilde{\mathbf{H}}_\lambda} = e^{-i\theta\lambda} \cdot e^{-i\theta\mathbf{H}}\boldsymbol{\rho}e^{i\theta\mathbf{H}} \cdot e^{i\theta\lambda} = e^{-i\theta\mathbf{H}}\boldsymbol{\rho}e^{i\theta\mathbf{H}}. \tag{1.7}$$

Quantum measurements and observables. A measurement of quantum states is specified by a set of matrices $\{\mathbf{M}_m\} \subset \mathbb{C}^{d\times d}$ with $\mathbf{M}_m^\dagger \mathbf{M}_m = \mathbf{I}_{d\times d}$, and a set of outcomes $\{\lambda_m\} \subset \mathbb{R}$. Such measurements on the density matrix $\boldsymbol{\rho}$ yield the outcome λ_m with probability $\operatorname{tr}\big(\mathbf{M}_m^\dagger \mathbf{M}_m \boldsymbol{\rho}\big)$. The probabilities are normalized as

$$\sum_m \operatorname{tr}\big(\mathbf{M}_m^\dagger \mathbf{M}_m \boldsymbol{\rho}\big) = \operatorname{tr}\big(\sum_m \mathbf{M}_m^\dagger \mathbf{M}_m \boldsymbol{\rho}\big) = \operatorname{tr}\big(\boldsymbol{\rho}\big) = 1. \tag{1.8}$$

In this paper we mainly focus on the expected outcome of a measurement. Let $\mathbf{M}$ denote the *observable* $\sum_m \lambda_m \mathbf{M}_m^\dagger \mathbf{M}_m$, the expected value of the outcome is

$$\sum_m \lambda_m \operatorname{tr}\big(\mathbf{M}_m^\dagger \mathbf{M}_m \boldsymbol{\rho}\big) = \operatorname{tr}\big(\sum_m \lambda_m \mathbf{M}_m^\dagger \mathbf{M}_m \boldsymbol{\rho}\big) = \operatorname{tr}\big(\mathbf{M}\boldsymbol{\rho}\big). \tag{1.9}$$

1.3.3 Variational Quantum Algorithms

Variational Quantum Algorithms (VQA) are a paradigm of quantum algorithms that searches over a family of parameterized quantum operations referred to as variational quantum circuits (also referred to as *quantum ansatze*). More concretely, a p-parameter ansatz on a d-dimensional Hilbert space $\mathbf{U}\colon \mathbb{R}^p \to \mathbb{C}^{d\times d}$ maps real parameters $\boldsymbol{\theta}$ to a unitary operator $\mathbf{U}(\boldsymbol{\theta})$. A variational quantum circuit resembles a classical neural network in terms of the layered structure.

Classical neural networks. Neural networks are parameterized families of mappings widely considered for practical problems. Typical feed-forward neural networks are parameterized by a sequence of matrices $\{\mathbf{W}_i\}_{i=1}^t$, such that $\mathbf{W}_i \in \mathbb{R}^{d_i \times d_{i-1}}$, with $d_t = 1$ and d_0 is the same as the dimension of the feature space $\mathcal{X}$. For a feature vector $\mathbf{x}$, the output $\hat{y}$ of the neural network is

$$\hat{y} = \mathbf{W}_t \sigma(\mathbf{W}_{t-1} \sigma(\cdots \sigma(\mathbf{W}_1 \mathbf{x}) \cdots)), \tag{1.10}$$

where $\sigma(\cdot)$ denotes an element-wise activation on the output of each layer. (See Figure 1.1.) Linear neural networks [16] is one special example where σ is the identity mapping $\sigma(w) = w$: $\hat{y} = \mathbf{W}_t \mathbf{W}_{t-1} \cdots \mathbf{W}_1 \mathbf{x}$. Another example is one-hidden layer neural networks with quadratic activation $\sigma(w) = w^2$ [17], where the output $\hat{y} = \mathbf{x}^T \mathbf{W}_1^T \mathbf{W}_1 \mathbf{x}$. Given the training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$, the empirical risk minimization with square loss solves the optimization problem:

$$\min_{\mathbf{W}_1} \frac{1}{m} \sum_{i=1}^{m} \left(\mathbf{x}_i^T \mathbf{W}_1^T \mathbf{W}_1 \mathbf{x}_i - y_i\right)^2 \tag{1.11}$$

A common choice of $\sigma(\cdot)$ is non-linear activations such as Relu or Sigmoid.

Variational quantum circuits. Variational quantum circuits share with classical neural networks the layered structure (Figure 1.1) where a linear transformation $\mathbf{U}_i$ is applied on the output of the previous layer with the following differences:

(1) *Input*. The inputs to classical neural networks are feature vectors. Yet for variational circuits, the inputs are quantum states $\boldsymbol{\rho}$ that are from the context of the physical problem or that encode classical data (e.g., [18, 19, 20]).

(2) *Linear Transformation & Parameterization*. The linear transformations $\{\mathbf{W}_i\}_{i=1}^t$ in

classical neural networks could be general matrices, whereas the corresponding $\{\mathbf{U}_i\}_{i=1}^t$ in variational circuits must be unitaries. Moreover, although $\{\mathbf{U}_i\}_{i=1}^t$ can be efficiently implemented by quantum machines, their classical representations are matrices of exponential dimension in terms of the system size (e.g., the number of qubits). This makes classical simulation of variational circuits extremely expensive (except for certain special cases, e.g. [21]) and also makes the parameterizations of $\{\mathbf{U}_i\}_{i=1}^t$ different from the straightforward parameterizations of $\{\mathbf{W}_i\}_{i=1}^t$: instead of entrywise-parameterized matrices W_i, variational quantum circuits typically consist of *classically parameterized* quantum gates. A general form of these gates is $\exp(-i\theta\mathbf{H})$, where θ is the parameter, $\mathbf{H}$ the Hamiltonian (i.e., a Hermitian matrix), and the exponential is a *matrix* exponential. For example, a commonly used gate set, called the Pauli rotation gate (e.g., [9, 22]), can be expressed as $\exp(-i\theta\mathbf{P}_c)$ (on c-th qubit) or $\exp(-i\theta\mathbf{P}_c \otimes \mathbf{P}_{c'})$ (on c-th and c'-th qubits), where $\mathbf{P}_c$ refers to Pauli $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ matrices. We can also group gates with respect to the layer structure in Figure 1.1 by putting gates that can be executed in parallel in the same layer. For example, let $\mathbf{V}_{i,j}(\theta_{i,j}) = \exp(-i\theta_{i,j}\mathbf{H}_{i,j})$ be the jth gate in the ith layer. Then $\mathbf{U}_i(\boldsymbol{\theta}) = \prod_j \mathbf{V}_{i,j}(\theta_{i,j})$ and

$$\mathbf{U}(\boldsymbol{\theta}) = \mathbf{U}_t(\boldsymbol{\theta})\mathbf{U}_{t-1}(\boldsymbol{\theta})\cdots\mathbf{U}_1(\boldsymbol{\theta}), \tag{1.12}$$

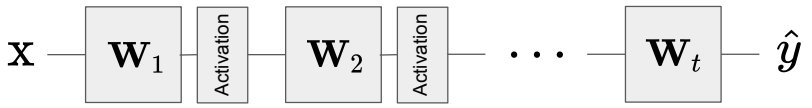
where $\mathbf{U}(\boldsymbol{\theta})$ refers to the unitary transformation of the entire circuit with parameters $\boldsymbol{\theta}$.

For technical convenience and to highlight the dependence on the number of parameters p, we consider the general expression for a p-parameter quantum circuit:

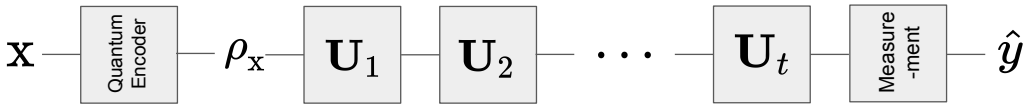
$$\mathbf{U}(\boldsymbol{\theta}) = \mathbf{V}_p(\theta_p)\mathbf{V}_{p-1}(\theta_{p-1})\cdots\mathbf{V}_1(\theta_1), \tag{1.13}$$

with $\mathbf{V}_l(\theta_l) = e^{-i\theta_l \mathbf{H}_l}$ for Hermitian $\mathbf{H}_l$ and $l \in [p]$. Without loss of generality, we assume $\mathbf{H}_l$'s are traceless.

(3) *Output.* Contrary to classical neural networks, one needs to make a quantum *measurement* to read information from variational circuits (explained below). While there exist more advanced models of variational circuits with additional nonlinearity, we mainly consider the most basic versions, where the measurements are the only source of slight non-linearity allowed by quantum mechanics.



(a) Classical neural networks



(b) Variational Quantum Circuits

Figure 1.1: An illustration of layer-structured classical neural networks and variational quantum circuits.

The variational quantum circuits, just like neural networks for deep learing, are nothing but a family of parameterized models. When equipped with different objective functions, they can be used for tasks like eigen-decomposition, supervised learning and distribution learning.

Optimization of Variational Circuits. A typical tool for solving the optimal parameters in VQA is gradient descent. Given an objective function L dependent on the variational parameters $\boldsymbol{\theta} \in \mathbb{R}^p$, the randomly initialized parameters are updated according to the following rule:

$$\boldsymbol{\theta}(t+1) \leftarrow \boldsymbol{\theta}(t) - \eta \nabla L(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}(t)}, \tag{1.14}$$

where η is the learning rate. We also consider the setting of gradient flow: when the learning rate

η is sufficiently small, the dynamics of gradient descent reduces to that of gradient flow

$$\frac{d}{dt}\boldsymbol{\theta}(t) = -\eta \nabla L(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}(t)}. \tag{1.15}$$

Chapter 2: Exponentially Many Local Minima in Quantum Neural Networks

When used for supervised learning tasks, variational quantum circuits are referred to as Quantum Neural Networks (QNNs). They are important quantum applications both because of their similar promises as classical neural networks and because of the feasibility of their implementation on near-term intermediate-size noisy quantum machines. However, the training task of QNNs is challenging and much less understood. In this chapter, we focus on the hardness of training quantum neural networks: We conduct a quantitative investigation on the landscape of loss functions of QNNs and identify a class of simple yet extremely hard QNN instances for training. Specifically, we show for typical under-parameterized QNNs, there exists a dataset that induces a loss function with the number of spurious local minima depending exponentially on the number of parameters. Moreover, we show the optimality of our construction by providing an almost matching upper bound on such dependence. While local minima in classical neural networks are due to non-linear activations, in quantum neural networks local minima appear as a result of the quantum interference phenomenon. Finally, we empirically confirm that our constructions can indeed be hard instances in practice with typical gradient-based optimizers, which demonstrates the practical value of our findings.

2.1 Introduction

Similar to the classical case, the success of QNN applications will critically depend on the effectiveness of the training procedure which optimizes a *loss function* in terms of the *read-outs* and the *parameters* of QNNs for specific applications. The design of effective training methods has been under intense investigation both empirically and theoretically for classical NNs. Moreover, understanding the landscape of the loss functions and designing corresponding training/optimization methods have recently emerged as a principled approach to tackle this problem: [23, 24, 25, 26, 27] showed the existence of spurious local minima for classical NNs; in turn, [16, 17, 28, 29, 30] characterized conditions for benign landscapes in terms of choice of activation, loss function and (over)-parameterization, providing insights on the design of classical NNs and motivating explanations to the success of gradient descent in training classical NNs in certain scenarios [31, 32, 33]; and training methods beyond simple variants of gradient descent have been devised for training with guarantees [34, 35, 36, 37].

Much less has been understood for QNNs. Most of the study of QNNs takes a trial-and-error approach by empirically comparing the performance of standard classical optimizers on training QNNs' loss functions [19]. It has been observed empirically that training QNNs could be very challenging due to the *non-convex* nature of the corresponding loss functions (e.g., [38, 39]). However, these empirical studies are unfortunately restricted to small cases due to the limited access to quantum machines of reasonable sizes and the exponential cost in simulating them classically.

A theoretical study on the training of QNNs would be more *favorable* and *scalable* given the limit on empirical study. Indeed, a handful of such theoretical progress has been made.

One prominent result is that random initialization of parameters will lead to vanishing gradients for much smaller size QNNs than ClaNNs [40] and hence pose one unique training difficulty for QNNs. Most of the remaining theoretical results are about special cases of QNNs such as *quantum approximate optimization algorithms* (QAOA) (e.g., [13, 41]) and extremely over-parameterized cases (e.g., [42, 43, 44]).

In this chapter, we conduct a quantitative investigation on the landscape of loss functions for QNNs as a way to study their training issue. In particular, we are interested in understanding the properties of local minima of loss functions, such as, (1) the number of local minima depending on the architecture of QNNs, and (2) whether these local minima are *benign* or *spurious* ones, meaning that they are either close to the global minima or saddle points that can be escaped, or they are truly bad local minima that will hinder the training procedure. We are also motivated by the observation that QNNs share some similarity with linear neural networks without non-linear activation layers [16] or one-hidden layer neural networks with quadratic activation [17] that are both known to have only benign local minima. The similarity is due to the fact that quantum mechanics underlying QNNs has a linear algebraic formulation similar to the linear part of classical NNs. (See for example Section 1.3.) It is hence natural to wonder whether the local minima of QNNs could share these nice properties.

Contributions. Contrary to our original hope, we identify a class of *simple yet extremely hard* instances of QNNs for the training. Despite the similarity between QNNs and linear classical neural networks, we demonstrate that *spurious* (or *sub-optimal*) local minima do appear in QNNs and provide a quantitative characterization of the possible number of them. We focus on QNNs with the commonly used *square loss* function under a practical range of the number of parameters (or gates). Specifically, we identify a general condition of under-parameterized

QNNs, which we refer to as QNNs *with linear independence*. We show for such QNNs, a dataset can be constructed such that the number of spurious local minima scales *exponentially* with the number of parameters. It demonstrates that QNNs behave quite differently from linear neural networks (e.g., [16]) but share the feature of neurons with *non-linear* activation functions (e.g., [23]). This conceptual paradox could be explained by one central phenomenon of quantum mechanics behind QNNs called *interference*. We observe that interference replaces the role of non-linear activation in creating bad local minima for QNNs. (Section 2.3)

We investigate further and prove that typical under-parameterized QNNs are indeed *with linear independence*. This indicates that for almost all under-parameterized QNNs, there is a dataset where training with simple variants of gradient-based methods is hard. (Section 2.4)

Moreover, we show our construction is almost *optimal* in terms of the dependence of the number of local minima on the number of parameters, by developing an almost matching upper bound. This upper bound also demonstrates a sharp separation between QNNs and ClaNNs: For ClaNNs, provided an arbitrary number of training samples, the number of local minima could be unbounded, and hence won't be upper bounded by any function of the number of parameters [23]. (Section 2.5)

Finally, we perform numerical experiments on concrete QNN instances with typical optimizers, and empirically confirm that our constructions can indeed be hard instances in practice. These experiments strengthen the value of our theoretical findings on the practical end. (Section 2.6)

It is worthwhile mentioning that our investigation on the landscape of loss functions has a direct implication on the hardness of gradient-based methods. While it does not rule out the possibility of efficient non-gradient-based training, there are no obvious solutions to the efficient training for our constructions. Identifying such training methods would be very interesting.

Previous works. There are only a few previous studies on the training of QNNs, each of which has targeted at some specific parameter range for QNNs. The observation of vanishing gradients for random initialization of QNNs [40] provides hard QNN instances for training, which, however, still require many layers to demonstrate the difficulty of training in practice. Our constructions are based on a general condition which includes simple special cases like 1-layer QNNs that are already able to demonstrate QNNs' training difficulty.

Another line of work [42, 43, 44] considers the extremely over-parameterized QNN cases. Specifically, when the number of parameters is comparable to the dimension of the underlying quantum system and the quantum *controllability* condition can be established, all local minima of QNNs' loss functions will become global [42, 43]. This theoretical prediction has also been observed empirically [44]. However, as the dimension of quantum systems grows *exponentially* with the number of qubits, this over-parameterized case can hardly be realistic for any QNN of reasonable size.

2.2 Preliminaries

In this section, we formally introduce the quantum neural networks as an instantiation of variational quantum algorithms introduced in Section 1.3. We start by describing the task of supervised learning:

Supervised learning. The goal of supervised learning is to identify a mapping from the feature space $\mathcal{X}$ to the label space $\mathcal{Y}$, given a training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^m \subset (\mathcal{X} \times \mathcal{Y})^m$ of m samples of *feature vectors* and *labels*. A common practice to find a mapping based on a training set is through *empirical risk minimization* (ERM), finding a mapping that best align with the training

sample with respect to a specific loss function $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$. Let $\hat{y}_i$ be the prediction of a certain mapping given $\mathbf{x}_i$ for $i \in [m]$. The goal of ERM is to find the mapping that minimizes the average loss $\frac{1}{m}\sum_{i=1}^{m} l(\hat{y}_i, y_i)$. In this chapter, we focus on the square loss $l(\hat{y}, y) = (\hat{y} - y)^2$.

Quantum neural networks. An instance of a d-dimensional, p-parameter and m-sample QNN is specified by a tuple $(\mathbf{M}, \mathcal{S}, \mathbf{U})$, with $\mathbf{M} \in \mathbb{C}^{d\times d}$, $\mathcal{S} \subset (\mathbb{C}^{d\times d} \times \mathbb{R})^m$ and $\mathbf{U} : \mathbb{R}^p \to \mathbb{C}^{d\times d}$. $\mathbf{M}$ is a Hermitian matrix representing the measurement for the readout; $\mathcal{S} = \{(\boldsymbol{\rho}_j, y_j)\}^m$ is the training set composed of m pairs of density matrices and labels. $\mathbf{U}$ is the variational circuit / ansatz, mapping p real parameters to a unitary matrix. The ERM for the QNN instance $(\mathbf{M}, \mathcal{S}, \mathbf{U})$ solves the following optimization problem:

$$\min_{\boldsymbol{\theta}\in\mathbb{R}^p} L(\boldsymbol{\theta}; \mathcal{S}) = \frac{1}{m}\sum_{i=1}^{m} \left(\mathrm{tr}(\mathbf{U}(\boldsymbol{\theta})\boldsymbol{\rho}_i\mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}) - y_i\right)^2. \tag{2.1}$$

When input quantum states are pure, namely $\boldsymbol{\rho}_i = \mathbf{v}_i\mathbf{v}_i^\dagger$ for $i \in [m]$, the loss function becomes

$$L(\boldsymbol{\theta}; \mathcal{S}) = \frac{1}{m}\sum_{j=1}^{m} \left(\mathbf{v}_j^\dagger\mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}\mathbf{U}(\boldsymbol{\theta})\mathbf{v}_j - y_j\right)^2 \tag{2.2}$$

which resembles the loss function of one-hidden layer neural networks with quadratic activation except for the restriction of $\mathbf{U}$ being unitary transformations.

Characterization of the landscape. For a differentiable function F defined on an unconstrained domain, $\boldsymbol{\theta}^*$ is a *critical* point if and only if the gradient vanishes at the point: $\nabla F(\boldsymbol{\theta}^*) = \mathbf{0}$. $\boldsymbol{\theta}$ is a local minimum if and only if there is an open set U containing $\boldsymbol{\theta}^*$ such that $F(\boldsymbol{\theta}^*) \leq F(\boldsymbol{\theta})$ for all $\boldsymbol{\theta} \in U$. A local minimum is global if the minimum value of F is attained at θ^*. For twice-differentiable function over an unconstrained domain, $\boldsymbol{\theta}^*$ is a local minimum if the Hessian is

positive definite at $\boldsymbol{\theta}^*$ (sufficient condition) and only if $\boldsymbol{\theta}^*$ is a critical point (necessary condition).

2.3 Exponentially Many Spurious Local Minima for Under-parameterized QNNs

In this section, we present our main result on the constructions of datasets for p-parameter quantum neural network instances with $\Omega(2^p)$ spurious local minima. We consider QNNs defined in Eqn. (1.13), with parameterized gates $\mathbf{V}_l(\theta_l)$ generated by $\mathbf{H}_l$:

$$\mathbf{U}(\boldsymbol{\theta}) = \exp(-i\theta_p \mathbf{H}_p) \cdots \exp(-i\theta_2 \mathbf{H}_2) \exp(-i\theta_1 \mathbf{H}_1) \tag{2.3}$$

We assume that $\{\mathbf{H}_l\}_{l=1}^p$ are with eigenvalues ± 1. This is the case for single- and multi-qubit parameterized gates generated by Kronecker products of Pauli matrices.

As stated in Section 1.3, shifting $\mathbf{H}_l$ by $\lambda \mathbf{I}$ for any $\lambda \in \mathbb{R}$ introduces a global phase factor to the output state and does not change the output $f(\boldsymbol{\rho}, \boldsymbol{\theta})$. Also, shifting the observable $\mathbf{M}$ by $\lambda \mathbf{I}$ is equivalent to shifting the labels in the dataset by $-\lambda$. Without loss of generality, we assume $\mathrm{tr}(\mathbf{H}_l) = 0$ and $\mathrm{tr}(\mathbf{M}) = 0$.

We start by characterizing the output $f(\boldsymbol{\rho}, \boldsymbol{\theta}) := \mathrm{tr}(\mathbf{U}(\boldsymbol{\theta}) \boldsymbol{\rho} \mathbf{U}(\boldsymbol{\theta})^\dagger \mathbf{M})$. For any $l \in [p]$, define linear maps $\Phi_l^{(0)}(\cdot)$, $\Phi_l^{(1)}(\cdot)$ and $\Phi_l^{(2)}(\cdot)$ such that

$$\Phi_l^{(0)}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} + \mathbf{H}_l \mathbf{A} \mathbf{H}_l) \tag{2.4}$$

$$\Phi_l^{(1)}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} - \mathbf{H}_l \mathbf{A} \mathbf{H}_l) \tag{2.5}$$

$$\Phi_l^{(2)}(\mathbf{A}) = \frac{i}{2}[\mathbf{H}_l, \mathbf{A}] \tag{2.6}$$

Here $[\cdot, \cdot]$ is the commutator of two matrices. For any Hermitian $\mathbf{A}$, $\Phi_l^{(0)}(\mathbf{A})$ commutes with $\mathbf{H}_l$,

and the output of $\Phi_l^{(1)}$ and $\Phi_l^{(2)}$ anti-commute with $\mathbf{H}_l$. For any vector $\boldsymbol{\xi} \in \{0,1,2\}^p$, define:

$$\Phi_{\boldsymbol{\xi}}(\mathbf{A}) = \Phi_1^{(\xi_1)} \circ \Phi_2^{(\xi_2)} \circ \cdots \circ \Phi_p^{(\xi_p)}(\mathbf{A}) \tag{2.7}$$

with $\circ$ denoting the composition of mappings.

The observable in Heisenberg picture $\mathbf{M}(\boldsymbol{\theta}) := \mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}\mathbf{U}(\boldsymbol{\theta})$ can be expanded as:

$$\sum_{\boldsymbol{\xi}\in\{0,1,2\}^p} \Phi_{\boldsymbol{\xi}}(\mathbf{M}) \prod_{l:\xi_l=1} \cos 2\theta_l \prod_{l':\xi_{l'}=2} \sin 2\theta_{l'} \tag{2.8}$$

The QNN output $f(\boldsymbol{\rho}, \boldsymbol{\theta}) = \mathrm{tr}(\boldsymbol{\rho}\mathbf{M}(\boldsymbol{\theta}))$ can be expressed as the following trigonometric polynomial:

$$\sum_{\boldsymbol{\xi}\in\{0,1,2\}^p} \mathrm{tr}(\boldsymbol{\rho}\Phi_{\boldsymbol{\xi}}(\mathbf{M})) \prod_{l:\xi_l=1} \cos 2\theta_l \prod_{l':\xi_{l'}=2} \sin 2\theta_{l'} \tag{2.9}$$

As shown in Section 2.8, the loss function remains invariant under the joint transformation $\theta_l \mapsto \theta_l + \frac{\pi}{2}$ and

$$\Phi_l^{(0)}(\cdot) \mapsto \mathbf{H}_l\Phi_l^{(0)}(\cdot)\mathbf{H}_l = \Phi_l^{(0)}(\cdot) \tag{2.10}$$

$$\Phi_l^{(1)}(\cdot) \mapsto \mathbf{H}_l\Phi_l^{(1)}(\cdot)\mathbf{H}_l = -\Phi_l^{(1)}(\cdot) \tag{2.11}$$

$$\Phi_l^{(2)}(\cdot) \mapsto \mathbf{H}_l\Phi_l^{(2)}(\cdot)\mathbf{H}_l = -\Phi_l^{(2)}(\cdot) \tag{2.12}$$

Under the transformation $\theta_l \mapsto \theta_l + \frac{\pi}{2}$, terms in Eqn. (2.9) associated with $\boldsymbol{\xi} : \xi_l = 0$ are invariant, while terms associated with $\boldsymbol{\xi} : \xi_l = 1, 2$ flip signs.

From an alternative perspective, $L(\boldsymbol{\theta}; \mathcal{S})$ contains *oscillating wave* components proportional to $\cos 4\theta_l$, $\sin 4\theta_l$, $\cos 2\theta_l$ and $\sin 2\theta_l$, hence periodic with π on each coordinate. However, due

the existence of lower frequency, the periodicity with $\frac{\pi}{2}$ does not always hold for all datasets. Our construction utilizes the presence and absence of this $\frac{\pi}{2}$-translational symmetry.

We will focus on a general class of QNN, which we call QNN *with linear independence*:

Definition 2.1 (QNN with linear independence)**.** A QNN is said to be with linear independence, if the associated set of $3^p - 1$ operators $\{\Phi_{\boldsymbol{\xi}}(\mathbf{M})\}_{\boldsymbol{\xi}\in\{0,1,2\}^p,\boldsymbol{\xi}\neq\mathbf{0}}$ forms a linearly independent set.

Note that for the linear independence condition to hold, the dimension of the QNN $d \geq 3^{p/2}$. Namely, it is a under-parameterized case, which differentiates us from the over-parameterized ones [42, 43, 44]. Our main result states:

Theorem 2.1 (Construction: exponentially many local minima)**.** *Consider QNNs composed of unitaries generated by two-level Hamiltonians, parameterized by $\boldsymbol{\theta} \in \mathbb{R}^p$. If the QNN is with linear independence, a dataset $\mathcal{S}$ can be constructed to induce a loss function $L(\boldsymbol{\theta}; \mathcal{S})$ with 2^p local minima within each period, and $2^p - 1$ of these minima are spurious with positive suboptimality gap.*

Proof of Theorem 2.1. The dataset we construct is composed of two parts $\mathcal{S}_0$ and $\mathcal{S}_1$. The first component of the loss function $L(\boldsymbol{\theta}; \mathcal{S}_0)$ is constructed with 2^p local minima using the $\frac{\pi}{2}$-translational symmetry:

Lemma 2.2 (Creating symmetry)**.** *For QNNs with linear independence as mentioned in Theorem 2.1, a dataset $\mathcal{S}_0$ can be constructed to induce a loss function $L(\boldsymbol{\theta}; \mathcal{S}_0)$ that (1) has a local minimum at some $\boldsymbol{\theta}^\star$, and (2) is invariant under translation $\theta_l \mapsto \theta_l + \frac{\pi}{2}$ for all $l \in [p]$.*

Due to the translational invariance, for any $\boldsymbol{\zeta} \in \{0, 1\}^p$, $\boldsymbol{\theta}^\star + \frac{\pi}{2}\boldsymbol{\zeta}$ is a local minimum for $L(\boldsymbol{\theta}; \mathcal{S}_0)$, forming a total of 2^p local minima. A second dataset $\mathcal{S}_1$ is introduced to break this symmetry, creating spurious local minima:

Lemma 2.3 (Breaking symmetry)**.** *Consider the QNN, dataset $\mathcal{S}_0$ and local minimum $\boldsymbol{\theta}^\star$ defined in Lemma 2.2. Let Θ denote the set of 2^p local minima due to the translational invariance. There exists a dataset $\mathcal{S}_1$ such that*

$$\inf_{\boldsymbol{\theta}\in\mathcal{N}(\boldsymbol{\theta}^\star)} L(\boldsymbol{\theta};\mathcal{S}_0)+L(\boldsymbol{\theta};\mathcal{S}_1) < \inf_{\boldsymbol{\theta}\in\mathcal{N}(\boldsymbol{\theta}')} L(\boldsymbol{\theta};\mathcal{S}_0)+L(\boldsymbol{\theta};\mathcal{S}_1) \tag{2.13}$$

for all $\boldsymbol{\theta}' \in \Theta/\{\boldsymbol{\theta}^\star\}$, and that

$$L(\boldsymbol{\theta};\mathcal{S}_0)+L(\boldsymbol{\theta};\mathcal{S}_1) > L(\boldsymbol{\theta}';\mathcal{S}_0)+L(\boldsymbol{\theta}';\mathcal{S}_1) \tag{2.14}$$

for all $\boldsymbol{\theta}' \in \Theta$ and all $\boldsymbol{\theta} \in \partial\mathcal{N}(\boldsymbol{\theta}')$. Here $\mathcal{N}(\cdot)$ denote a bounded and closed neighbourhood, such that $\mathcal{N}(\boldsymbol{\theta}) \cap \mathcal{N}(\boldsymbol{\theta}') = \emptyset$ for any $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta$; $\partial\mathcal{N}$ denotes its boundary.

Eqn. (2.14) in Lemma 2.3 ensures the existence of a local minimum within $\mathcal{N}(\boldsymbol{\theta})$ for each $\boldsymbol{\theta} \in \Theta$, and Eqn. (2.13) promises that only the local minimum within $\mathcal{N}(\boldsymbol{\theta}^\star)$ achieves the global optimal value. Combining $\mathcal{S}_0$ and $\mathcal{S}_1$ finishes the proof for Theorem 2.1. □

We give proof sketches for Lemma 2.2 and 2.3. The full proofs are postponed to Section 2.8.

Proof sketch for Lemma 2.2. It suffices to construct a dataset $\mathcal{S}_0 = \{(\boldsymbol{\rho}_k, y_k)\}_{k=1}^{m_0}$, such that (1) for all $k \in [p]$, $f_k(\boldsymbol{\theta}) := \langle \boldsymbol{\rho}_k, \mathbf{M}(\boldsymbol{\theta})\rangle - y_k$ is either symmetric or anti-symmetric under $\theta_l \mapsto \theta_l + \frac{\pi}{2}$ for all $l \in [p]$, and (2) the intersection Θ of the set of roots Θ_k of $f_k(\boldsymbol{\theta}) = 0$ is non-empty and contains at least one isolated point $\boldsymbol{\theta}^\star$. For such $\mathcal{S}_0$, $\boldsymbol{\theta}^\star$ is an isolated root of the non-negative loss function $L(\boldsymbol{\theta};\mathcal{S}_0) = \sum_{k=1}^{m_0} f_k(\boldsymbol{\theta})^2$.

The existence of such dataset $\mathcal{S}_0$ follows from the linear independence of operators for the QNN. As a result, for any $k \in [m_0]$, the solution to the following linear system for Hermitian

$\mathbf{D}_k \in \mathbb{C}^{d\times d}$ is non-empty:

$$\begin{cases} \operatorname{tr}(\mathbf{D}_k \cdot \mathbf{I}) = 0, \\ \operatorname{tr}(\mathbf{D}_k \cdot \Phi_{\boldsymbol{\xi}}(\mathbf{M})) = \hat{f}_{\boldsymbol{\xi},k},\ \forall \boldsymbol{\xi} \neq \mathbf{0}. \end{cases} \tag{2.15}$$

Here $\hat{f}_{\boldsymbol{\xi},k}$ is the coefficient corresponding to the term $\prod_{l:\xi_l=1} \cos 2\theta_l \prod_{l':\xi_{l'}=2} \sin 2\theta_{l'}$ in $f_k(\boldsymbol{\theta})$. Given the solution $\{\mathbf{D}_k\}_{k=1}^{m_0}$, $\mathcal{S}_0$ can be constructed by setting $\boldsymbol{\rho}_k := \frac{1}{d}\mathbf{I} + \kappa \mathbf{D}_k$ for a proper scaling factor κ and let $y_k = \operatorname{tr}(\boldsymbol{\rho}_k \Phi_{\mathbf{0}}(\mathbf{M}))$. □

Proof sketch for Lemma 2.3. Rewrite the loss function as

$$L(\boldsymbol{\theta}; \mathcal{S}_1) = -\frac{2}{m_1}\sum_{k=1}^{m_1} y_k \operatorname{tr}(\boldsymbol{\rho}_k \mathbf{M}(\boldsymbol{\theta})) + \frac{1}{m_1}\sum_{k=1}^{m_1} \operatorname{tr}(\boldsymbol{\rho}_k \mathbf{M}(\boldsymbol{\theta}))^2 + \frac{1}{m_1}\sum_{k=1}^{m_1} y_k^2 \tag{2.16}$$

As will be made clear in Section 2.8, our key observation is that, under a joint scaling of y_k and $\boldsymbol{\rho}_k$, the second term can be arbitrarily suppressed while the first term remains the same. Therefore it suffices to study the first term $L'(\boldsymbol{\theta}; \mathcal{S}_1) := -\frac{2}{m_1}\sum_{k=1}^{m_1} y_k \operatorname{tr}(\boldsymbol{\rho}_k \mathbf{M}(\boldsymbol{\theta}))$. The linear independence allows us to solve a linear system to construct $\mathcal{S}_1$ that satisfies the requirements in Lemma 2.3. □

Remarks. The statements above involve unitaries generated by two-level Hamiltonians only. For more general local quantum gates, $\{\mathbf{H}_l\}_{l=1}^{p}$ are allowed to have more than two distinct eigenvalues. We are especially interested in Hamiltonians with eigenvalues $\{E_1, \cdots, E_d\} \subset \mathbb{Z}$, as arbitrary Hamiltonians with rational spectrum can be converted to ones with integral spectrum with proper shifting and scaling. Theorem 2.1 can be generalized for $\mathbf{H}_l$'s with largest eigen-gap $\max_{c,c'\in[d]} |E_c - E_{c'}|$ bounded by Δ, with the number of spurious local minima being $\Omega(\Delta^p)$.

This observation further supports the intuition of interference as the source of local minima.

1-layer QNN. A simple example of QNNs with linear independence is a one-layer circuit with local $\mathbf{H}_l$ acting on the l-th qubit, and a product operator $\mathbf{M}$ as the observable:

Proposition 2.3.1 (One-layer QNNs with product observables)**.** *Consider the family of QNNs composed of unitaries generated by two-level Hamiltonians, parameterized by $\boldsymbol{\theta} \in \mathbb{R}^p$. For all $l \in [p]$, let $\mathbf{H}_l$ be a local Hamiltonian on the l-qubit, taking the form $\mathbf{I} \otimes \cdots \otimes \mathbf{h}_l \otimes \cdots \otimes \mathbf{I}$ for some Hermitian $\mathbf{h}_l$ at the l-th position, and $\mathbf{M} = \mathbf{m}_1 \otimes \cdots \otimes \mathbf{m}_p$ such that $\mathbf{m}_l + \mathbf{h}_l\mathbf{m}_l\mathbf{h}_l$ and $\mathbf{m}_l - \mathbf{h}_l\mathbf{m}_l\mathbf{h}_l$ are non-zero for any l. There exists a dataset that induces a loss function with $2^p - 1$ spurious local minima.*

This follows from the fact that $\mathrm{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'}(\mathbf{M})) = 0$ if and only if $\boldsymbol{\xi} \neq \boldsymbol{\xi}'$. In Section 2.8, we provide proof for Proposition 2.3.1 and several concrete example QNNs to demonstrate that our construction can have local minima at arbitrary $\boldsymbol{\theta}$, and does not allow trivial solutions such as coordinate-wise greedy optimization.

2.4 Typical QNNs are with Linear Independence

In Section 2.3, we provided a general condition (Definition 2.1) for QNNs to have exponentially many bad local minima for some datasets. In this section, we show that this condition is met for typical under-parameterized QNNs. To see this, we consider the following measure over instances of QNNs: Let $\mathbf{H}$ be a d-dimensional Hermitian such that $\mathrm{tr}(\mathbf{H}) = 0$ and $\mathbf{H}^2 = \mathbf{I}$. A random circuit $\mathbf{U}(\boldsymbol{\theta})$ is specified as

$$\mathbf{U}(\boldsymbol{\theta}) = e^{-i\theta_p \mathbf{W}_p \mathbf{H} \mathbf{W}_p^\dagger} \cdots e^{-i\theta_1 \mathbf{W}_1 \mathbf{H} \mathbf{W}_1^\dagger} \tag{2.17}$$

with $\{\mathbf{W}_l\}_{l=1}^p$ independently sampled with respect to the Haar measure on the d-dimensional unitary group $U(d)$.

Up to a unitary transformation, this random model is equivalent to a circuit with p interleaving parameterized gate $\{e^{-i\theta_l \mathbf{H}}\}_{l=1}^p$ and unitary $\{\tilde{\mathbf{W}}_l\}_{l=1}^p$ randomly sampled with respect to the Haar measure:

$$\mathbf{U}(\boldsymbol{\theta}) = \tilde{\mathbf{W}}_p e^{-i\theta_p \mathbf{H}} \tilde{\mathbf{W}}_{p-1} \cdots \tilde{\mathbf{W}}_1 e^{-i\theta_1 \mathbf{H}} \tag{2.18}$$

The equivalence is due to the left (or right) invariance of the Haar measure. This interleaving nature of fixed and parameterized gates are shared by existing designs of QNNs, and any p-parameter QNN generated by two-level Hamiltonians can be expressed in Eqn. (2.18). Moreover, applying polynomially many random 2-qubit gates on random pairs of qubits generates a distribution over gates that approximates the Haar measure up to the 4-th moments [45], which is what we require in this section.

The Gram matrix for the set $\{\Phi_{\boldsymbol{\xi}}(\mathbf{M})\}_{\boldsymbol{\xi}\in\{0,1,2\}^p, \boldsymbol{\xi}\neq\mathbf{0}}$ is defined such that the element corresponding to the pair $(\boldsymbol{\xi}, \boldsymbol{\xi}')$ is $\mathrm{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'}(\mathbf{M}))$. The Gram matrix is always positive semidefinite, and a positive definite Gram matrix implies the linear independence of the set.

Using the integral formula with respect to Haar measure on unitary groups [46], we can estimate the expectations and variances of the diagonal and off-diagonal terms, and upper bound the probability of the event:

$$\exists \boldsymbol{\xi} : \mathrm{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})^2) \leq \sum_{\boldsymbol{\xi}'\neq\boldsymbol{\xi}} |\,\mathrm{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'}(\mathbf{M}))| \tag{2.19}$$

Applying the Gershgorin circle theorem [47], we can lower bound the probability for a random QNN to have linear independent terms:

Theorem 2.4 (Typical under-parameterized QNNs are with linear independence)**.** *Consider a random p-parameter d-dimensional QNN with two-level Hamiltonians sampled from the model specified in Eqn. (2.17). Let the observable* $\mathbf{M}$ *be an arbitrary non-zero trace-*0 *Hermitian. Such QNN is with linear independence with probability* $\geq 1 - O(d^{-1})$ *for fixed* p*, and with probability* $\geq 1 - O(e^{-p})$ *for dimension* $d : \log(d) = \Theta(p)$.

Please refer to Section 2.9 for the full proof of Theorem 2.4.

2.5 Upper Bound on the Number of Local Minima

Our construction above possesses 2^p local minima for p parameters, whereas the classical work of [23] demonstrates a construction for a single neuron with $\lfloor m/p \rfloor^p$ local minima for m training samples. Note that the latter could grow unboundedly with m. In this section, we show, however, this classical unbounded growth of local minima does not hold for QNNs. In fact, we could establish an almost matching upper bound for 2^p. All the formal proofs are deferred to Section 2.10.

To that end, let us examine the *Fourier* expansion of the loss function $L(\boldsymbol{\theta}, \mathcal{S})$ (Eqn. (2.1)). Let T_l be the period of $L(\boldsymbol{\theta}; \mathcal{S})$ corresponding to θ_l, and $\hat{L}(\mathbf{k})$ the Fourier coefficient for $\mathbf{k} = (k_1, \cdots, k_p)^T \in \mathbb{Z}^p$. We have

$$L(\boldsymbol{\theta}; \mathcal{S}) = \sum_{\mathbf{k} \in K} \hat{L}(\mathbf{k}) \prod_{l=1}^{p} \left(\cos \frac{k_l \theta_l}{T_l} + i \sin \frac{k_l \theta_l}{T_l} \right) \tag{2.20}$$

where $K \subseteq \mathbb{Z}^p$ is the support of the Fourier coefficients.

One critical observation is that, for arbitrary choice of two-level $\{\mathbf{H}_l\}_{l=1}^p$, observable $\mathbf{M}$ and training set $\mathcal{S}$, the support K of the Fourier spectrum is bounded in ℓ_1-norm: $\max_{\mathbf{k}\in K} \sum_{l=1}^p |k_l| \leq 2p$, indicating that the Fourier degree of $L(\boldsymbol{\theta}; \mathcal{S})$ is upper bounded by $2p$ (See Section 2.10.1).

By definition, a local minimum must be a critical point, hence it suffices to bound the number of critical points for functions with Fourier spectrum supported on a ℓ_1-bounded set. Define $G_l(\boldsymbol{\theta})$ as $\frac{\partial}{\partial \theta_l} L(\boldsymbol{\theta}; \mathcal{S})$:

$$G_l(\boldsymbol{\theta}) = \sum_{\mathbf{k}\in K} k_l \hat{L}(\mathbf{k}) \big(-\sin \frac{k_l \theta_1}{T_l} + i \cos \frac{k_l \theta_l}{T_l}\big) \cdot \prod_{l' \neq l} \big(\cos \frac{k_{l'} \theta_{l'}}{T_{l'}} + i \sin \frac{k_{l'} \theta_{l'}}{T_{l'}}\big) \tag{2.21}$$

Notice that the Fourier spectrum of G_l is supported on the same set K. A critical point of $L(\boldsymbol{\theta}; \mathcal{S})$ must satisfy that for all $l \in [p]$, $G_l(\boldsymbol{\theta}) = 0$. By basic trigonometry, $\cos k\theta$ can be expressed as a degree-k polynomial of $\cos\theta$ and $\sin k\theta$ as a degree-$(k-1)$ polynomial of $\cos\theta$ multiplied by $\sin\theta$. Consider the change of variable

$$c_l = \cos(\theta_l / T_l), \ s_l = \sin(\theta_l / T_l), \ \forall l \in [p]. \tag{2.22}$$

Let $g_l(c_1, s_1, \cdots, c_p, s_p)$ be the multivariate polynomial constraints corresponding to $G_l(\boldsymbol{\theta})$ after the change of variable. For each g_l, the sum of degrees of $c_{l'}$ and $s_{l'}$ is bounded by $\max_{\mathbf{k}\in K} |k_{l'}|$, and the degree $\deg(g_l)$ of g_l is bounded by $\max_{\mathbf{k}\in} \sum_{l=1}^p |k_l| \leq 2p$. The change of variable is one-to-one from $\theta_l \in [0, T_l)$ to a pair of $(c_l, s_l) \in \mathbb{R}^2$ under the constraint $c_l^2 + s_l^2 = 1$. Therefore, it suffices to count the number of roots of the polynomial system with $2p$ parameters and $2p$

constraints:

$$g_l(c_1, s_1, \cdots, c_p, s_p) = 0,\ c_l^2 + s_l^2 - 1 = 0 \tag{2.23}$$

for all $l \in [p]$. Notice that for a general polynomial system, the number of critical points can be unbounded. For example, consider a system composed of constant polynomials, every point in the domain is a critical point. This corresponds to the constant loss function, where the gradients vanish everywhere with positive semidefinite Hessians. For this reason, we will focus on the non-degenerated case with finitely many local minima. Under the premise of non-degeneracy, by Bézout's Theorem (e.g. Section 3.3 in [48]), the number of roots can be bounded by the product of the degree of polynomial constraints $2^p \mathsf{deg}(g_1)\mathsf{deg}(g_2)\cdots\mathsf{deg}(g_p) \leq (4p)^p$. A formal statement of the above derivation is as follows:

Theorem 2.5 (Upper bound: the number of local minima)**.** *Consider non-degenerated QNNs composed of unitaries generated by two-level Hamiltonians $\{H_l\}_{l=1}^p$ with p parameters. For training set $\mathcal{S}$, within each period, the loss function $L(\boldsymbol{\theta}; \mathcal{S})$ possesses at most $(4p)^p$ local minima.*

We also prove a similar result for the more general case where the generators are Hamiltonians with integral spectrum: let Δ be the largest eigen-gap for each of the Hamiltonians, the number of local minima within each period is upper bounded by $O((\Delta p)^p)$. Please refer to Section 2.10 for details.

2.6 Experiments

We investigate the practical performance of the common optimizers on our construction in this section. It is well-known in the classical literature that the existence of spurious local minima does not necessarily cause difficulties in optimization: When the suboptimality of spurious local minima is small (e.g.[49]), or when the random initialization avoids spurious local minima, gradient-descent methods can still converge with high probability to a local minimum competitive to the global minima. We show, however, our constructions can indeed be hard instances for training in practice.

To that end, we evaluate a specific construction from Proposition 2.3.1 in Section 2.3 by using the standard optimizers with randomly initialized parameters uniformly sampled from the domain and visualize the distribution of function values at convergence (for p-parameter instances, we uniformly sample the initial parameters from $[0, 2\pi)^p$).

For p-parameter instances, our construction involves p-qubits. We choose $\mathbf{h}_1 = \cdots = \mathbf{h}_p = \mathbf{Z}$ and $\mathbf{m}_1 = \cdots = \mathbf{m}_p = \mathbf{Y} + \mathbf{I}$. The specific form of the instance and all the training details are provided in Section 2.11.

Implementation The experiments are run on Intel Core i7-7700HQ Processor (2.80GHz) with 16G memory. We classically simulate the training with Pytorch [50], using the analytical form of the objective function for the purpose of efficiency.

Optimizers The QNN instances are trained with three popular optimizers in classical optimization or machine learning: Adam[51], RMSProp[52], and L-BFGS[53]. The first two methods [51, 52] are variants of vanilla gradient descent with adaptive learning rate. and are widely used for training large-scale deep neural networks as well as for the quantum counterparts [20, 22, 54, 55,

56]. The last method [53] is an efficient implementation of the approximate Newton method that utilizes the second-order information (i.e. the Hessian). For all instances and optimizers, we use the exact gradient induced by the dataset without stochasticity from the mini-batched gradient descent.

It turns out, for all the examined instances and all three optimizers, under random initialization, the optimizations converge to local minima with non-negligible suboptimality (i.e., different from the global one by a non-negligible amount) with high probability. In Figure 2.1, we train the 4-parameter construction with RMSProp and repeat for 100 times. Let $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_f$ denote the parameters at initialization and at convergence. The function values at initialization $L(\boldsymbol{\theta}_i; \mathcal{S})$ are supported on a continuous spectrum as shown in gray. After training and converging with RMSProp, the function values $L(\boldsymbol{\theta}_f; \mathcal{S})$ fall into discretized values as shown in orange. The smallest training loss attainable in our construction is 0, therefore only the leftmost bar (to the left of the dotted **black** vertical line) corresponds to the global minimum. Namely, the success probability of converging to the global minimum is very small. A similar phenomenon persists for instances with more parameters and with different optimizers in Figure 2.2. As the number of bad local minima grows exponentially in our construction, the success probability should also in theory decay exponentially. This is empirically confirmed in Figure 2.3, where we illustrate the precise empirical success probability for all these tests. Moreover, as shown in Section 2.11.3, the tendency of exponential decay remains unchanged in the presence of label noises, indicating the robustness of our constructions.

Beyond the constructed datasets To demonstrate the generality of our results, we repeat the experiments for datasets with more practical significance: for p-parameter instances, we choose the input state to be a p-qubit encoding of $\mathbf{x} \in [0, 2\pi)^{2p}$ via $\mathbf{X}$- and $\mathbf{Y}$-rotations on each of the

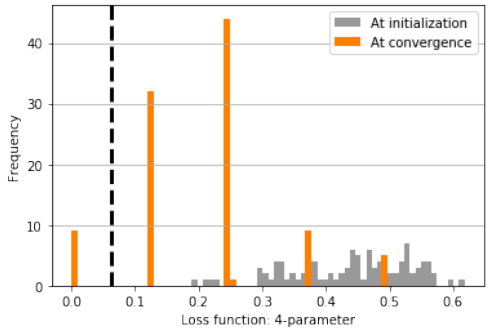


Figure 2.1: Loss functions at random initialization and at convergence for 4-parameter instances trained with RMSProp, repeated for 100 times. The function values are supported on a continuous spectrum at initialization as plotted in **gray** and converge to discretized values as plotted in **orange**.

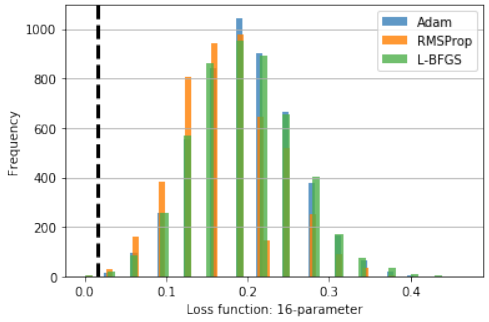


Figure 2.2: Distributions of loss functions at convergence for instances with 16 parameters trained with Adam, RMSProp and L-BFGS, repeated 5000 times with uniformly random initialization. All methods fail to converge to the global minimum 0.0 with high probability.

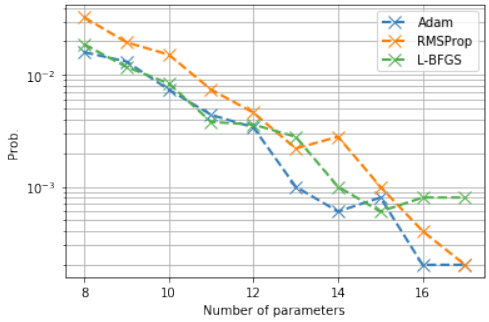


Figure 2.3: The decay of success rate for finding the global minimum under random initialization with Adam, RMSProp, L-BFGS. For each data point, we repeat the experiments for 5000 times.

qubits. The associated label is either 1 or 0, depending on the sign of $\mathbf{w}^T\mathbf{x}$, with $\mathbf{w}$ being the normal vector of a hyperplane in $\mathbb{R}^{2p}$. These datasets have the interpretation as an encoding of a linearly separable classical concept. In Figure2.4, we plot the function values at convergence for an 8-parameter instance: no more than 4 of the 70 random initializations have reached the global minima. This is repeated for instances with $2, 4$ and 6 qubits. While we no longer have a clear exponential dependency in the success rate, the number of local minima increases significantly as the number of parameters increases (see Section 2.11.4). This observation suggests that our theory and experiments on the constructed datasets can capture the practical difficulty in training under-parameterized QNNs with gradient-based methods.

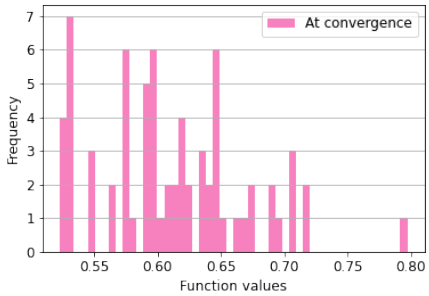


Figure 2.4: Function values at convergence for training an 8-parameter instance with RMSProp on the linearly-separable classical concept. No more than 4 among the 70 random initializations find the global minima, indicating the existence of many sub-optimal local minma.

2.7 Conclusion

In this work, we provide a characterization of the landscape for under-parameterized QNNs, by showing that in the worst-case, the number of local minima can increase exponentially with the number of parameters. Supported by numerical simulations, our result suggests when under-parameterized, QNNs may not be efficiently solved by gradient-based black-box methods.

This work leaves several open questions:

- Given the knowledge of the data distribution, can we design a QNN architecture with a benign landscape?
- We know that when sufficiently parameterized (e.g. [43]), the landscape for optimizing variational quantum ansatz can be benign. It is therefore natural to ask, fixing the system size, how does the landscape change as the number of parameters increases? This question has been later answered by [57].
- Classically, despite the provable bad landscape of shallow neural networks(e.g. [24]), [35] came up with algorithms that can minimize the loss with guarantees. Can we design an algorithm (beyond gradient-based method) that can solve the optimization problem efficiently and provably?

2.8 Proofs for Constructions

In this section, we provide the detailed proof for the existence of constructions appeared in Section 2.3.

We start by recalling the definitions of $\Phi_l^{(j)}(\cdot)$ and summarize some useful facts about these linear maps in Section 2.8.1, as well as the observables in the Heisenberg picture. In Section 2.8.2 we elaborate on our result on the existence of hard datasets for p-parameter QNNs with linear independence, which leads to Theorem 2.1.

In Section 2.8.3 we identify a family of 1-layer QNNs with linear independence, which gives rise to Proposition 2.3.1. In addition, we instantiated concrete datasets to illustrate the generality of our construction.

2.8.1 Linear Maps $\Phi_l^{(j)}(\cdot)$

Recall the definitions of $\Phi_l^{(j)}$: For all $l \in [p]$, define $\Phi_l^{(j)}$ such that for any Hermitian $\mathbf{A}$:

$$\Phi_l^{(0)}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} + \mathbf{H}_l\mathbf{A}\mathbf{H}_l) \tag{2.24}$$

$$\Phi_l^{(1)}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} - \mathbf{H}_l\mathbf{A}\mathbf{H}_l) \tag{2.25}$$

$$\Phi_l^{(2)}(\mathbf{A}) = \frac{i}{2}[\mathbf{H}_l, \mathbf{A}] \tag{2.26}$$

The subscript l will be dropped for general $\mathbf{H}$.

It can be easily verified that these mappings maps Hermitians to Hermitians, and the traces of the output:

$$\mathrm{tr}\left(\Phi_l^{(0)}(\mathbf{A})\right) = \mathrm{tr}\left(\frac{\mathbf{A} + \mathbf{H}_l\mathbf{A}\mathbf{H}_l}{2}\right) = \mathrm{tr}(\mathbf{A}) \tag{2.27}$$

$$\mathrm{tr}\left(\Phi_l^{(1)}(\mathbf{A})\right) = \mathrm{tr}\left(\frac{\mathbf{A} - \mathbf{H}_l\mathbf{A}\mathbf{H}_l}{2}\right) = 0 \tag{2.28}$$

$$\mathrm{tr}\left(\Phi_l^{(2)}(\mathbf{A})\right) = \mathrm{tr}\left(\frac{i[\mathbf{H}_l, \mathbf{A}]}{2}\right) = 0 \tag{2.29}$$

Adjoint The adjoints of the maps are:

$$\left(\Phi_l^{(0)}\right)^*(\mathbf{A}) = \Phi_l^{(0)}(\mathbf{A}) \tag{2.30}$$

$$\left(\Phi_l^{(1)}\right)^*(\mathbf{A}) = \Phi_l^{(1)}(\mathbf{A}) \tag{2.31}$$

$$\left(\Phi_l^{(2)}\right)^*(\mathbf{A}) = -\Phi_l^{(2)}(\mathbf{A}) \tag{2.32}$$

For all pairs of $j, k \in \{0, 1, 2\}$, we summarize the compostition of mappings $(\Phi_l^{(j)})^* \circ \Phi_l^{(k)}(\cdot)$ in

Table 2.1, and the inner products $\langle \Phi_l^{(j)}(\mathbf{A}), \Phi_l^{(k)}(\mathbf{A}) \rangle$ in Table 2.2.

Table 2.1: Composition $(\Phi_l^{(j)})^* \circ \Phi_l^{(k)}(\cdot)$

j\k	0	1	2
0	$\Phi_l^{(0)}(\cdot)$	0	0
1	0	$\Phi_l^{(1)}(\cdot)$	$\Phi_l^{(2)}(\cdot)$
2	0	$-\Phi_l^{(2)}(\cdot)$	$\Phi_l^{(1)}(\cdot)$

Table 2.2: Inner product $\langle \Phi_l^{(j)}(\mathbf{A}), \Phi_l^{(k)}(\mathbf{A}) \rangle$

j\k	0	1	2
0	$\langle \mathbf{A}, \Phi_l^{(0)}(\mathbf{A}) \rangle$	0	0
1	0	$\langle \mathbf{A}, \Phi_l^{(1)}(\mathbf{A}) \rangle$	0
2	0	0	$\langle \mathbf{A}, \Phi_l^{(1)}(\mathbf{A}) \rangle$

All off-diagonal elements are zero in Table 2.2, implying the orthogonality of $\{\Phi_l^{(j)}\}_{j=0,1,2}$. This follows from the fact

$$\langle \mathbf{A}, \Phi_l^{(2)}(\mathbf{A}) \rangle = \frac{i}{2}(\mathrm{tr}(\mathbf{AHA}) - \mathrm{tr}(\mathbf{A}^2\mathbf{H})) = 0 \tag{2.33}$$

We are now ready to prove the expansion of the observable in Heisenberg picture:

Claim 2.1 (Observable in Heisenberg Picture). For quantum neural networks defined in Theorem 2.1, the observable in Heisenberg picture $\mathbf{U}(\boldsymbol{\theta})^\dagger \mathbf{M} \mathbf{U}(\boldsymbol{\theta})$ can be expressed as:

$$\sum_{\boldsymbol{\xi} \in \{0,1,2\}^p} \Phi_{\boldsymbol{\xi}}(\mathbf{M}) \prod_{l:\xi_l=1} \cos 2\theta_l \prod_{l':\xi_{l'}=2} \sin 2\theta_{l'} \tag{2.34}$$

where $\Phi_{\boldsymbol{\xi}}$ is defined as the following composed mapping:

$$\Phi_1^{(\xi_1)} \circ \Phi_2^{(\xi_2)} \circ \cdots \circ \Phi_p^{(\xi_p)} \tag{2.35}$$

Proof. For a two-level Hamiltonian $\mathbf{H}$ with eigenvalues ± 1, let $\mathbf{P}_+$ and $\mathbf{P}_-$ be projections into subspaces of $\mathbb{C}^d$ corresponding to eigenvalues $+1$ and -1:

$$\mathbf{H} = \mathbf{P}_+ - \mathbf{P}_-, \text{ and } \mathbf{P}_+ + \mathbf{P}_- = \mathbf{I}; \tag{2.36}$$

For all $\theta \in \mathbb{R}$, the parameterized unitary

$$\exp(-i\theta\mathbf{H}) = e^{-i\theta}\mathbf{P}_+ + e^{i\theta}\mathbf{P}_- \tag{2.37}$$

$$= \cos\theta(\mathbf{P}_+ + \mathbf{P}_-) - \sin\theta i(\mathbf{P}_+ - \mathbf{P}_-) \tag{2.38}$$

$$= \cos\theta\mathbf{I} - i\sin\theta\mathbf{H} \tag{2.39}$$

By basic trigonometry, $e^{i\theta\mathbf{H}}\mathbf{A}e^{-i\theta\mathbf{H}}$ can be expressed as

$$\Phi^{(0)}(\mathbf{A}) + \Phi^{(1)}(\mathbf{A})\cos 2\theta + \Phi^{(2)}(\mathbf{A})\sin 2\theta \tag{2.40}$$

for any Hermitian $\mathbf{A}$. Claim 2.1 then follows from sequantial application of Eqn. (2.40). □

Under transformation $\boldsymbol{\theta} \mapsto \boldsymbol{\theta} + \boldsymbol{\alpha}$ for some $\boldsymbol{\alpha} \in \mathbb{R}^p$, the linear maps transform as:

$$\begin{cases} \Phi_l^{(0)}(\cdot) & \to \Phi_l^{(0)}(\cdot) \\ \Phi_l^{(1)}(\cdot) & \to \cos 2\alpha_l \Phi_l^{(1)}(\cdot) + \sin 2\alpha_l \Phi_l^{(2)}(\cdot) \\ \Phi_l^{(2)}(\cdot) & \to -\sin 2\alpha_l \Phi_l^{(1)}(\cdot) + \cos 2\alpha_l \Phi_l^{(2)}(\cdot) \end{cases} \tag{2.41}$$

This is because $e^{i(\theta+\alpha)\mathbf{H}}\mathbf{A}e^{-i(\theta+\alpha)\mathbf{H}}$ can be expressed as

$$\begin{aligned} &\Phi^{(0)}(\mathbf{A}) + (\cos 2\alpha \Phi^{(1)}(\mathbf{A}) + \sin 2\alpha \Phi^{(2)}(\mathbf{A})) \cos 2\theta \\ &\qquad + (-\sin 2\alpha \Phi^{(1)}(\mathbf{A}) + \cos 2\alpha \Phi^{(2)}(\mathbf{A})) \sin 2\theta \end{aligned} \tag{2.42}$$

As a consequence, each term of $f(\boldsymbol{\rho}; \boldsymbol{\theta})$ remain invariant under the joint transformation $\theta_l \mapsto \theta_l + \frac{\pi}{2}$ and

$$\Phi_l^{(0)}(\cdot) \mapsto \mathbf{H}_l \Phi_l^{(0)}(\cdot) \mathbf{H}_l = \Phi_l^{(0)}(\cdot) \tag{2.43}$$

$$\Phi_l^{(1)}(\cdot) \mapsto \mathbf{H}_l \Phi_l^{(1)}(\cdot) \mathbf{H}_l = -\Phi_l^{(1)}(\cdot) \tag{2.44}$$

$$\Phi_l^{(2)}(\cdot) \mapsto \mathbf{H}_l \Phi_l^{(2)}(\cdot) \mathbf{H}_l = -\Phi_l^{(2)}(\cdot) \tag{2.45}$$

2.8.2 Proof for Lemma 2.2 and 2.3

The construction for Theorem 2.1 consists of two steps. For the first step, dataset $\mathcal{S}_0$ is constructed with 2^p local minima invariant under the $\frac{\pi}{2}$ translational symmetry:

$$\theta_l \mapsto \theta_l + \frac{\pi}{2}. \tag{2.46}$$

Therefore the existence of a single local minimum $\boldsymbol{\theta}^\star$ indicates a set of local minima Θ. For the second step, we construct a data set $\mathcal{S}_1$ to break the symmetry. A combination of these two data set with proper scaling gives us a desired dataset for Theorem 2.1.

Here we provide the proof of Lemma 2.2 and Lemma 2.3 in details.

Lemma 2.6 (Creating symmetry)**.** *For QNNs with linear independence as mentioned in Theorem 2.1, a dataset $\mathcal{S}_0$ can be constructed to induce a loss function $L(\boldsymbol{\theta}; \mathcal{S}_0)$ that (1) has a local minimum at some $\boldsymbol{\theta}^\star$, and (2) is invariant under translation $\theta_l \mapsto \theta_l + \frac{\pi}{2}$ for all $l \in [p]$.*

Proof. It suffices to construct a dataset $\mathcal{S}_0 = \{(\boldsymbol{\rho}_k, y_k)\}_{k=1}^{m_0}$, such that (1) for all $k \in [p]$, $f_k(\boldsymbol{\theta}) := \langle \boldsymbol{\rho}_k, \mathrm{M}(\boldsymbol{\theta}) \rangle - y_k$ is either symmetric or anti-symmetric under $\theta_l \mapsto \theta_l + \frac{\pi}{2}$ for all $l \in [p]$, and (2) the intersection Θ of the set of roots Θ_k of $f_k(\boldsymbol{\theta}) = 0$ is non-empty and contains at least one isolated point $\boldsymbol{\theta}^\star$. For such $\mathcal{S}_0$, $\boldsymbol{\theta}^\star$ is an isolated root of the non-negative loss function $L(\boldsymbol{\theta}; \mathcal{S}_0) = \sum_{k=1}^{m_0} f_k(\boldsymbol{\theta})^2$.

For a concrete construction, consider $\{f_k(\boldsymbol{\theta})\}_{k=1}^{m_0}$ such that $f_k(\boldsymbol{\theta}) =\propto \sin(2\sum_{l=1}^{p} \eta_l^{(k)}(\theta_l - \theta_l^\star))$ for a set of vectors $\{\boldsymbol{\eta}^{(k)}\}_{k=1}^{m_0} \subseteq \{-1, 0, 1\}_{l=1}^{p}$ that spans $\mathbb{R}^p$, and arbitrary $\boldsymbol{\theta}^\star \in \mathbb{R}^p$.

The translational symmetry holds due to the periodicity of $\sin$-functions; the loss function has a minima at the vanishing point $\boldsymbol{\theta}^\star$.

To see that $\boldsymbol{\theta}^\star$ is indeed a isolated minima (that there exists a neighbourhood within which the loss function vanishes only at $\boldsymbol{\theta}^\star$), consider $\mathcal{N}(\boldsymbol{\theta}^\star) := \{\boldsymbol{\theta} \in \mathbb{R}^p : \forall k \in [m_0], |(\boldsymbol{\eta}^{(k)})^T(\boldsymbol{\theta} - \boldsymbol{\theta}^\star)| < \frac{\pi}{4}\}$. Conditioned on $\boldsymbol{\theta} \in \mathcal{N}(\boldsymbol{\theta}^\star)$,

$$L(\boldsymbol{\theta}; \mathcal{S}_0) = 0 \implies \forall k \in [m_0], \sum_{l=1}^{p} \eta_l^{(k)}(\theta_l - \theta_l^\star) = 0 \tag{2.47}$$

which then implies $\boldsymbol{\theta} = \boldsymbol{\theta}^\star$ since $\{\boldsymbol{\eta}_k^{(l)}\}_{l=1}^p$ spans $\mathbb{R}^p$.

The existence of such dataset $\mathcal{S}_0$ follows from the linear independence of operators for the QNN: for any $k \in [m_0]$, the solution set to the following linear system for Hermitian $\mathbf{D}_k \in \mathbb{C}^{d\times d}$ is non-empty:

$$\begin{cases} \langle \mathbf{D}_k, \mathbf{I} \rangle = 0, \\ \langle \mathbf{D}_k, \Phi_{\boldsymbol{\xi}}(\mathbf{M}) \rangle = \hat{f}_{\boldsymbol{\xi},k}, \ \forall \boldsymbol{\xi} \neq \mathbf{0}. \end{cases} \tag{2.48}$$

where $\mathbf{I}$ is the d-dimensional identity, and $\hat{f}_{\boldsymbol{\xi},k}$ denotes the coefficient corresponding to the term $\prod_{l:\xi_l=1} \cos 2\theta_l \prod_{l':\xi_{l'}=2} \sin 2\theta_{l'}$ in $\sin(2\sum_{l=1}^p \eta_l^{(k)}(\theta_l - \theta_l^\star))$.

As $\mathrm{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})) = 0$, $\mathbf{I}$ is orthogonal to all $\Phi_{\boldsymbol{\xi}}(\mathbf{M})$. Therefore the constraint set $\{\Phi_{\boldsymbol{\xi}}(\mathbf{M})\}_{\boldsymbol{\xi}\neq\mathbf{0}} \cup \{\mathbf{I}\}$ is linear independent. As a result, the linear system is guaranteed to have a set of solution $\{\mathbf{D}_k\}_{k=1}^{m_0}$.

Given the solution $\{\mathbf{D}_k\}_{k=1}^{m_0}$, the dataset can be constructed as:

$$\boldsymbol{\rho}_k := \frac{1}{d}\mathbf{I} + \kappa \mathbf{D}_k, \quad y_k = \mathrm{tr}(\boldsymbol{\rho}_k \Phi_{\mathbf{0}}(\mathbf{M})) \tag{2.49}$$

for all k, with κ be the largest positive real number such that $\frac{1}{d}\mathbf{I} + \kappa \mathbf{D}_k$ is positive semidefinite.

It can be verified by elementary calculation that such dataset yields a loss function $L(\boldsymbol{\theta}; \mathcal{S}_0) = \kappa \sum_{k=1}^{m_0} \sin(2(\boldsymbol{\eta}^{(k)})^T(\boldsymbol{\theta} - \boldsymbol{\theta}^\star))$. □

Lemma 2.7 (Breaking symmetry)**.** *Consider the QNN, dataset $\mathcal{S}_0$ and local minimum $\boldsymbol{\theta}^\star$ defined in Lemma 2.2. Let Θ denote the set of 2^p local minima due to the translational invariance. There exists a dataset $\mathcal{S}_1$ such that*

$$\inf_{\boldsymbol{\theta} \in \mathcal{N}(\boldsymbol{\theta}^\star)} L(\boldsymbol{\theta}; \mathcal{S}_0) + L(\boldsymbol{\theta}; \mathcal{S}_1) < \inf_{\boldsymbol{\theta} \in \mathcal{N}(\boldsymbol{\theta}')} L(\boldsymbol{\theta}; \mathcal{S}_0) + L(\boldsymbol{\theta}; \mathcal{S}_1) \tag{2.13}$$

for all $\boldsymbol{\theta}' \in \Theta / \{\boldsymbol{\theta}^\star\}$, and that

$$L(\boldsymbol{\theta}; \mathcal{S}_0) + L(\boldsymbol{\theta}; \mathcal{S}_1) > L(\boldsymbol{\theta}'; \mathcal{S}_0) + L(\boldsymbol{\theta}'; \mathcal{S}_1) \tag{2.14}$$

for all $\boldsymbol{\theta}' \in \Theta$ and all $\boldsymbol{\theta} \in \partial\mathcal{N}(\boldsymbol{\theta}')$. Here $\mathcal{N}(\cdot)$ denote a bounded and closed neighbourhood, such that $\mathcal{N}(\boldsymbol{\theta}) \cap \mathcal{N}(\boldsymbol{\theta}') = \emptyset$ for any $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta$; $\partial\mathcal{N}$ denotes its boundary.

Proof. Rewrite the loss function induced by $\mathcal{S}_1$ as:

$$L(\boldsymbol{\theta}; \mathcal{S}_1) = -\frac{2}{m_1} \sum_{k=1}^{m_1} \langle y_k \boldsymbol{\rho}_k, \mathbf{M}(\boldsymbol{\theta}) \rangle \tag{2.50}$$

$$+ \frac{1}{m_1} \sum_{k=1}^{m_1} \left(\langle \boldsymbol{\rho}_k, \mathbf{M}(\boldsymbol{\theta}) \rangle \right)^2 \tag{2.51}$$

$$+ \frac{1}{m_1} \sum_{k=1}^{m_1} y_k^2 \tag{2.52}$$

For any positive ϵ, consider the following joint scaling of $\boldsymbol{\rho}_k$ and y_k:

$$\begin{cases} \boldsymbol{\rho}_k & \mapsto \epsilon \boldsymbol{\rho}_k + \frac{1-\epsilon}{d}\mathbf{I} \\ y_k & \mapsto \frac{1}{\epsilon} y_k \end{cases} \tag{2.53}$$

Under such scaling, for arbitrary ϵ, term (2.50) remains the same; the term (2.51) can be arbitrarily suppressed by choosing sufficiently small ϵ. Therefore it suffices to consider the first term $L'(\cdot; \mathcal{S}_1) := -\frac{2}{m_1}\sum_{k=1}^{m_1} y_k \langle \boldsymbol{\rho}_k, \mathbf{M}(\boldsymbol{\theta})\rangle$.

Without loss of generality, assume $\boldsymbol{\theta}^\star = \mathbf{0}$. Consider a dataset $\mathcal{S}_1$ such that $L'(\boldsymbol{\theta}; \mathcal{S}_1) \propto -\sum_{k=1}^{m_1} \cos(2(\tilde{\boldsymbol{\eta}}^{(k)})^T \boldsymbol{\theta})$ for a set $\{\tilde{\boldsymbol{\eta}}^{(k)}\}_{k=1}^{m_1} \subset \{0,1\}^p$ that spans $\{0,1\}^p$.

For any $\boldsymbol{\zeta} \in \{0,1\}^p$,

$$L'(\boldsymbol{\theta}^\star + \frac{\pi}{2}\boldsymbol{\zeta}; \mathcal{S}_1) \propto -\sum_{k=1}^{m_1} \cos(2(\tilde{\boldsymbol{\eta}}^{(k)})^T \boldsymbol{\theta}^\star + \langle \tilde{\boldsymbol{\eta}}^{(k)}, \boldsymbol{\zeta}\rangle \pi) \propto -\sum_{k=1}^{m_1} (-1)^{\langle \tilde{\boldsymbol{\eta}}^{(k)}, \boldsymbol{\zeta}\rangle} \tag{2.54}$$

The fact that $\{\tilde{\boldsymbol{\eta}}^{(k)}\}_{k=1}^{m_1}$ spans $\{0,1\}^p$ indicate that the solution to

$$\forall k \in [m_1],\ \langle \tilde{\boldsymbol{\eta}}^{(k)}, \boldsymbol{\zeta}\rangle \equiv 0 \ (\mathsf{mod}\ 2) \tag{2.55}$$

is unique. Therefore such $\mathcal{S}_1$ breaks the $\pi/2$-translational symmetry as required. Similar to the proof of Lemma 2.2, the existence of such dataset follows from the linear independence of the operators.

For a concrete construction, let $\mathcal{S}_1$ be a dataset such that $L'(\boldsymbol{\theta}; \mathcal{S}_1) = -\frac{2}{m_1}\sum_{k=1}^{m_1} \langle y_k \boldsymbol{\rho}_k, \mathbf{M}(\theta)\rangle = -c\sum_{l=1}^{p} \cos(\theta_l)$. Due to the flexible scaling of the labels $\{y_k\}_{k=1}^{m_1}$, such $\mathcal{S}_1$ can be found for arbitrary c.

Let $\mathcal{B}_r(\boldsymbol{\theta})$ denote the closed ℓ_2-ball centered at $\boldsymbol{\theta}$ with radius r, and let $\partial\mathcal{B}_r$ denote its boundary. For the local minimum $\boldsymbol{\theta}^\star$ of $L(\boldsymbol{\theta}; \mathcal{S}_0)$, let (r, L_0) be a pair of real numbers such that $\inf_{\boldsymbol{\theta}\in\partial\mathcal{B}_r(\boldsymbol{\theta}^\star)} L(\boldsymbol{\theta}; \mathcal{S}_0) > L_0$ for some positive real number L_0.

The requirements in Lemma 2.3 is then met by choosing $c < \frac{L_0}{2pr^2}$. To see this we will make use of the estimation $1-\frac{1}{2}\theta_l^2 \leq \cos\theta_l \leq 1$. For any $\boldsymbol{\theta}' \in \Theta$, the loss function $L(\boldsymbol{\theta}; \mathcal{S}_0)+L'(\boldsymbol{\theta}; \mathcal{S}_1)$ evaluated at all points on the boundary of $B_r(\theta')$ is at least $\frac{L_0}{2}$ larger than $L(\boldsymbol{\theta}', \mathcal{S}_0) + L'(\boldsymbol{\theta}', \mathcal{S}_1)$. Therefore the second requirement in Lemma 2.3 is met.

For the first requirement, we have that (1) $L(\boldsymbol{\theta}^\star; \mathcal{S}_0)+L'(\boldsymbol{\theta}^\star; \mathcal{S}_0) = -pc$, and (2) for all $\boldsymbol{\theta}' \in \Theta/\{\boldsymbol{\theta}^\star\}$, for all $\boldsymbol{\theta} \in \mathcal{B}_r(\boldsymbol{\theta}')$, $L(\boldsymbol{\theta}; \mathcal{S}_0)+L'(\boldsymbol{\theta}; \mathcal{S}_0) > 0-(p-1)c+(1-\frac{1}{2}r^2)c = -pc+(2-\frac{1}{r^2})c$. The suboptimality gap is therefore at least $c(2-\frac{1}{2}r^2)$. □

Remarks. In the proof for Lemma 2.2 and 2.3, we made use of specific forms of sin- and cos-functions for the clarity of proof. However, the linear independence of the operators allow us to construct loss function beyond these specific forms, as will be made clear in Example 2 and 3.

2.8.3 Proof for Proposition 2.3.1 and Concrete Constructions

Proposition 2.3.1 (One-layer QNNs with product observables)**.** *Consider the family of QNNs composed of unitaries generated by two-level Hamiltonians, parameterized by $\boldsymbol{\theta} \in \mathbb{R}^p$. For all $l \in [p]$, let $\mathbf{H}_l$ be a local Hamiltonian on the l-qubit, taking the form $\mathbf{I}\otimes\cdots\otimes\mathbf{h}_l\otimes\cdots\otimes\mathbf{I}$ for some Hermitian $\mathbf{h}_l$ at the l-th position, and $\mathbf{M} = \mathbf{m}_1\otimes\cdots\otimes\mathbf{m}_p$ such that $\mathbf{m}_l + \mathbf{h}_l\mathbf{m}_l\mathbf{h}_l$ and $\mathbf{m}_l - \mathbf{h}_l\mathbf{m}_l\mathbf{h}_l$ are non-zero for any l. There exists a dataset that induces a loss function with 2^p-1 spurious local minima.*

Proof. Proposition 2.3.1 follows directly from the orthogonality of the operators. For any $\boldsymbol{\xi} \in$

$\{0,1,2\}^p$, the operator $\Phi_{\boldsymbol{\xi}}(\mathbf{M})$ can be expressed in the tensor product form $\otimes_{l=1}^{p}\tilde{\Phi}_l^{(\xi_l)}(\mathbf{m}_l)$. where $\tilde{\Phi}_l^{(j)}$ are linear maps associated with $\mathbf{h}_l$. For any $\boldsymbol{\xi}$ and $\boldsymbol{\xi}' \in \{0,1,2\}^p/\mathbf{0}$:

$$\langle \Phi_{\boldsymbol{\xi}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'}(\mathbf{M})\rangle = \prod_{l=1}^{p}\langle \tilde{\Phi}_l^{(\xi_l)}(\mathbf{m}_l), \tilde{\Phi}_l^{(\xi_l')}(\mathbf{m}_l)\rangle = \prod_{l=1}^{p}\|\tilde{\Phi}_l^{(\xi_l)}(\mathbf{m}_l)\|_F^2\delta_{\xi_l,\xi_l'} \tag{2.56}$$

Therefore $\{\Phi_{\boldsymbol{\xi}}(\mathbf{M})\}$ forms a linear independent set. □

We now move on to concrete constructions of datasets for QNNs defined in Proposition 2.3.1. To facilitate narrative, assume $\mathbf{M} = \mathbf{M}_0^{\otimes p}$, and $\mathbf{h}_l = \mathbf{H}_0$.

For $j = 0,1,2$, define $\mathbf{D}^{(j)}$ as $\Phi^{(j)}(\mathbf{M}_0)$. And let $\boldsymbol{\rho}^{(j)}$ be a proper linear combination of $\mathbf{D}^{(j)}$ and $\mathbf{I}$ that is positive semidefinite and trace-1.

For $l \in [p]$, define

$$\boldsymbol{\rho}_{0,l} = \left(\otimes_{r=1}^{l-1}\boldsymbol{\rho}^{(0)}\right) \otimes \boldsymbol{\rho}^{(1)} \otimes \left(\otimes_{r=l+1}^{p}\boldsymbol{\rho}^{(0)}\right) \tag{2.57}$$

$$\boldsymbol{\rho}_{1,l} = \left(\otimes_{r=1}^{l-1}\boldsymbol{\rho}^{(0)}\right) \otimes \boldsymbol{\rho}^{(2)} \otimes \left(\otimes_{r=l+1}^{p}\boldsymbol{\rho}^{(0)}\right) \tag{2.58}$$

$$y_{0,l} = 0 \tag{2.59}$$

$$y_{1,l} = \operatorname{tr}(\rho^{(0)}\Phi^{(0)}(\mathbf{M}_0))^{p-1}\operatorname{tr}(\rho^{(1)}\Phi^{(1)}(\mathbf{M}_0)) \tag{2.60}$$

Let $\mathcal{S}_0 = \{(\boldsymbol{\rho}_{0,l}, y_{0,l})\}_{l=1}^{p}$ be the dataset with $\frac{\pi}{2}$-translational symmetry, with the resulting loss function proportional to $\sum_{l=1}^{p}\sin\theta_l^2$; Let $\mathcal{S}_1 = \{(\boldsymbol{\rho}_{1,l}, y_{1,l})\}_{l=1}^{p}$ be the dataset that breaks the symmetry, with the loss function proportional to $\sum_{l=1}^{p}(\cos\theta_l - 1)^2$.

Example 1. For a concrete example, consider $\mathbf{M}_0 = \mathbf{Y}+\mathbf{I}$, and $\mathbf{H}_0 = \mathbf{Z}$. Choose $\boldsymbol{\rho}^{(0)} = \frac{1}{2}(\mathbf{Z}+\mathbf{I})$, $\boldsymbol{\rho}^{(1)} = \frac{1}{2}(\mathbf{X}+\mathbf{I})$, $\boldsymbol{\rho}^{(2)} = \frac{1}{2}(\mathbf{Y}+\mathbf{I})$. Construct the dataset $\mathcal{S}_0$ and $\mathcal{S}_1$ as described above, and let the dataset $\mathcal{S}$ be the combination of $\mathcal{S}_0$ and $\mathcal{S}_1$ with reweighting factor $4:1$. The loss function

takes the form:

$$\frac{1}{2p}\sum_{l=1}^{p}\sin^2(2\theta_l)+\frac{1}{4}(\cos 2\theta_l-1)^2 \tag{2.61}$$

For clarity, the construction in Example 1 was purposefully designed with two limitations. First of all, the construction has a fixed global minima at $\boldsymbol{\theta}^\star=\mathbf{0}$. Also, the loss function of the construction in Example 1 can be decomposed into p single-parameter functions. Therefore the training problem can be solved by greedily optimizing each of the coordinate θ_l.

To address the first limitation, we propose the following example:

Example 2. For

$$\boldsymbol{\rho}_{0,l}=\left(\otimes_{r=1}^{l-1}\boldsymbol{\rho}^{(0)}\right)\otimes\boldsymbol{\rho}^{(1)}\otimes\left(\otimes_{r=l+1}^{p}\boldsymbol{\rho}^{(0)}\right) \tag{2.62}$$

$$\boldsymbol{\rho}_{1,l}=\left(\otimes_{r=1}^{l-1}\boldsymbol{\rho}^{(0)}\right)\otimes\boldsymbol{\rho}^{(2)}\otimes\left(\otimes_{r=l+1}^{p}\boldsymbol{\rho}^{(0)}\right) \tag{2.63}$$

$$y_{0,l}=\sin(\frac{\pi}{50}),\quad y_{1,l}=\cos(\frac{\pi}{50}) \tag{2.64}$$

Construct dataset $\mathcal{S}_0$, $\mathcal{S}_1$ as:

$$\mathcal{S}_0=\{(\boldsymbol{\rho}_{0,l},y_{0,l})\}_{l=1}^{p},\quad \mathcal{S}_1=\{(\boldsymbol{\rho}_{1,l},y_{1,l})\}_{l=1}^{p} \tag{2.65}$$

and let the dataset $\mathcal{S}$ be the combination of $\mathcal{S}_0$ and $\mathcal{S}_1$ with reweighting factor $4:1$. The loss function takes the form:

$$\frac{1}{2p}\sum_{l=1}^{p}\left((\sin 2\theta_l-\sin\frac{\pi}{50})^2+\frac{1}{4}(\cos 2\theta_l-\cos\frac{\pi}{50})^2\right). \tag{2.66}$$

The resulting loss function has a local minimum at $(\frac{\pi}{100}, \cdots, \frac{\pi}{100})^T$.

To address the decomposability issue, consider the following example:

Example 3. Let $\mathcal{S}_0, \mathcal{S}_1$ be as defined in Example 2. Let $\boldsymbol{\rho}_{2,l,k}$ denote a product state with $\boldsymbol{\rho}^{(1)}$ for the k-th and l-th qubits, and $\boldsymbol{\rho}^{(0)}$ for the rest. The loss function for training sample $(\boldsymbol{\rho}_{2,l,k}, \cos^2 \frac{\pi}{50})$ is $(\cos 2\theta_l \cos 2\theta_k - \cos^2 \frac{\pi}{50})^2$. Combining this additional term with $\mathcal{S}_0$ and $\mathcal{S}_1$ gives rise to non-decomposable loss functions that cannot be solved by optimizing each coordinate independently.

As will be seen in Section 2.11, the construction of Example 2 and 3 indicates that our construction method is general, and can lead to instances that are hard to optimize with gradient-based methods and do not admit other trivial optimization methods.

2.9 Proof for Typical QNNs with Linear Dependence

In this section, we show that typical under-parameterized QNNs are with linear independence. To that end, we consider random d-dimensional p-parameter QNNs sampled with respect to the following measure:

Let $\mathbf{H}$ be a d-dimension Hermitian such that $\mathrm{tr}(\mathbf{H}) = 0$ and $\mathbf{H}^2 = \mathbf{I}$. The circuit for our random QNN is:

$$\mathbf{U}(\boldsymbol{\theta}) = e^{-i\theta_p \mathbf{W}_p \mathbf{H} \mathbf{W}_p^\dagger} \cdots e^{-i\theta_1 \mathbf{W}_1 \mathbf{H} \mathbf{W}_1^\dagger} \tag{2.67}$$

with $\{\mathbf{W}_l\}_{l=1}^p$ independently sampled with respect to the Haar measure on the d-dimensional unitary group $U(d)$.

Following from the fact $e^{-i\theta \mathbf{W}\mathbf{H}\mathbf{W}^\dagger} = \mathbf{W} e^{-i\theta \mathbf{H}} \mathbf{W}^\dagger$ for Hermitian $\mathbf{H}$ and unitary $\mathbf{W}$, we

can rewrite Eqn. (2.67) as:

$$\mathbf{U}(\boldsymbol{\theta}) = \mathbf{W}_p e^{-i\theta_p \mathbf{H}} \left(\mathbf{W}_p^\dagger \mathbf{W}_{p-1}\right) \cdots \left(\mathbf{W}_2^\dagger \mathbf{W}_1\right) e^{-i\theta_1 \mathbf{H}} \mathbf{W}_1^\dagger \tag{2.68}$$

Due to the left (and right) invariance of the Haar measure, up to a unitary transformation, the random model in Eqn. (2.67) is equivalent to a circuit with p interleaving parameterized gate $\{e^{-i\theta_l \mathbf{H}}\}_{l=1}^p$ and unitary $\{\tilde{\mathbf{W}}_l\}_{l=1}^p$ randomly sampled with respect to the Haar measure:

$$\mathbf{U}(\boldsymbol{\theta}) = \tilde{\mathbf{W}}_p e^{-i\theta_p \mathbf{H}} \tilde{\mathbf{W}}_{p-1} \cdots \tilde{\mathbf{W}}_1 e^{-i\theta_1 \mathbf{H}} \tag{2.69}$$

This interleaving nature of fixed and parameterized gates is shared by existing designs of QNNs, and any p-parameter QNN generated by two-level Hamiltonians can be expressed in Eqn. (2.69). Morever, applying polynomially many random 2-qubit gates on random pairs of qubits generates a distribution over gates that approximates the Haar measure up to the 4-th moments [45], which is what we require in the proof in this section.

In the rest of this section, we provide detailed proof for Theorem 2.4.

Theorem 2.4 (Typical under-parameterized QNNs are with linear independence)**.** *Consider a random p-parameter d-dimensional QNN with two-level Hamiltonians sampled from the model specified in Eqn. (2.17). Let the observable $\mathbf{M}$ be an arbitrary non-zero trace-0 Hermitian. Such QNN is with linear independence with probability $\geq 1 - O(d^{-1})$ for fixed p, and with probability $\geq 1 - O(e^{-p})$ for dimension $d : \log(d) = \Theta(p)$.*

2.9.1 Proof of Theorem 2.4

Proof. Let $\Xi = \{0,1,2\}^p/\{\mathbf{0}\}$ denote the set of all $\boldsymbol{\xi} \in \{0,1,2\}^p$ except for $\boldsymbol{\xi} = (0,\cdots,0)^T$. Our goal is to show that $\{\Phi_{\boldsymbol{\xi}}\}_{\xi\in\Xi}$ is linearly independent with high probability.

To show the linear independence of $\{\Phi_{\boldsymbol{\xi}}\}$, it suffices to show that its Gram matrix is positive semidefinite (Theorem 7.2.10 in [58]). The Gram matrix $\mathbf{G}$ is defined such that the $(\boldsymbol{\xi},\boldsymbol{\xi}')$-element $G_{\boldsymbol{\xi},\boldsymbol{\xi}'} := \langle \Phi_{\xi}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'}(\mathbf{M})\rangle$, for all pairs of $\boldsymbol{\xi},\boldsymbol{\xi}' \in \Xi$.

By the Gershgorin circle theorem [47], it suffices to show that with high probability

$$\langle \Phi_{\boldsymbol{\xi}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}}(\mathbf{M})\rangle > \sum_{\boldsymbol{\xi}'\in\Xi,\boldsymbol{\xi}'\neq\boldsymbol{\xi}} |\langle \Phi_{\boldsymbol{\xi}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'}(\mathbf{M})\rangle| \tag{2.70}$$

for all $\boldsymbol{\xi} \in \Xi$.

Using the Chebyshev inequality with the moment estimations in Lemma 2.8, we have:

$$\Pr\left[\operatorname{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})^2) < \frac{2}{3}\frac{\operatorname{tr}(\mathbf{M}^2)}{2^p}\right] \leq O(d^{-1}) \tag{2.71}$$

and

$$\Pr\left[|\operatorname{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'}(\mathbf{M}))| > \frac{1}{3\cdot 3^p}\frac{\operatorname{tr}(\mathbf{M}^2)}{2^p}\right] \leq O(3^p d^{-1}) \tag{2.72}$$

for $\boldsymbol{\xi} \neq \boldsymbol{\xi}'$. Combined with the union bound, we can show that:

$$\Pr\left(\exists\boldsymbol{\xi} \in \Xi : \langle \Phi_{\boldsymbol{\xi}}(\mathbf{M})\Phi_{\boldsymbol{\xi}}(\mathbf{M})\rangle \leq \sum_{\boldsymbol{\xi}'\in\Xi:\boldsymbol{\xi}'\neq\boldsymbol{\xi}} |\langle \Phi_{\boldsymbol{\xi}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'}(\mathbf{M})\rangle|\right) \leq O(e^p d^{-1}) \tag{2.73}$$

□

2.9.2 Moments

Lemma 2.8 (Expectations and variances)**.** *Consider the set of operators* $\{\Phi_{\boldsymbol{\xi}}(\mathbf{M})\}$ *of random* d*-dimensional* p*-parameter QNNs defined in Eqn. (2.67). The expectations of diagonal and off-diagonal terms of the associated Gram matrix are:*

$$\mathbb{E}[\mathrm{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})\Phi_{\boldsymbol{\xi}}(\mathbf{M}))] = \frac{\mathrm{tr}(\mathbf{M}^2)}{2^p}(1 + O(pd^{-2})) \tag{2.74}$$

$$\mathbb{E}[\mathrm{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'}(\mathbf{M}))] = 0 \tag{2.75}$$

$$\tag{2.76}$$

for all $\boldsymbol{\xi} \in \Xi$ *and* $\boldsymbol{\xi}' \neq \boldsymbol{\xi}$*. The variances are:*

$$\mathbb{V}[\mathrm{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'}(\mathbf{M}))] = \frac{\mathrm{tr}(\mathbf{M}^2)^2}{4^p}O(d^{-1}) \tag{2.77}$$

for all $\boldsymbol{\xi}, \boldsymbol{\xi}' \in \Xi$*.*

Proof. Throughout the proof, we will use $\mathbb{E}_l[\cdot]$ to denote expectation with respect to $\mathbf{H}_l$, and use $\mathbb{E}_{l:p}[\cdot]$ to denote integral over the product measure over $\mathbf{W}_l, \cdots, \mathbf{W}_p$. The subscripts will be dropped when there is no confusion.

We start by showing some basic (in-)equalities using the formula for integrals over the Haar measure [46]. Recall that $\mathbf{H}_l = \mathbf{W}_l \mathbf{H} \mathbf{W}_l^\dagger$. Let $\mathbf{A}$ and $\mathbf{B}$ be two Hermitian matrices such that

$\mathrm{tr}(\mathbf{A}) = \mathrm{tr}(\mathbf{B}) = 0$. For integrals where $\mathbf{H}_l$ appears once:

$$\mathbb{E}_l[\mathrm{tr}(\mathbf{A}\mathbf{H}_l)] = \frac{1}{d}\,\mathrm{tr}(\mathbf{A})\,\mathrm{tr}(\mathbf{H}) = 0 \tag{2.78}$$

For integrals where $\mathbf{H}_l$ appears twice, we have:

$$\mathbb{E}_l[\mathrm{tr}(\mathbf{A}\mathbf{H}_l)\,\mathrm{tr}(\mathbf{B}\mathbf{H}_l)] \tag{2.79}$$

$$= \frac{\mathrm{tr}(\mathbf{A}\mathbf{B})\,\mathrm{tr}(\mathbf{H}^2) + \mathrm{tr}(\mathbf{A})\,\mathrm{tr}(\mathbf{B})\,\mathrm{tr}(\mathbf{H})^2}{d^2-1} \tag{2.80}$$

$$- \frac{\mathrm{tr}(\mathbf{A}\mathbf{B})\,\mathrm{tr}(\mathbf{H})^2 + \mathrm{tr}(\mathbf{A})\,\mathrm{tr}(\mathbf{B})\,\mathrm{tr}(\mathbf{H}^2)}{d(d^2-1)} \tag{2.81}$$

$$= \frac{1}{d-d^{-1}}\,\mathrm{tr}(\mathbf{A}\mathbf{B}) \tag{2.82}$$

$$\mathbb{E}_l[\mathrm{tr}(\mathbf{A}\mathbf{H}_l\mathbf{B}\mathbf{H}_l)] \tag{2.83}$$

$$= \frac{\mathrm{tr}(\mathbf{A}\mathbf{B})\,\mathrm{tr}(\mathbf{H})^2 + \mathrm{tr}(\mathbf{A})\,\mathrm{tr}(\mathbf{B})\,\mathrm{tr}(\mathbf{H}^2)}{d^2-1} \tag{2.84}$$

$$- \frac{\mathrm{tr}(\mathbf{A}\mathbf{B})\,\mathrm{tr}(\mathbf{H}^2) + \mathrm{tr}(\mathbf{A})\,\mathrm{tr}(\mathbf{B})\,\mathrm{tr}(\mathbf{H})^2}{d(d^2-1)} \tag{2.85}$$

$$= -\frac{1}{d^2-1}\,\mathrm{tr}(\mathbf{A}\mathbf{B}) \tag{2.86}$$

For integrals with $\mathbf{H}_l$ appearing 4 times, we use the following estimation

$$|\mathbb{E}[\mathrm{tr}(\mathbf{A}\mathbf{H}_l\mathbf{A}\mathbf{H}_l)\,\mathrm{tr}(\mathbf{B}\mathbf{H}_l\mathbf{B}\mathbf{H}_l)]| \tag{2.87}$$

$$=O(d^{-4})\max\{|\,\mathrm{tr}(\mathbf{H}^2)|^2, |\,\mathrm{tr}(\mathbf{H}^4)|\} \tag{2.88}$$

$$\cdot\max\{|\,\mathrm{tr}(\mathbf{A}^2\mathbf{B}^2)|, |\,\mathrm{tr}(\mathbf{A}\mathbf{B}\mathbf{A}\mathbf{B})|, |\,\mathrm{tr}(\mathbf{A}\mathbf{B})^2|, |\,\mathrm{tr}(\mathbf{A}^2)\,\mathrm{tr}(\mathbf{B}^2)|\} \tag{2.89}$$

$$=O(d^{-2})\,\mathrm{tr}(\mathbf{A}^2)\,\mathrm{tr}(\mathbf{B}^2) \tag{2.90}$$

Here the first relation follows from formula on Equation (3) in [46] and expressions in Sec. 6 of [59]); and the second relation follows from matrix Cauchy-Schwarz inequalities and trace inequalities from [60, 61].

Similarly we have: $|\mathbb{E}[\mathrm{tr}(\mathbf{A}\mathbf{H}_l\mathbf{B}\mathbf{H}_l)\,\mathrm{tr}(\mathbf{A}\mathbf{H}_l\mathbf{B}\mathbf{H}_l)]| = O(d^{-2})\,\mathrm{tr}(\mathbf{A}^2)\,\mathrm{tr}(\mathbf{B}^2)$.

First moments We start by calculating the first moments of $\langle\Phi_{\boldsymbol{\xi}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}}(\mathbf{M})\rangle$ and $\langle\Phi_{\boldsymbol{\xi}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'}(\mathbf{M})\rangle$.

$$\mathbb{E}\langle\Phi_{\boldsymbol{\xi}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'}(\mathbf{M})\rangle \tag{2.91}$$

$$=\mathbb{E}\langle\Phi_1^{(\xi_1)}\circ\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}), \Phi_1^{(\xi_1')}\circ\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\rangle \tag{2.92}$$

$$=\mathbb{E}\langle\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}), (\Phi_1^{(\xi_1)})^*\circ\Phi_1^{(\xi_1')}\circ\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\rangle \tag{2.93}$$

By the basic results on the adjoints in Section 2.8.1, $(\Phi_l^{(\xi_l)})^*\circ\Phi_l^{(\xi_l')}(\cdot) = \Phi_l^{(0)}(\cdot)$ (or $\Phi_l^{(1)}(\cdot)$) if $\xi_l = \xi_l' = 0$ (or $\xi_l = \xi_l' \neq 0$). In the case $\xi_l \neq \xi_l'$, if ξ_l and ξ_l' are both in $\{1, 2\}$, $(\Phi_l^{(\xi_l)})^*\circ\Phi_l^{(\xi_l')}(\cdot) = \pm\Phi_l^{(2)}(\cdot)$. Otherwise $(\Phi_l^{(\xi_l)})^*\circ\Phi_l^{(\xi_l')}(\cdot) = 0$. We treat these three cases separately.

Case 1: $\xi_1 = \xi_1'$**.**

$$\text{Line (2.91)} = \mathbb{E}\langle \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}), \Phi_1^{(0/1)} \circ \Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M}) \rangle \tag{2.94}$$

$$= \frac{1}{2}\mathbb{E}_{2:p}\langle \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M}) \rangle \tag{2.95}$$

$$\pm \frac{1}{2}\mathbb{E}\langle \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}), \mathbf{H}_1 \Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M}) \mathbf{H}_1 \rangle \tag{2.96}$$

$$= \frac{1}{2}(1 \mp \frac{1}{d^2-1})\mathbb{E}_{2:p}\langle \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M}) \rangle \tag{2.97}$$

The sign of the second term depends on whether ξ_1 is 0 or $1, 2$. Therefore the first moment of the Frobenius norm of $\Phi_{\boldsymbol{\xi}}(\mathbf{M})$:

$$\mathbb{E}\langle \Phi_{\boldsymbol{\xi}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}}(\mathbf{M}) \rangle \tag{2.98}$$

$$= \frac{1}{2}(1 \pm \frac{1}{d^2-1})\mathbb{E}_{2:p}\langle \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}) \rangle \tag{2.99}$$

$$= (1 + O(pd^{-2}))\frac{\operatorname{tr}(\mathbf{M}^2)}{2^p} \tag{2.100}$$

Case 2: $\xi_1, \xi_1' \in \{1, 2\}$ **and** $\xi_1 \neq \xi_1'$**.**

$$\text{Line (2.91)} = \pm\, \mathbb{E}\langle \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}), \Phi_1^{(2)} \circ \Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M}) \rangle \tag{2.101}$$

$$= \pm\, \frac{i}{2}\mathbb{E}_{2:p}\langle \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}), \tag{2.102}$$

$$\mathbf{H}_1 \Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M}) - \Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M}) \mathbf{H}_1 \rangle \tag{2.103}$$

$$= \pm\, \frac{i}{2}\big(\operatorname{tr}(\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}))\operatorname{tr}(\mathbf{H}) \tag{2.104}$$

$$- \operatorname{tr}(\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M}))\operatorname{tr}(\mathbf{H})\big) \tag{2.105}$$

$$= 0 \tag{2.106}$$

Case 3: Either ξ_1 or $\xi_1' = 0$ and $\xi_1 \neq \xi_1'$.

$$\text{Line } (2.91) = \mathbb{E}\langle \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M}), 0 \circ \Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\rangle = 0 \tag{2.107}$$

Combining Case 2 and Case 3, $\mathbb{E}\langle \Phi_{\boldsymbol{\xi}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'}(\mathbf{M})\rangle = 0$ for $\boldsymbol{\xi} \neq \boldsymbol{\xi}'$.

Second moments The correlation between the square of Frobenius norm can calculated recursively as:

$$\mathbb{E}[\|\Phi_{\boldsymbol{\xi}_{1:p}}(\mathbf{M})\|_F^2 \cdot \|\Phi_{\boldsymbol{\xi}'_{1:p}}(\mathbf{M})\|_F^2] \tag{2.108}$$

$$= \mathbb{E}[\langle \Phi_{\boldsymbol{\xi}_{1:p}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}_{1:p}}(\mathbf{M})\rangle \tag{2.109}$$

$$\langle \Phi_{\boldsymbol{\xi}'_{1:p}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'_{1:p}}(\mathbf{M})\rangle] \tag{2.110}$$

$$= \mathbb{E}\langle \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\Phi_l^{(0/1)} \circ \Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\rangle \tag{2.111}$$

$$\cdot \langle \Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\Phi_l^{(0/1)} \circ \Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\rangle \tag{2.112}$$

$$= \frac{1}{4}\Big\{\mathbb{E}[\|\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\|_F^2 \cdot \|\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\|_F^2] \tag{2.113}$$

$$\pm \mathbb{E}[\mathrm{tr}(\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\mathbf{H}_l\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\mathbf{H}_l)\|\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\|_F^2] \tag{2.114}$$

$$\pm \mathbb{E}[\|\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\|_F^2 \,\mathrm{tr}(\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\mathbf{H}_l\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\mathbf{H}_l)] \tag{2.115}$$

$$\pm \mathbb{E}[\mathrm{tr}(\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\mathbf{H}_l\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\mathbf{H}_l) \tag{2.116}$$

$$\cdot \mathrm{tr}(\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\mathbf{H}_l\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\mathbf{H}_l)]\Big\} \tag{2.117}$$

$$= \frac{1 + O(d^{-2})}{4}\mathbb{E}[\|\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\|_F^2 \cdot \|\Phi_{\boldsymbol{\xi}'_{2:p}}(\mathbf{M})\|_F^2] \tag{2.118}$$

Therefore the diagonal elements of the Gram matrix has second moments:

$$\mathbb{E}[\|\Phi_{\boldsymbol{\xi}_{1:p}}(\mathbf{M})\|_F^2 \cdot \|\Phi_{\boldsymbol{\xi}_{1:p}}(\mathbf{M})\|_F^2] \tag{2.119}$$

$$=\frac{1+O(d^{-2})}{4}\mathbb{E}[\|\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\|_F^2 \cdot \|\Phi_{\boldsymbol{\xi}_{2:p}}(\mathbf{M})\|_F^2] \tag{2.120}$$

$$=(1+O(pd^{-2}))\frac{\operatorname{tr}(\mathbf{M}^2)^2}{4^p} \tag{2.121}$$

We are now ready to calculate the second moments for the off-diagonal elements of the Gram matrix. For $\boldsymbol{\xi}, \boldsymbol{\xi} \in \Xi$:

$$\mathbb{E}\langle\Phi_{\boldsymbol{\xi}_{l:p}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'_{l:p}}(\mathbf{M})\rangle^2 \tag{2.122}$$

$$=\mathbb{E}\langle\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M}), (\Phi_l^{\xi_l})^* \circ \Phi_l^{\xi'_l} \circ \Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\rangle^2 \tag{2.123}$$

Case 1: $\xi_l = \xi'_l$**.**

$$(2.122) \tag{2.124}$$

$$=\mathbb{E}\langle\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M}), \Phi_l^{(0/1)} \circ \Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\rangle^2 \tag{2.125}$$

$$=\frac{1}{4}\mathbb{E}\big[\operatorname{tr}(\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M}))^2 \tag{2.126}$$

$$\pm 2\operatorname{tr}(\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})) \tag{2.127}$$

$$\cdot \operatorname{tr}(\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\mathbf{H}_l\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\mathbf{H}_l) \tag{2.128}$$

$$+\operatorname{tr}(\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\mathbf{H}_l\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\mathbf{H}_l)^2\big] \tag{2.129}$$

$$=\frac{1}{4}\{(1\pm\frac{2}{d^2-1})\mathbb{E}\big[\operatorname{tr}(\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M}))^2\big] \tag{2.130}$$

$$+O(d^{-2})\mathbb{E}[\|\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\|_F^2 \cdot \|\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\|_F^2]\} \tag{2.131}$$

Case 2: $\xi_l, \xi_l' \in \{1, 2\}$ and $\xi_l \neq \xi_l'$. Up to a sign flip, we have

$$
\begin{aligned}
& (2.122) && (2.132)\\
=&\mathbb{E}\langle \Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M}), \Phi_l^{(2)} \circ \Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\rangle^2 && (2.133)\\
=&-\frac{1}{4}\mathbb{E}[\langle \Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M}), && (2.134)\\
& \qquad \mathbf{H}_l \Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M}) - \Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\mathbf{H}_l\rangle^2] && (2.135)\\
=&-\frac{1}{4}\mathbb{E}\{\operatorname{tr}(\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\mathbf{H}_l)^2 && (2.136)\\
&+\operatorname{tr}(\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\mathbf{H}_l)^2 && (2.137)\\
&-2\operatorname{tr}(\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\mathbf{H}_l) && (2.138)\\
&\cdot \operatorname{tr}(\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\mathbf{H}_l)\} && (2.139)\\
=&-\frac{1}{2(d-d^{-1})}\mathbb{E}\big[\operatorname{tr}((\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M}))^2) && (2.140)\\
&-\operatorname{tr}(\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})^2\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})^2)\big] && (2.141)\\
\leq&\frac{1}{d-d^{-1}}\mathbb{E}[\operatorname{tr}(\Phi_{\boldsymbol{\xi}_{l+1:p}}(\mathbf{M})^2)\operatorname{tr}(\Phi_{\boldsymbol{\xi}'_{l+1:p}}(\mathbf{M})^2)] && (2.142)
\end{aligned}
$$

Case 3: Either ξ_l or $\xi_l' = 0$ and $\xi_l \neq \xi_l'$. For the case where $\xi_l \neq \xi_l'$, and one of ξ_l and ξ_l' is 0, the correlation $\mathbb{E}\langle \Phi_{\boldsymbol{\xi}_{l:p}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'_{l:p}}(\mathbf{M})\rangle^2 = 0$.

Combining the above three cases, we have the variance bounded:

$$
\mathbb{V}(\langle \Phi_{\boldsymbol{\xi}}(\mathbf{M}), \Phi_{\boldsymbol{\xi}'}(\mathbf{M})\rangle) = O(d^{-1})\frac{\operatorname{tr}(\mathbf{M}^2)^2}{4^p} \tag{2.143}
$$

□

2.10 Proofs for Upper Bounds

For any p-parameter quantum circuit of consideration, we can express the circuit as:

$$\mathbf{U}(\boldsymbol{\theta}) = \mathbf{V}_p(\theta_p)\mathbf{V}_{p-1}(\theta_{p-1})\cdots\mathbf{V}_1(\theta_1), \tag{2.144}$$

where $\mathbf{V}_l(\theta_l) = \exp(-i\theta_l\mathbf{H}_l)$ for some Hermitian $\mathbf{H}_l$. We can bound the number of local minima in $L(\boldsymbol{\theta}; \mathcal{S})$ depending the choice of $\{\mathbf{H}_l\}_{l=1}^p$. In this section, we provide a proof to Theorem 2.5 in Section 2.5:

Theorem 2.5 (Upper bound: the number of local minima)**.** *Consider non-degenerated QNNs composed of unitaries generated by two-level Hamiltonians $\{H_l\}_{l=1}^p$ with p parameters. For training set $\mathcal{S}$, within each period, the loss function $L(\boldsymbol{\theta}; \mathcal{S})$ possesses at most $(4p)^p$ local minima.*

QNN with multi-qubit parameterized gates can have generating Hamiltonians $\{\mathbf{H}_l\}_{l=1}^p$ with more than two different eigenvalues. Specifically we consider Hamiltonians with integral eigenvalues $\{E_1, \cdots, E_d\}$ such that $\max_{c,c'} |E_c - E'_c| \leq \Delta$. We can generalize Theorem 2.5 as:

Theorem 2.9 (An upper bound for the more general setting)**.** *Consider p-parameter quantum neural networks composed of unitaries generated by Hamiltonians with integral spectrum gaps bounded by Δ. The loss function $L(\boldsymbol{\theta}; \mathcal{S})$ possesses at most $(4\Delta p)^p$ local minima, within each period, provided that the instance is not degenerated (i.e. the number of critical points is finite).*

Note that any Hamiltonians with rational eigenvalues are included with proper scaling and shifting.

In Section 2.10.1, we provide an upper bound on the Fourier degree of the loss function. In

Section 2.10.2, we bound the number of local minima for functions with bounded Fourier degree by considering the number of roots of a polynomial system.

2.10.1 Fourier Spectrum of the Loss Function

We first present a lemma on the Fourier spectrum of the loss function. For all $l \in [p]$, let $\{E_i^{(l)}\}_{i=1}^d$ be the integral eigenvalues for $\mathbf{H}_l$ and let Δ_l denote the largest eigen-gap in absolute value: $\Delta_l := \max_{i,j\in[d]} |E_i^{(l)} - E_j^{(l)}|$. For arbitrary choice of training set $\mathcal{S}$, the loss function $L(\boldsymbol{\theta}; \mathcal{S})$ in Equation 2.1 has the following property:

Lemma 2.10 (Fourier Transformation of the loss function: Generalized version)**.** *Let $\hat{L} : \mathbb{R}^p \to \mathbb{C}$ be the Fourier Transformation of $L(\boldsymbol{\theta}; \mathcal{S})$, namely for any $\mathbf{k} = (k_1, k_2, \cdots, k_p)^T \in \mathbb{Z}^p$, define:*

$$\hat{L}(\mathbf{k}) := \frac{1}{T_1 T_2 \cdots T_p} \int_{[0,T_1]\times\cdots\times[0,T_p]} L(\boldsymbol{\theta}; \mathcal{S}) \cdot \exp\Big(-i \sum_{l=1}^{p} \frac{k_l \theta_l}{T_l}\Big) \mathrm{d}\boldsymbol{\theta} \tag{2.145}$$

where T_l is the period of $L(\boldsymbol{\theta}; \mathcal{S})$ in θ_l. Let K be the support of $\hat{L}$ (i.e. $K := \{\mathbf{k} \in \mathbb{R}^p | \hat{L}(\mathbf{k}) \neq 0\}$). The Fourier degree of the loss function $\Delta_K := \max_{\mathbf{k}\in K} \sum_{l=1}^p |k_l|$ is bounded by $\sum_{l=1}^p \frac{T_l \cdot \Delta_l}{\pi}$.

Proof. For all $l \in [p]$, let $\{\mathbf{u}_i^{(l)}\}_{i=1}^d$ be the eigenvectors of $\mathbf{H}_l$ with corresponding eigenvalues $\{E_i^{(l)}\}_{i=1}^d$:

$$\mathbf{H}_l = \sum_{i=1}^{d} E_i^{(l)} \mathbf{u}_i^{(l)} \mathbf{u}_i^{(l)\dagger} \tag{2.146}$$

The unitary gate paramtrized by θ_l is therefore

$$\mathbf{V}_l(\theta_l) = \exp(-i\theta_l \mathbf{H}_l) = \sum_{i=1}^{d} e^{-i\theta_l E_i^{(l)}} \mathbf{u}_i^{(l)} \mathbf{u}_i^{(l)\dagger} \tag{2.147}$$

And the unitary gate $\mathbf{U}(\boldsymbol{\theta})$ can be written as:

$$\mathbf{U}(\boldsymbol{\theta}) = \mathbf{V}_p(\theta_p)\mathbf{V}_{p-1}(\theta_{p-1})\cdots\mathbf{V}_1(\theta_1) \tag{2.148}$$

$$= \Big(\sum_{i_p\in[d]} e^{-i\theta_p E_{i_p}^{(p)}}\mathbf{u}_{i_p}^{(p)}\mathbf{u}_{i_p}^{(p)\dagger}\Big) \tag{2.149}$$

$$\cdots\Big(\sum_{i_1\in[d]} e^{-i\theta_1 E_{i_1}^{(1)}}\mathbf{u}_{i_1}^{(1)}\mathbf{u}_{i_1}^{(1)\dagger}\Big) \tag{2.150}$$

$$= \sum_{\mathbf{i}\in[d]^p}\Big(\prod_{l=1}^{p-1}\mathbf{u}_{i_{l+1}}^{(l+1)\dagger}\mathbf{u}_{i_l}^{(l)}\Big)\Big(\prod_{l=1}^{p} e^{-i\sum_{l=1}^p\theta_l E_{i_l}^{(l)}}\Big) \tag{2.151}$$

$$\cdot\,(\mathbf{u}_{i_p}^{(p)}\mathbf{u}_{i_1}^{(1)\dagger}) \tag{2.152}$$

The output of the neural network given the density matrix $\boldsymbol{\rho}$ is therefore

$$f(\boldsymbol{\rho},\boldsymbol{\theta}) = \mathrm{tr}\Big(\mathbf{V}_p(\theta_p)\mathbf{V}_{p-1}(\theta_{p-1})\cdots\mathbf{V}_1(\theta_1)\boldsymbol{\rho}\mathbf{V}_1(\theta_1)^\dagger\mathbf{V}_2(\theta_2)^\dagger\cdots\mathbf{V}_p(\theta_p)^\dagger\mathbf{M}\Big) \tag{2.153}$$

$$= \sum_{\mathbf{i}\in[d]^p}\sum_{\mathbf{j}\in[d]^p}\hat{f}_{\mathbf{ij}}(\boldsymbol{\rho})\cdot e^{i\sum_{l=1}^p\theta_l(E_{j_l}^{(l)}-E_{i_l}^{(l)})} \tag{2.154}$$

where for any $\mathbf{i},\mathbf{j}\in[d]^p$

$$\hat{f}_{\mathbf{ij}}(\boldsymbol{\rho}) = \big(\mathbf{u}_{i_1}^{(1)\dagger}\boldsymbol{\rho}\mathbf{u}_{j_1}^{(1)}\big)\big(\mathbf{u}_{j_p}^{(p)\dagger}\mathbf{M}\mathbf{u}_{i_p}^{(p)}\big)\Big(\prod_{l=1}^{p-1}\mathbf{u}_{i_{l+1}}^{(l+1)\dagger}\mathbf{u}_{i_l}^{(l)}\mathbf{u}_{j_l}^{(l)\dagger}\mathbf{u}_{j_{l+1}}^{(l+1)}\Big) \tag{2.155}$$

This indicates the Fourier coefficients of $f(\boldsymbol{\rho};\boldsymbol{\theta})$ is supported on a subset of $\tilde{K} := \{(k_1,\cdots,k_p)|\forall l\in[p],\exists i,j\in[d]:k_l=\frac{(E_i^{(l)}-E_j^{(l)})T_l}{2\pi}\}$, and that the Fourier degree of $f(\boldsymbol{\rho},\boldsymbol{\theta})$ is bounded by $\sum_{l=1}^p\frac{T_l\cdot\Delta_l}{2\pi}$. Therefore for arbitray $\boldsymbol{\rho}$ and label y, the Fourier degree of the square loss $\big(f(\boldsymbol{\rho},\boldsymbol{\theta})-y\big)^2$, $\Delta_K\le\sum_{l=1}^p\frac{T_l\cdot\Delta_l}{\pi}$. Same holds for loss function $L(\boldsymbol{\theta};\mathcal{S})$ with arbitrary training set $\mathcal{S}$. □

For $\mathbf{H}_l$ with integral eigenvalues,

$$\exp(-i(\theta_l + 2\pi)\mathbf{H}_l) = \sum_{i=1}^{d} e^{-i(\theta_l+2\pi)E_i^{(l)}} \mathbf{u}_i^{(l)} \mathbf{u}_i^{(l)\dagger} = \exp(-i\theta_l \mathbf{H}_l). \tag{2.156}$$

Hence $T_l \leq 2\pi$ and $\Delta_K \leq 2\sum_{l=1}^{p} \Delta_l$. Let Δ be $\max_l \Delta_l$, the Fourier degree Δ_K of the loss function is bounded by $2\Delta p$.

For Hamiltonians with two-levels, we have the following corollary:

Corollary 2.11. *For quantum neural network instances composed of unitaries generated by two-level Hamiltonians, the Fourier degree of the loss function $L(\boldsymbol{\theta}; \mathcal{S})$ is bounded by $2p$ for arbitrary dataset $\mathcal{S}$.*

Proof. As shown earlier, for any Hermitian $\mathbf{M}$ and for $\mathbf{H}_l$ with the eigenvalues ± 1, the output $f(\boldsymbol{\rho}, \boldsymbol{\theta})$ is periodic in π for each coordinate. Also notice for all $l \in [p]$, $\Delta_l = 2$. Hence the Fourier degree Δ_K of $L(\boldsymbol{\theta}; \mathcal{S})$ is bounded by $2p$. □

2.10.2 Change of Variable and Root Counting

In this subsection, we elaborate on the change of variable and the upper bound on number of critical points by Bézout's Theorem. This would complete our proof for Theorem 2.5 and 2.9.

Let T_l be the period of $L(\boldsymbol{\theta}; \mathcal{S})$ in θ_l, and $\hat{L}(\mathbf{k})$ the Fourier coefficient for $\mathbf{k} = (k_1, \cdots, k_p)^T \in \mathbb{Z}^p$. We have

$$L(\boldsymbol{\theta}; \mathcal{S}) = \sum_{\mathbf{k} \in K} \hat{L}(\mathbf{k}) \Big(\cos \frac{k_1 \theta_1}{T_1} + i \sin \frac{k_1 \theta_1}{T_1} \Big) \cdot \cdots \cdot \Big(\cos \frac{k_p \theta_p}{T_p} + i \sin \frac{k_p \theta_p}{T_p} \Big) \tag{2.157}$$

Here $K \subseteq \mathbb{Z}^p$ is the support of the Fourier coefficients.

By definition, a local minimum must be a critical point, hence it suffices to bound the number of critical points for $L(\boldsymbol{\theta}; \mathcal{S})$. Define $G_l(\boldsymbol{\theta})$ as

$$G_l(\boldsymbol{\theta}) = \frac{\partial}{\partial \theta_l} L(\boldsymbol{\theta}; \mathcal{S}) \tag{2.158}$$

$$= \sum_{\mathbf{k} \in K} k_l \hat{L}(\mathbf{k}) \big(-\sin \frac{k_l \theta_1}{T_l} + i \cos \frac{k_l \theta_l}{T_l} \big) \tag{2.159}$$

$$\cdot \prod_{l' \neq l} \big(\cos \frac{k_{l'} \theta_{l'}}{T_{l'}} + i \sin \frac{k_{l'} \theta_{l'}}{T_{l'}} \big) \tag{2.160}$$

We can tell from above expression that the Fourier spectrum of G_l is supported on the same set K. A critical point of $L(\boldsymbol{\theta}; \mathcal{S})$ must satisfies that for all $l \in [p]$, $G_l(\boldsymbol{\theta}) = 0$.

By induction, $\cos k\theta$ can be expressed as a degree-k polynomial of $\cos \theta$ and $\sin k\theta$ as a degree-$(k-1)$ polynomial of $\cos \theta$ multiplied by $\sin \theta$. Consider the change of variable

$$c_l = \cos(\theta_l / T_l), \ s_l = \sin(\theta_l / T_l), \ \forall l \in [p]. \tag{2.161}$$

Let $g_l(c_1, s_1, \cdots, c_p, s_p)$ be the multivariate polynomial constraints corresponding to $G_l(\boldsymbol{\theta})$ after the change of variable:

$$g_l(c_1, s_1, \cdots, c_p, s_p) = \sum_{\mathbf{k} \in K} k_l \hat{L}(\mathbf{k}) \big(-s_l U_{k_l - 1}(c_l) + i T_{k_l}(c_l) \big) \prod_{l' \neq l} \big(T_{k_{l'}}(c_{l'}) + i s_{l'} U_{k_{l'} - 1}(c_{l'}) \big) \tag{2.162}$$

where $T_k(\cdot)$ and $U_k(\cdot)$ are Chebyshev polynomials of the first and second kind. For each g_l, the sum of degrees of $c_{l'}$ and $s_{l'}$ is bounded by $\max_{\mathbf{k} \in K} |k_{l'}|$, and the degree $\mathsf{deg}(g_l)$ of g_l is bounded by $\Delta_K = \max_{\mathbf{k} \in} \sum_{l=1}^{p} |k_l|$. The change of variable is one-to-one from $\theta_l \in [0, T_l)$ to a pair of

$(c_l, s_l) \in \mathbb{R}^2$ under the constraint $c_l^2 + s_l^2 = 1$. Therefore, it suffices to count the number of roots of the polynomial system with $2p$ parameters and $2p$ constraints:

$$\begin{cases} g_1(c_1, s_1, \cdots, c_p, s_p) = 0, \\ \vdots \\ g_p(c_1, s_1, \cdots, c_p, s_p) = 0, \\ h_1(c_1, s_1, \cdots, c_p, s_p) = c_1^2 + s_1^2 - 1 = 0, \\ \vdots \\ h_p(c_1, s_1, \cdots, c_p, s_p) = c_p^2 + s_p^2 - 1 = 0. \end{cases} \tag{2.163}$$

Notice that for general polynomial system, the number of critical points can be unbounded. For example, consider a system composed of constant polynomials, every point in the domain is a critical point. This corresponds to constant loss function, where the gradients vanishes everywhere with positive semidefinite Hessians. For this reason, we will focus on the non-degenerated case with finitely many critical points. Under the premise of non-degeneracy, Bézout's Theorem (e.g. Section 3.3 in [48]) states that the number of roots can bounded by the product of degree of polynomial constraints $2^p \mathsf{deg}(g_1)\mathsf{deg}(g_2)\cdots\mathsf{deg}(g_p). \leq (2\Delta_K)^p$.

We also prove a similar result for the more general case where the generators are Hamiltonians with integral spectrum: let Δ be the largest eigen-gap for each of the Hamiltonians, the number of local minima within each period is upper bounded by $O((\Delta p)^p)$. Combined with results in Section 2.10.1, the proof for Theorem 2.5 and 2.9 is complete.

2.11 More on Numerical Results

For all the experiments in this section, we study the p-parameter QNN as mentioned in Example 1 in Section 2.8, where :

$$\mathbf{M} := \otimes_{l=1}^{p}(\mathbf{Y} + \mathbf{I}) \tag{2.164}$$

$$\mathbf{H}_l := \left(\otimes_{r=1}^{l-1}\mathbf{I}\right) \otimes \mathbf{Z} \otimes \left(\otimes_{l+1}^{p}\mathbf{I}\right),\ \forall l \in [p] \tag{2.165}$$

In Section 2.11.1, we provide details and more numerical results for experiments described in Section 2.6. In Section 2.11.2, we visualize the 2-d loss landscape of Example 3.

2.11.1 Training with Gradient-based Methods

In this subsection, we use Example 2 to demonstrate that our construction can be hard to train with *gradient-based methods*. The loss function of the example can be expressed as

$$\frac{1}{2p}\sum_{l=1}^{p}\left((\sin 2\theta_l - \sin\frac{\pi}{50})^2 + \frac{1}{4}(\cos 2\theta_l - \cos\frac{\pi}{50})^2\right). \tag{2.166}$$

with global minima at $\boldsymbol{\theta}^\star = (\frac{\pi}{100}, \cdots, \frac{\pi}{100})^T$.

Hyperparameters. For all three optimizers, we choose the (initial) learning rate to be 0.01. For RMSProp, we choose the smoothing constant α for mean-square estimation to be 0.99. For Adam, we set the averaging coefficients $\beta_1 = 0.9$ for the gradients and $\beta_2 = 0.999$ for the its square. For L-BFGS we choose the history size to be 100. The numbers of iterations for training are set to 200-th for each pair of instances and optimizers; as can be seen in Figure 2.5, all pairs

have already converged at the 100-th iteration.

Training curves We plot the training curve for QNN instances with 2, 4, 8, 16 and 32 parameters with Adam, RMSProp and L-BFGS. For each pair of instance and optimizer, we repeat the experiments with uniform random initialization. As shown in Figure 2.5, for all the experiment setting considered here, while all initialization converge efficiently, there are initializations that does not converge to the global minima.

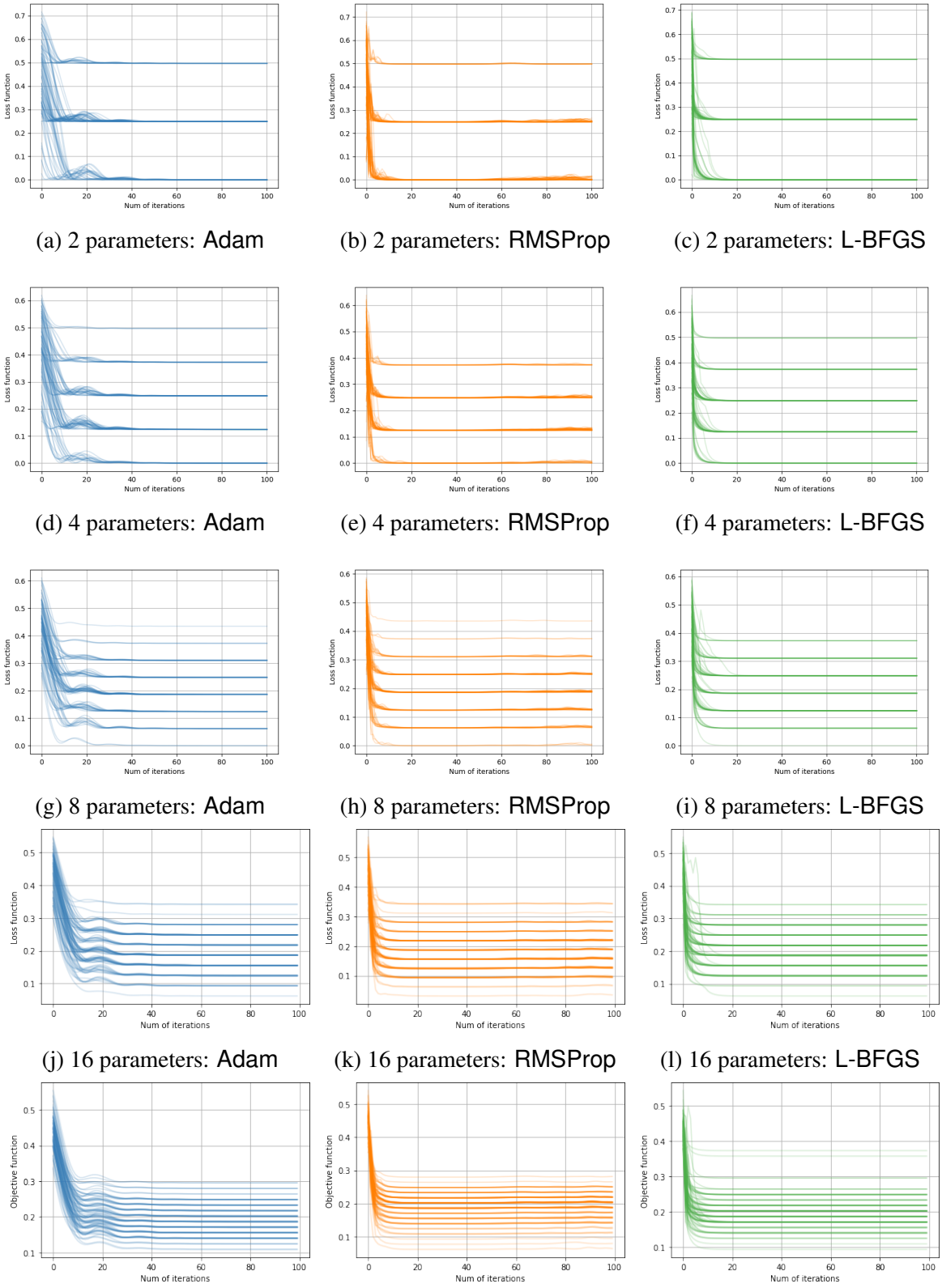
More on distributions of function values In Figure 2.7, we plot more results on the distribution of function values under RMSProp and have similar observation as mentioned in Section 2.6.

Calculating the empirical probabilities For all the instances of consideration, the function values of local minima can be calculated. For p-parameter instances, the function value of global minima is 0, and for the other local minima, the function values are at least $0.5/p$. For calculating the empirical probability that random initialization converges to the global minima, we count the number of trial that converge to values less than $0.25/p$.

2.11.2 Visualization: Non-decomposable Construction

For low-dimensional cases, it is possible to visualize the loss function of Example 3 of the construction by plotting the contour of the landscape. In Figure 2.8, we plot the contour of our construction for $p = 2$, with loss function proportional to

$$\begin{aligned}(\sin 2\theta_1 - \sin\frac{\pi}{50})^2 + (\sin 2\theta_2 - \sin\frac{\pi}{50})^2 \\ + \frac{1}{16}((\cos 2\theta_1 - \cos\frac{\pi}{50})^2 + \frac{3}{2}(\cos 2\theta_2 - \cos\frac{\pi}{50})^2) \\ + \frac{1}{8}(\cos 2\theta_1 \cos 2\theta_2 - \cos^2\frac{\pi}{50})^2 \quad (2.167)\end{aligned}$$



(a) 2 parameters: Adam (b) 2 parameters: RMSProp (c) 2 parameters: L-BFGS

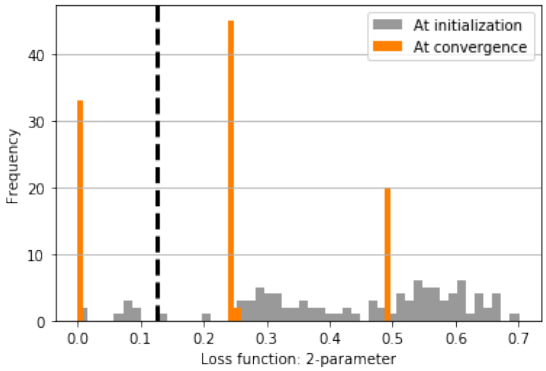
(d) 4 parameters: Adam (e) 4 parameters: RMSProp (f) 4 parameters: L-BFGS

(g) 8 parameters: Adam (h) 8 parameters: RMSProp (i) 8 parameters: L-BFGS

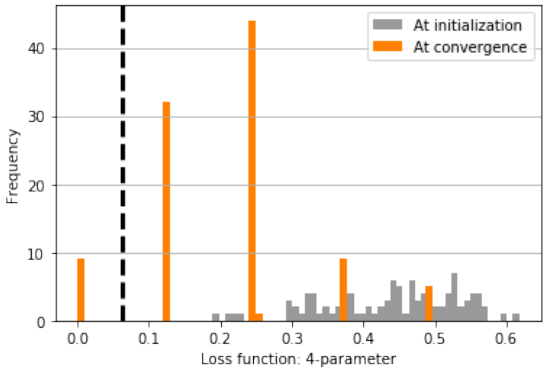
(j) 16 parameters: Adam (k) 16 parameters: RMSProp (l) 16 parameters: L-BFGS

(m) 32 parameters: Adam (n) 32 parameters: RMSProp (o) 32 parameters: L-BFGS

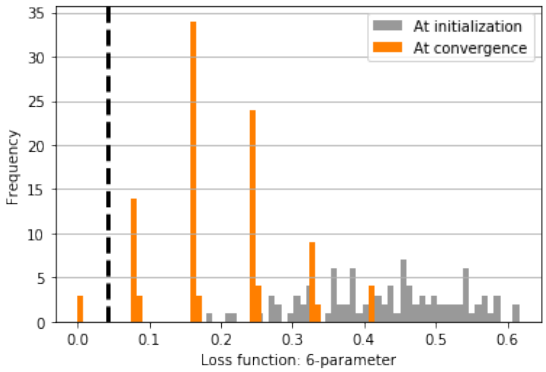
Figure 2.5: Empirical risk minimization of different QNN instance with Adam, RMSProp and L-BFGS. For each experiment setting we repeat 100 times.



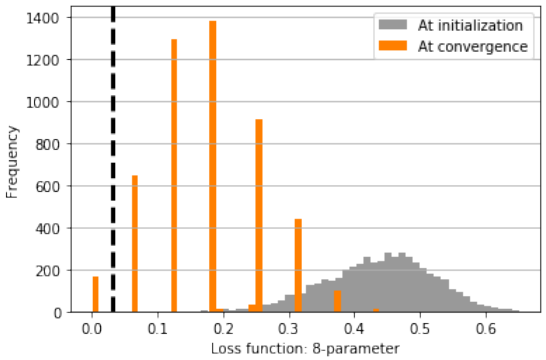
(a) 2 parameters: RMSProp



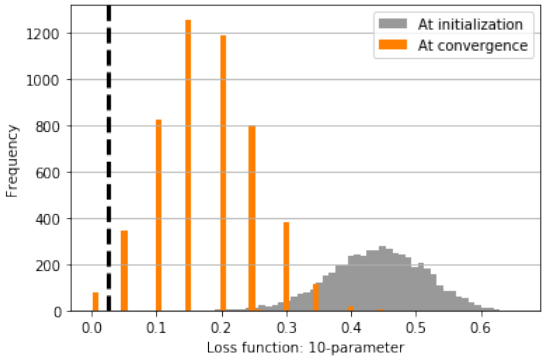
(b) 4 parameters: RMSProp



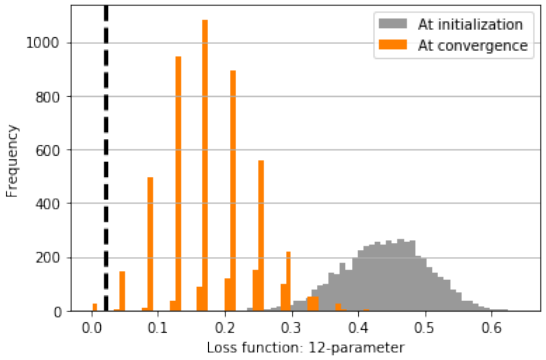
(c) 6 parameters: RMSProp



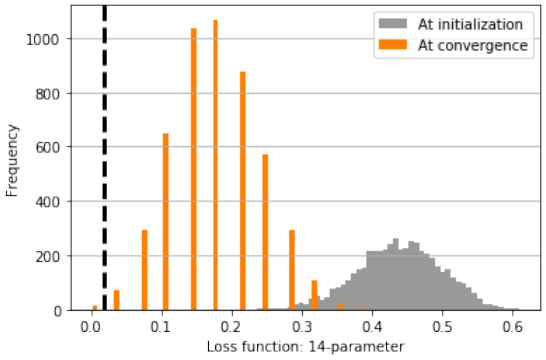
(d) 8 parameters: RMSProp



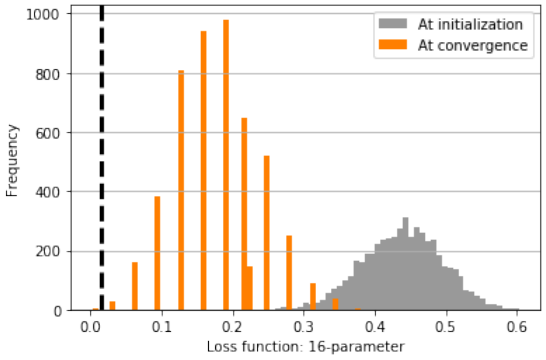
(e) 10 parameters: RMSProp



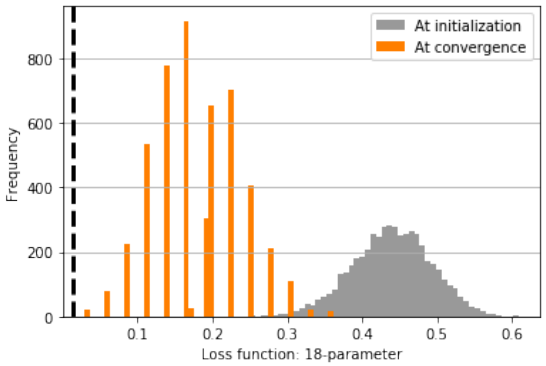
(f) 12 parameters: RMSProp



(g) 14 parameters: RMSProp



(h) 16 parameters: RMSProp



(i) 18 parameters: RMSProp

Figure 2.7: Distribution of function values of QNN instances with RMSProp. For instances with size $2, 4, 6$, the experiments are repeated 200 times; for the rest of the instances, the experiments are repeated 5000 times.

The global minima are $(k_1\pi + \frac{\pi}{100}, k_2\pi + \frac{\pi}{100})$ with $k_1, k_2 \in \mathbb{Z}$. Within each period, there are a total of 4 local minima where black box local search methods might stuck at. Among them, the global minima are marked in black. The gradient-based methods only converge to the global minimum when the initial value of the parameter lies in certain region.

2.11.3 Robustness of the Constructions

Our construction above demonstrates that in the worst-case, under-parameterized QNNs can have exponentially many local minima. It is natural to ask whether the local minima in our constructions are stable under perturbation. To this end, we repeated our experiments with Gaussian noises $\mathcal{N}(0, \sigma)$ added to the labels. The function values at local minima, as shown in Figure 2.9a (Cf. Figure 2.1,2.2 in the main text and Figure 2.7(h) in the supplementary material), have changed, as the noise breaks the symmetry of sub-optimal minima. But as shown in Figure 2.9b (Cf. Figure 2.3 in the main text), the exponential decay of success rate in finding the global minima remains for different σ (recall that the labels in our construction are in $[0, 1]$).

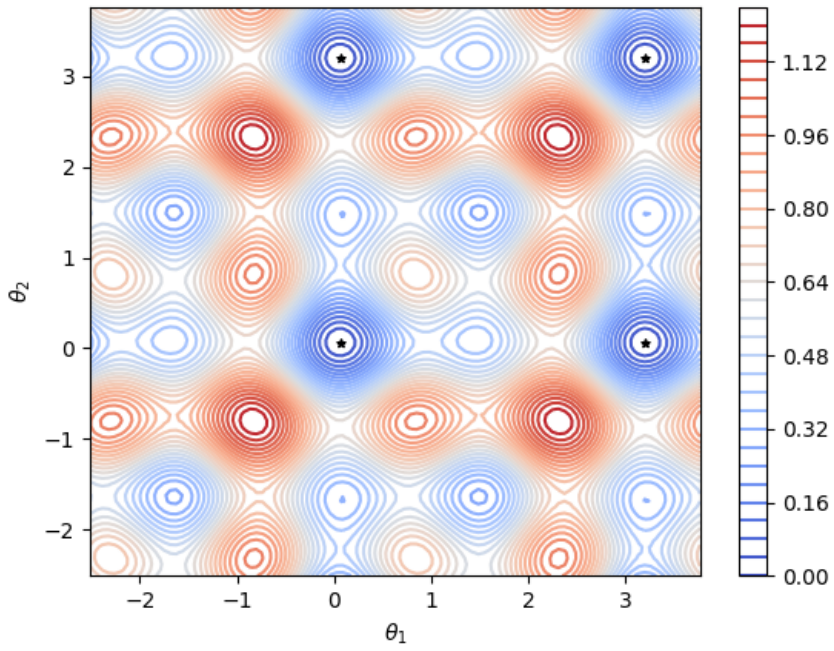
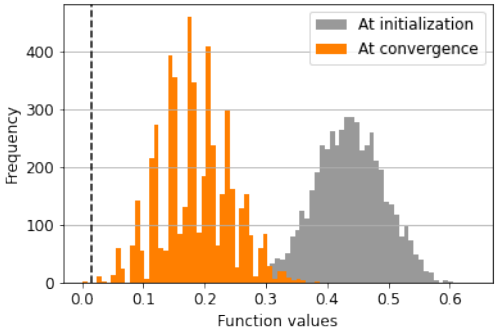


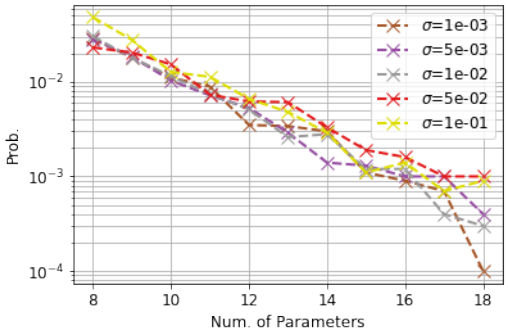
Figure 2.8: Landscape of the constructed QNN instance with 2 qubits and 2 parameters. The global optima are marked in black.

We used RMSProp optimizer, with other hyperparameters the same as the pervious experiments.

Moreover, by direct calculation of the suboptimality gaps and eigenvalues of Hessians at local minima, it can be proved that our examples are indeed robust against random label perturbations, quantum noise due to noisy gates, or due to the finite number of measurements, and even **adversarial** perturbations, as long as the resulting perturbation in the loss function is bounded in ℓ_∞-norm.



(a) Function values at initialization and at convergence for the 16-parameter instance with noisy labels, repeated for 5000 random initializations.



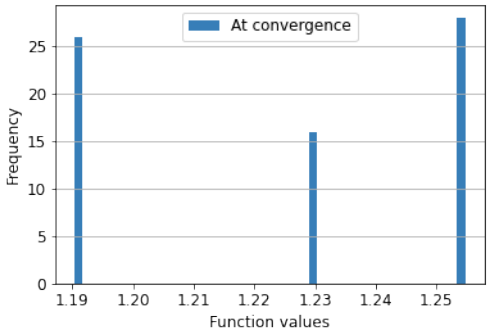
(b) The exponential decay of success rate for finding the global minimum under 10000 random initialization with label noise $\mathcal{N}(0, \sigma)$.

Figure 2.9: Empirical risk minimization with noisy labels. (a) the function values at convergence for a 16-parameter instance; the perturbation breaks the symmetry of the local minima, hence the more continuous spectrum of function values (Cf. Figure 2.2). (b) the decay of success rate for finding the global minima; the exponential tendency remains in the presence of Gaussian label noise up to $\sigma = 1e - 1$.

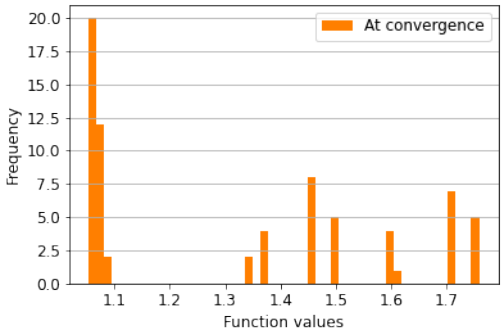
2.11.4 More Experimentrs on Datasets beyond Our Construction

So far we have observed exponentially many local minima in the datasets in our construction. Now we turn to more natural datasets that may appear in practice. Specifically, we consider the following family of datasets with a clear interpretation as an encoding of a classical, linearly separable concept: for the p-parameter instance, we first randomly choose $\mathbf{w} \in \mathbb{R}^{2p}$ as the

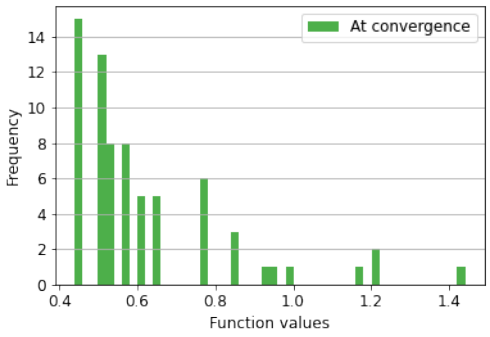
normal vector to the separating hyperplane. The classical dataset $\{(\mathbf{x}, y)|\mathbf{x} \in \mathbb{R}^{2p}, y \in \{0, 1\}\}$ is generated as follows: (1) uniformly draw the feature vector $\mathbf{x} = (x_1, \cdots, x_p, x_{p+1}, \cdots, x_{2p})^T$ from $[0, 2\pi]^{2p}$; (2) $y = 1$ if $\mathbf{w}^T\mathbf{x} > 0$ and $y = 0$ otherwise. The classical feature $\mathbf{x}$ is encoded into a quantum state $\boldsymbol{\rho}(\mathbf{x}) = |\Psi(\mathbf{x})\rangle\langle\Psi(\mathbf{x})|$ using the two-layer XY-encoder: $|\Psi(\mathbf{x})\rangle := \otimes_{l=1}^{p} \exp(-ix_{p+l}\mathbf{Y}_l) \otimes_{l=1}^{p} \exp(-ix_l\mathbf{X}_l)|0\rangle^{\otimes p}$. This process is repeated to construct a 100-sample dataset. For each QNN instance, we sampled 70 initial points and optimize the mean-square loss with RMSProp for 2000 iterations. The rest of the settings are the same as our original experiments. In Figure 2.10 (Cf. Figure 2.2 in the main text and Figure 2.7 (a)-(d) in the supplementary material), we trained instances with 2,4,6,8 qubits, each with 70 random initialization, and plotted the distribution of function values at convergence. There is a large number of local minima, and only a few random initialization ended up at the global minima. While we no longer have a clear exponential dependency, we did observe that as the number of parameters increases, the number of local minima increases significantly, and the success rate for finding global minima drops sharply. Such a phenomenon is also resilient to random choices of $\mathbf{w}$ and random sampling of feature vectors. This could be initial numerical evidence supporting the generality of our observed phenomena.



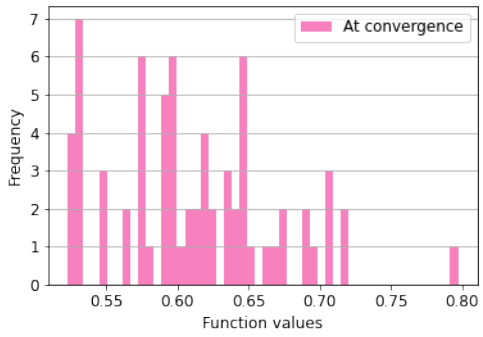
(a) 2 parameters: RMSProp



(b) 4 parameters: RMSProp



(c) 6 parameters: RMSProp



(d) 8 parameters: RMSProp

Figure 2.10: Empirical risk minimization for the common dataset using RMSProp. For each experiment setting, we repeat for 70 random initializations and run for 2000 iterations. The number of local minima increases significantly with the number of parameters.

Chapter 3: A Convergence Theory of Variational Quantum Eigensolvers

As we have seen in the last chapter, training VQAs is a notoriously challenging optimization problem due to its non-convex nature. In this chapter, we switch gear and consider another instantiation of variational algorithms, the variational quantum eigensolvers (VQEs). VQEs can be used for finding the ground states of physical Hamiltonians as well as for solving combinatorial optimization problems.

We will present a framework for analyzing the convergence of variational quantum systems when the number of parameters is sufficiently large. Applying it on VQEs, we provide the first rigorous result on the convergence of the gradient-based method in VQAs.

More concretely, we derive a threshold on the number of parameters to ensure the efficient convergence. The threshold is dependent on the properties of the ansatz through newly introduced quantities termed as the "effective dimension" and "effective spectral ratio". As an application, the threshold serves as a first-principled guidline for designing and comparing VQA procedures that are resource-efficient.

Our framework draws connections with the theory of over-parameterization in deep neural networks and illustrates the subtleties in adopting these approaches to variational quantum algorithms, and is applicable to the general VQA including QNNs (see Chapter 4).

3.1 Introduction

To execute a variational quantum algorithm, the quantum circuit must be repeatedly evaluated in order to find the optimal parameters. The cost on quantum resource for running a VQA is dominated by the number of executions of the quantum circuit and therefore primarily determined by how efficiently the optimizer finds optimal parameters. Empirically it has been observed that, the number of parameters controlling the variational circuits need to be sufficiently large in order to achieve efficient optimization with gradient descent (e.g. [44]). From a theoretical perspective, [57] shows that the optimization landscape transits from one that is swarmed by local minima, to one for which all local optima are almost global.

In this work, we take a further step to provide the first rigorous convergence theory for the variational algorithms. We focus on an instantiation of the VQA paradigm called the variational quantum eigensolvers (VQEs). The goal of a VQE is to approximate the ground state of a given Hamiltonian with the output of a variational ansatz. We show that, when the number of parameters in a variational ansatz exceeds certain threshold (referred to as the *trainability threshold*), the method of gradient flow can efficiently find a good approximation of the ground state. More concretey, with high probability over the random initialization of the ansatz parameters, the infidelity between the output state and the ground state decays to 0 as an exponential function of the training time. We show that the smallest number of parameters to ensure such efficient convergence (also known as the *trainability threshold*) depends polynomially on the dimension d of the physical system, and a quantity κ referred to as the spectral ratio.

Our theory provides a practical way to compare and predict the performance of different ansatz designs: to evaluate the performance of ansatz, it typically requires repeated training

sweeping across different random seeds and number of parameters. Highlighted by our theoretical result, it suffices to evaluate the dimension d and the spectral ratio κ (defined later in Section 3.3), which can be estimated by classical simulation by sampling and is empirically more efficient than benchmarking with repeated trainings.

Our theory also leads to a principled way for designing VQEs: in practice, by designing different variational quantum circuits (also referred to as the ansatze), users can easily trade off between the quality of solutions and the budget on quantum resources. Such examples include designing the number of layers of an ansatz and the Hamiltonians that generate the parameterized quantum gates. Up to now, ansatz designs are mainly based on heuristics: there are inspirations drawn from the literature of machine learning and optimization, as well as ideas based on the quantum adiabatic theorem (e.g. [62]) or implementation considerations (e.g. [63]). To showcase our theory as a guideline for ansatz designs, we consider two practices called the VQE compression and the VQE preconditioning.

The VQE compression is a two-stage procedure for finding a shallow quantum circuits for ground state preparation. It can significantly save the run-time quantum resources for instances with the spectral ratio $\kappa \gg 1$. The VQE preconditioning allows the trainability of a shallow variational circuit at the price of more quantum readout measurements. It can potentially extend the capability of the near-term quantum computers as the circuit depth is typically the bottleneck on NISQ machines.

We summarize our practical contributions as follows:

1. We pinpoint two "effective" quantities that are dependent on the ansatz design, allowing the performance comparison of different ansatz without training over different random

initializations and sweeping over different number of parameters.

2. As a corollary of our main theorem, we recover the smallest number of parameters such that the output of a variational ansatz can approximate any pure state (also referred to as the *expressivity threshold*). The gap between the *trainability* and *expressivity threshold* inspires a two-stage procedure for compressing the run-time variational circuit, which can drastically save the quantum resources for repeated preparing the ground state of a given Hamiltonian;

3. By showing the dependency of the *trainability threshold* on a effective quantity dependent on the eigenvalues of the Hamiltonian to be solved, we highlight the role of preconditioning in VQEs: by implementing a preconditioning procedure, a variational circuit with fewer number of parameters can be made trainable at the price of more measurements.

In addition, our analysis can be adapted to guide the design of general variational quantum algorithms.

3.1.1 Related Works

Landscape of VQA Training. The variational quantum algorithm faces several practical issues with their deployment. In particular, the optimization problems associated with VQA training are highly *non-convex* and not efficiently solvable, as the gradient based methods are prone to converge to local minima and saddle points. As proven in [57, 64], when a variational circuit is under-parameters (e.g. with number of parameters being a poly-log function of the system dimension), the landscape of the underlying optimization problem can be swarmed with sub-optimal minima. The VQA landscape of VQEs also suffer from the 'vanishing gradients' or

barren plateaus [40]: at random initialization, the magnitude of the gradient would decrease polynomially with the system size, making the optimization procedure intolerant of noise. Instead of looking into the optimization landscape, we prove the convergence by directly studying the optimization dynamics under gradient flow. Our work extends the landscape study by providing the convergence rate (note that a landscape with convexity or without spurious minima does not imply a fast rage of convergence) and highlight the role of the spectral ratios.

3.2 Preliminaries

In this section, we layout the definitions and notions for VQEs, and review some classical results on training over-parameterized models.

3.2.1 Variational Quantum Eigensolvers

An instance of variational quantum eigensolvers is specified by the problem Hamiltonian, input state and the ansatz. It is defined as below:

Definition 3.1 (Variational quantum eigensolvers)**.** A d-dimensional variational quantum eigensolver instance is specified by a triplet $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ with $d \times d$ problem Hamiltonian $\mathbf{M}$, an input state $|\Phi\rangle \in \mathbb{C}^d$ and an ansatz $\mathbf{U}\colon \mathbb{R}^p \to \mathbb{C}^{d\times d}$. Let $\lambda_1 < \lambda_2 \leq \cdots \leq \lambda_d$ be the eigenvalues of $\mathbf{M}$ in an ascending order. The goal is to approximate the ground state of $\mathbf{M}$ (i.e. the eigenvector associated with λ_1) with $\mathbf{U}(\boldsymbol{\theta})|\Phi\rangle$ by solving the optimization problem:

$$\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) := \langle\Phi|\mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}\mathbf{U}(\boldsymbol{\theta})|\Phi\rangle \tag{3.1}$$

The search for the optimal parameters $\boldsymbol{\theta}^\star$ are commonly performed by gradient descent: at each time step t, the parameters are updated as:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}). \tag{3.2}$$

For sufficiently small learning rate η, the dynamics of gradient descent reduces to that of gradient flow

$$d\boldsymbol{\theta}/dt = -\eta \nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}). \tag{3.3}$$

In this chapter as well as the next chapter, we will focus on the gradient flow following [32].

Fully- and Partially-Trainable Ansatz. The parameterization of $\mathbf{U}$ is referred to as the ansatz design in the quantum computing literature. A popular choice of ansatz design is the *hardware-efficient ansatz* (HEA, e.g. [63]). HEA makes use of native gates of a quantum computer and is typically composed of interleaving single-/two-qubit Pauli rotations and entanglement unitaries implemented with CZ / CNOT gates. The main motivation behind the design is to facilitate the implementation on real quantum machines. Another popular choice of ansatz design is the *Hamiltonian variational ansatz* (HVA). It is partially inspired by the adiabatic theorem and utilizes the structure of the problem Hamiltonians (e.g. [1, 62]). For HVA $\mathbf{U}$ composed of parameterized rotations generated by a set of Hermitians that sums to the problem Hamiltonian.

In this work, we consider a general family of ansatze taking the HEA and HVA as special cases:

Definition 3.2 (Fully-trainable ansatz)**.** A fully-trainable L-layer ansatz with a set of Hermitians

$\mathcal{A} = \{\mathbf{H}^{(1)}, \mathbf{H}^{(2)}, \cdots, \mathbf{H}^{(K)}\}$ has $K \cdot L$ trainable parameters and is defined as

$$\mathbf{U}^{(L)}(\boldsymbol{\theta}) = \prod_{l=1}^{L} \prod_{k=1}^{K} \exp(-i\theta_{l,k}\mathbf{H}^{(k)}). \tag{3.4}$$

The superscript L will be omitted when there is no ambiguity.

To see that the ansatz defined in Definition 3.2 is a superset of HVA and HEA, notice that the fully-trainable ansatz is an HVA if the problem Hamiltonian $\mathbf{M}$ can be represented as a linear combination of $\{\mathbf{H}^{(k)}\}_{k=1}^{K}$. As for the HEA, it can in general can be expressed as

$$\mathbf{U}^{(L)}(\boldsymbol{\theta}) = \prod_{l=1}^{L} \Big(\prod_{k=1}^{K'} \exp(-i\theta_{l,k'}\mathbf{H}^{(k')})\mathbf{U}_{\mathsf{ent}} \Big) \tag{3.5}$$

where $\mathbf{H}^{(k')}$ are single-/two-qubit Pauli rotations and $\mathbf{U}_{\mathsf{ent}}$ corresponds to an entanglement layer composed of CZ and CNOT gates. If the smallest integer C such that $\mathbf{U}_{\mathsf{ent}}^{C} = \mathbf{I}$ exists, the HEA can be expressed as a fully-trainable ansatz with $K = C \cdot K'$, with each generating Hermitian represented as $\mathbf{U}_{\mathsf{ent}}^{c}\mathbf{H}^{(k)}\big(\mathbf{U}_{\mathsf{ent}}^{c}\big)^{\dagger}$ for $c \in [C]$ and $k \in [K]$.

Under Definition 3.2, the parameterization is determined by a fixed set of Hermitians $\mathcal{A} = \{\mathbf{H}^{(1)}, \cdots, \mathbf{H}^{(K)}\}$ and a domain of parameters in each layer $\Theta \subseteq \mathbb{R}^{K}$ up to the choice of number of layers L.

Given an ansatz design $(\mathcal{A}, \Theta)$, the set of all achievable unitary matrices forms a subgroup of $SU(d)$:

$$G_{\mathcal{A},\Theta} = \cup_{L=0}^{\infty}\{\mathbf{U}^{(L)}(\boldsymbol{\theta}) : \boldsymbol{\theta} \in \Theta^{L} \subseteq \mathbb{R}^{K \cdot L}\}. \tag{3.6}$$

For many choices of ansatz with a limited set of $\mathcal{A}$, $G_{\mathcal{A},\Theta}$ is a proper subgroup of $SU(d)$. We omit the subscript Θ and denote the subgroup as $G_{\mathcal{A}}$ with the domain of the parameters is clear from the context.

Define a *partially-trainable ansatz* associated with $\mathcal{A}$ as:

Definition 3.3 (Partially-trainable anastz for $\mathcal{A}$)**.** Let the subgroup $G_{\mathcal{A}}$ be a subgroup of $SU(d)$ associated with fully-trainable ansatz with a set of Hermitians $\mathcal{A} = \{\mathbf{H}^{(1)}, \mathbf{H}^{(2)}, \cdots \mathbf{H}^{(K)}\}$. The corresponding p-parameter partially-trainable ansatz is defined as:

$$\mathbf{U}(\boldsymbol{\theta}) = \mathbf{U}_p \exp(-i\theta_p \mathbf{H}) \cdots \mathbf{U}_l \exp(-i\theta_l \mathbf{H}) \cdots \mathbf{U}_1 \exp(-i\theta_1 \mathbf{H}) \mathbf{U}_0. \tag{3.7}$$

Here $\mathbf{H}$ is an arbitrary Hermitian in $\mathcal{A}$ and $\mathbf{U}_l$ are $i.i.d.$ sampled from the Haar measure over $G_{\mathcal{A}}$.

We highlight that the partially-trainable ansatz can be viewed as a fully-trainable ansatz trained on a subset of the parameters, hence the name "partially trainable": without loss of generality, assume we choose $\mathbf{H}^{(1)}$ as the generating Hermitian $\mathbf{H}$ in Definition 3.3. Performing gradient descent on the parameters corresponding to $\mathbf{H}^{(1)}$ in every L'-layers (i.e. gradient descent on $\theta_{1,1}, \theta_{L'+1,1}, \theta_{2L'+1,1}, \cdots$) of a randomly-initialized fully-trainable ansatz is then equivalent to optimizing the partially-trainable ansatz with $\mathbf{H} = \mathbf{H}^{(1)}$ with $\mathbf{U}_l$ being a L'-step random walk with step sample from $S := \{\prod_{k=1}^{K} \exp(-i\theta_k \mathbf{H}^{(k)}) : \boldsymbol{\theta} \in \Theta \subseteq \mathbb{R}^K\}$. Under mild regularity conditions, the random walk converges to the Haar measure over $G_{\mathcal{A},\Theta}$ (See [65, Section 3]).

3.2.2 Convergence in over-parameterized classical systems

The over-parameterization has been proposed as an explanation to the convergence of deep learning (e.g. [31, 32, 66]). The convergence of over-parameterized model arises from two main

phenomenon:

1. **Convergence of expected dynamics:** When the parameters are randomly initialized, the expected dynamics of the training are shown to exhibit convergence to a global minima. The expected dynamics is therefore a smoothed version of the actual dynamics that removes some of the irregularities that can lead to a failure in convergence.

2. **Convergence under perturbation:** Despite the convergence of the training dynamics in expectation, the actual training corresponds to a particular setting of initial parameters. This leads to the actual training being a perturbed version of the expected dynamics, it is thus necessary to show that the convergence of this dynamics is robust to small perturbations.

3. **Concentration at initialization:** Due to the law of large numbers, with high probability, deviations from the expected dynamics decrease as the number of random parameters increases. Over-parameterization thus plays the crucial role of leading to the *concentration* of the dynamics around the expected value, allowing the magnitude of random perturbations to be bounded with high probability.

4. **Lazy training:** It must be shown that the actual training concentrates throughout the training given the convergence at initialization. This phenomenon has been characterized as *lazy training* [66], where the dynamics of a system at initialization remain a good approximation throughout its training. Once again, over-parameterization plays an important role in ensuring this phenomenon; as the number of parameters increases the changes in each parameter become smaller with high probability over the course of training.

This method can be illustrated by the example of the *Neural Tangent Kernel* [31], which has been

used to show convergence while training several over-parameterized classical neural networks including wide feedforward networks [32].

Consider a classical classification problem where the input data is drawn from a distribution p_{in} over $\mathcal{R}^{n_0}$ and an output in $\mathcal{R}^{n_L}$, the space of valid functions is given by $\mathcal{F} = \{\mathbf{f} \colon \mathcal{R}^{n_0} \to \mathcal{R}^{n_L}\}$. The model is specified as a *realization function* mapping p parameters to candidate functions $\mathbf{F}^{(L)} \colon \mathcal{R}^p \to \mathcal{F}$. Denoting the parameters at time t by $\theta(t) = (\theta_1(t), \ldots, \theta_p(t))$, the function at time t is given by $\mathbf{F}^{(L)}(\theta(t))$. The data distribtution induces an inner product over $\mathcal{F}$ given by $\langle \mathbf{f}, \mathbf{g} \rangle_{p_{in}} = \mathbb{E}_{x \sim p_{in}}[\mathbf{f}(x)^T \mathbf{g}(x)]$. Given a cost function C, the gradient flow dynamics of the system correspond to *kernel training* with respect to the *neural tangent kernel* (NTK) given by $\tilde{\mathbf{K}} = \sum_{l=1}^{p} \frac{\partial}{\partial \theta_l} \mathbf{F}^{(l)}(\theta) \otimes \frac{\partial}{\partial \theta_l} \mathbf{F}^{(l)}(\theta)$.

Let $\mathbf{y} \in \mathcal{F}$ be the true function mapping inputs to ouputs resulting in the residual function $\nabla(\theta(t)) = \mathbf{y} - \mathbf{F}^{(L)}(\theta(t))$. If C is the squared loss function, the dynamics of the system is simply given by $\dot{\mathbf{r}} = -\eta \tilde{\mathbf{K}} \mathbf{r}$ where η is the chosen step size. It is known that if $\tilde{\mathbf{K}}$ is a constant positive definite matrix, the system exhibits linear convergence. Following the above recipe, this leads to a framework for showing the convergence of classical neural networks, it is shown that $\mathbf{K} = \mathbb{E}(\tilde{\mathbf{K}}(\theta(0)))$ is a positive definite constant matrix. It is also shown that the dynamics $\dot{\mathbf{r}} = -\eta \tilde{\mathbf{K}} \mathbf{r}$ converges whenever $\|\tilde{\mathbf{K}} - \mathbf{K}\| \leq \epsilon_0$. Further define an over-parameterization threshold $P^{(L)}(n_0, n_L)$Convergence can then be established via the following propositions:

1. **Concentration at initialization**: If $p > P^L$, $\|\tilde{\mathbf{K}}(\theta(0)) - \mathbf{K}\| \leq \epsilon_0$ with probability at least $9/10$.

2. **Small perturbations imply convergence**: $\|\tilde{\mathbf{K}}(\theta(t)) - \mathbf{K}\| \leq \epsilon_0$ for all $t < t'$, we have $\|\mathbf{r}(t) - \tilde{\mathbf{r}}(\mathbf{t})\| \leq \epsilon_1$ for all $t \leq t_1$, where $\tilde{\mathbf{r}}$ denotes the residuals when the kernel is frozen

at initialization (in which case the system is known to converge).

3. **Convergence implies small perturbations**: If $p > P^L$, and $\|\mathbf{r}(t) - \tilde{\mathbf{r}}(\mathbf{t})\| \leq \epsilon_1$ for all $t < t'$, we have $\|\tilde{\mathbf{K}}(\theta(t)) - \mathbf{K}\| \leq \epsilon_0$ for all $t \leq t'$ with probability at least $9/10$

These propositions are sufficient to inductively prove the convergence of the training dynamics to a global minimum. Consider the earliest time t_0 where the perturbation in the kernel is too large; by the final proposition this can only occur if the convergence of the system is violated at some time $t_0' < t_o$. However, by the second proposition, this would imply that for an earlier time t_0'' the kernel perturbation must have been too large, contradicting our initial assumption that t_0 was the earliest such time. This shows that both the small perturbation condition as well as the convergence of the system are maintained throughout the training.

3.3 Main Result

Our main result states that, for both the fully-trainable and partially-trainable ansatz, with high probability over random initializations, the VQE objective function converges exponentially in terms of time t under gradient flow, when the number of parameters exceeds a threshold of over-parameterization depending polynomially with the system dimension and a characterizing quantity called spectral ratio $\kappa := \frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}$.

Convergence in VQE with General Ansatze. We start by characterizing the dynamics of VQE training for general variational ansatze as defined similarly in Chapter 2:

$$\mathbf{U}(\boldsymbol{\theta}) := \exp(-i\theta_p \mathbf{H}_p) \cdots \exp(-i\theta_2 \mathbf{H}_2) \exp(-i\theta_1 \mathbf{H}_1), \tag{3.8}$$

where $\{\mathbf{H}_1, \mathbf{H}_2, \cdots, \mathbf{H}_p\}$ are non-zero, traceless $d \times d$ Hermitian matrices. The traceless condition is without loss of generality modulo a phase factor.

Lemma 3.1 (VQE dynamics under gradient flow)**.** *Consider a d-dimensional VQE instance $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$, with arbitrary input state $|\Phi\rangle$ and initial parameters $\boldsymbol{\theta}(0)$. Let $\mathbf{U}$ be parameterized by $\{\mathbf{H}_1, \cdots, \mathbf{H}_p\}$ as defined in Equation 3.8. Under gradient flow with learning rate $\eta = \frac{d^2-1}{\sum_{l=1}^{p} \operatorname{tr}(\mathbf{H}_l^2)}$, the output state $|\Psi(t)\rangle = \mathbf{U}(\boldsymbol{\theta}(t))|\Phi\rangle$ follow the dynamics*

$$\frac{d}{dt}|\Psi(t)\rangle = -\Xi_t\big([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\big)|\Psi(t)\rangle. \tag{3.9}$$

Here $\Xi_t(\cdot)$ is an endomorphism on skew Hermitians defined as

$$\Xi_t(\mathbf{A}) = \frac{d^2-1}{\sum_{l=1}^{p} \operatorname{tr}(\mathbf{H}_l^2)} \sum_{l=1}^{p} \operatorname{tr}(\mathbf{H}_{l:p}(\boldsymbol{\theta}(t))\mathbf{A})\mathbf{H}_{l:p}(\boldsymbol{\theta}(t)) \tag{3.10}$$

with $\mathbf{H}_l(\boldsymbol{\theta}) := \mathbf{U}_{l:p}(\boldsymbol{\theta})\mathbf{H}_l\mathbf{U}_{l:p}(\boldsymbol{\theta})$ defined as $\mathbf{H}_l$ conjugated by

$$\mathbf{U}_{l:p}(\boldsymbol{\theta}) := \exp(-i\theta_p\mathbf{H}_p)\cdots\exp(-i\theta_l\mathbf{H}_l). \tag{3.11}$$

When $\Xi_t(\cdot)$ is exactly the identity map $\mathsf{id}(\cdot)$ on skew Hermitians, the dynamics in Equation 3.36 coincides with that of *Riemannian gradient flow* (RGF) on the unit sphere: The gradient of the objective function in Equation 3.1 with respect to the output state $|\Psi(t)\rangle$ is $\mathbf{M}|\Psi(t)\rangle$. The projector onto the tangent space of the unit sphere at $|\Psi(t)\rangle$ is $\mathbf{I} - |\Psi(t)\rangle\langle\Psi(t)|$ with $\mathbf{I}$ being the

identity matrix. The Riemannian gradient is therefore

$$(\mathbf{I} - |\Psi(t)\rangle\langle\Psi(t)|)\mathbf{M}|\Psi(t)\rangle = [\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]|\Psi(t)\rangle. \tag{3.12}$$

Compared with regular gradient flow, RGF on the unit sphere preserves the length of the state vector. The main result by Xu et al in [67] implies that, under mild initial condition on $|\Psi(0)\rangle$, RGF finds an ϵ-approximation of the ground state in $O(\log(1/\epsilon))$ time.

We show that the same claim is valid for $\Xi_t(\cdot)$ sufficiently close to $\mathsf{id}(\cdot)$, measured in terms of the induced norm $\|\cdot\|_{\infty,1}$ defined as $\|\Phi\|_{\infty,1} := \max\{\|\Phi(\mathbf{A})\|_{\mathsf{op}} : \text{skew Hermitian } \mathbf{A} \in \mathbb{C}^{d\times d}, \|\mathbf{A}\|_{\mathrm{tr}} = 1\}$. Here $\|\cdot\|_{\mathsf{op}}$ and $\|\cdot\|_{\mathrm{tr}}$ are the matrix operator- and trace-norm.

Lemma 3.2 (Robustness of RGF convergence for VQE)**.** *Consider a d-dimensional VQE instance $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ with $\mathbf{U}$ parameterized by $\{\mathbf{H}_1, \cdots, \mathbf{H}_p\}$ as defined in Equation 3.8. Let $\lambda_1 < \lambda_2 \leq \cdots \leq \lambda_d$ be the eigenvalues of $\mathbf{M}$.*

If (1) the output state at initialization $|\Psi(0)\rangle$ has non-negligible overlap with the target ground state $|\Psi^\star\rangle$, such that $|\langle\Psi(0)|\Psi^\star\rangle|^2 \geq \Omega(\frac{1}{d})$, and (2) for all $t \in [0, T]$, $\|\Xi_t - \mathsf{id}\|_{\infty,1} \leq O(\frac{\lambda_2-\lambda_1}{\lambda_d-\lambda_1} \cdot \frac{1}{d})$, then under the dynamics $\frac{d}{dt}|\Psi(t)\rangle = -\Xi_t([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|])|\Psi(t)\rangle$, the output state converges to the ground state efficiently as $1 - |\langle\Psi(t)|\Psi^\star\rangle|^2 \leq \exp(-c\frac{\lambda_2-\lambda_1}{\log d}t)$ for some constant c and $t \in [0, T]$.

The condition of non-negligible overlap is satisfied with constant probability when $|\Psi(0)\rangle$ is chosen uniformly random from all pure states; to establish the convergence result, it remains to show that a given over-parameterized VQE instance leads to small $\|\Xi_t - \mathsf{id}\|_{\infty,1}$ throughout the optimization.

Convergence for Partially-Trainable Ansatze. For ansatze in Definition 3.3, we rigorously

show that the premises of Lemma 3.12 holds with high probability and therefore the following theorem holds:

Theorem 3.3 (Exponential convergence of VQE)**.** *Consider a d-dimensional VQE instance $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ with $\mathbf{U}$ generated by $\mathbf{H}$ as in Definition 3.3. Let $\lambda_1 < \lambda_2 \leq \cdots \leq \lambda_d$ be the eigenvalues of $\mathbf{M}$ and $|\Phi^\star\rangle$ be the ground state. If the number of parameters p of order $\mathsf{poly}(d, \kappa)$ with $\kappa := \frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}$, then under gradient flow on $\boldsymbol{\theta}$ with learning rate $\eta = \frac{d^2-1}{p\,\mathrm{tr}(\mathbf{H^2})}$, the output state $|\Psi(t)\rangle$ converges to an ϵ-approximation of the ground state $|\Psi^*\rangle$ such that $\epsilon = 1 - |\langle\Psi(T_\epsilon)|\Psi^\star\rangle|^2$ in time $T_\epsilon = O\big(\frac{\log d}{\lambda_2 - \lambda_1} \log \frac{1}{\epsilon}\big)$, with success probability $2/3$.*

The success probability in Theorem 3.3 can be boosted to $1 - \delta$ for any $0 \leq \delta \leq 1$ using $O\left(\log\left(\frac{1}{\delta}\right)\right)$ repetitions, with the parameters randomly reinitialized each time. The proof of Theorem 3.3 follows from the fact that the largest deviation from RGF scales with the number of parameters p as $O(1/\sqrt{p})$: later in Lemma 3.14 and 3.15 (Section 3.11), we show that $\sup_{t\in[0,T_\epsilon]} \|\Xi_t - \mathsf{id}\|_{\infty,1}$ is $O(\mathsf{poly}(d,\kappa)/\sqrt{p})$ both at random initialization and during training. Furthermore, we empirically observe that the same scaling holds for fully-trainable ansatze (see Section 3.11.4 for more details).

3.4 Experiments: Trainability and Expressive Threshold

For a quantitative demonstration of Theorem 3.3, we examine how the convergence depends on the number of parameters p for varying (d, κ) in Figure 3.1 for synthetic instances.

More concretely, for a VQE instance, define the success rate as the probability for finding an approximation of the ground state with fidelity ≥ 0.99 under random initialization. For each set of (d, κ), the success rate starts as 0, increases as the number of parameters increases, and

saturates at 1 for large number of parameters.

Define the over-parameterization threshold $p^\star$ as the smallest number of parameters p to achieve a success rate of at least 98%. In the insets of Figure 3.1, we observe $p^\star$ as polynomial functions of d and κ.

For each synthetic instance (d, κ), we choose $d \times d$ problem Hamiltonian

$$\mathbf{M} = \mathbf{U}_{\mathsf{rot}}^\dagger \mathsf{diag}(0, 1/\kappa, 1, 1, \cdots, 1)\mathbf{U}_{\mathsf{rot}} \tag{3.13}$$

with random unitary $\mathbf{U}_{\mathsf{rot}}$. The ansatz $\mathbf{U}(\cdot)$ is as defined in Definition 3.3 generated by Pauli-like $\mathbf{H}$ with eigenvalues ± 1, normalized such that $\frac{\mathrm{tr}(\mathbf{H}^2)}{d^2-1} = 1$. More concretely,

$$\mathbf{H} = \sqrt{\frac{d^2-1}{d^2}}\mathsf{diag}(1, \cdots, 1, -1, \cdots, -1). \tag{3.14}$$

For instances with p parameters, we choose the learning rate $\eta = 1 \times 10^{-2}/p$ and optimize with 10000 iterations.

Expressivity Threshold $p_\star$. In addition, Figure 3.1 reveals information on $p_\star$, the smallest p such that the success rate exceeds 0. $p_\star$ increases with d and remains approximately the same as κ increases. Assuming the number of random initializations is sufficiently large, the gradient-based methods exhaustively search the parameter space, and the success rate exceeding 0 is equivalent to the existence of a set of parameters that realize the target state. This suggests that $p_\star$ is the *expressivity* threshold of a variational ansatz.

A concept closely related to expressivity in the literature of quantum control is the *controllability*: a parameterized quantum system is said to have (complete) controllability if for any unitary, there

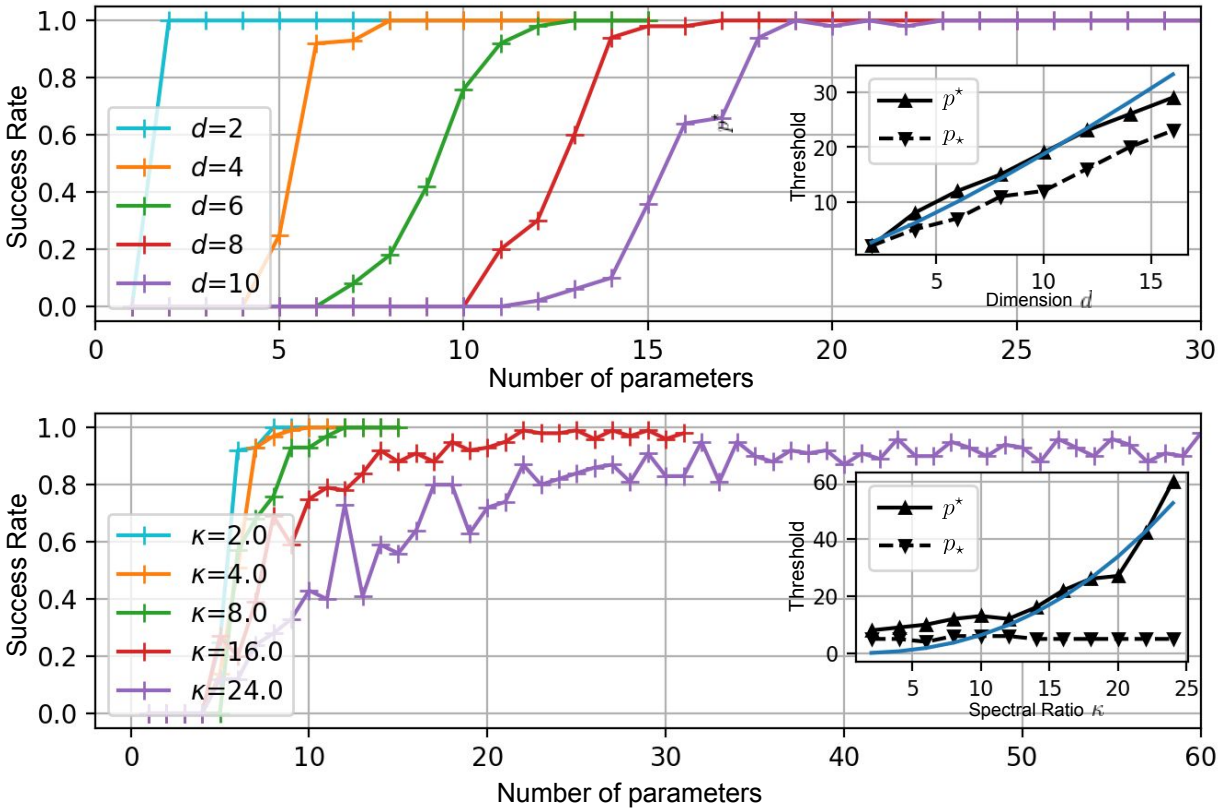


Figure 3.1: Success rates for finding good approximations to the ground states increase with the number of parameters p for synthetic instances. The solution is considered a good approximation if the infidelity with the target state is ≤ 0.01. Each data point is evaluated over 50 random initializations. Top: fix $\kappa = 2.0$, vary the dimension d from 2 to 10; Bottom: fix $d = 4$, vary the spectral ratio κ from 2.0 to 24.0. The insets plot the over-parameterization thresholds $p^\star$ and $p_\star$ against d and κ. Thresholds $p^\star$ (resp. $p_\star$) are defined as the smallest p such that the success rates exceed 98% (resp. 0%). The reference lines in blue are monomials with degree 1.22 and 2.41 respectively.

exists a set of parameters such that the unitary can be realized (see e.g. [68, 69]). Previous results (e.g. [43, 57]) suggest that $p_\star$ depends polynomially on d. Our main theorem contains a similar result as a corollary: for any target state $|\Psi^\star\rangle$, construct the VQE instance with $\mathbf{M} = -|\Psi^\star\rangle\langle\Psi^\star|$. Since the problem Hamiltonian is well-conditioned with $\kappa = 1$, according to Theorem 3.3, it suffices to have $\mathsf{poly}(d, 1)$ parameters in order for gradient descent to find a good approximation of $|\Psi^\star\rangle$. Since the choice of $|\Psi^\star\rangle$ is arbitrary, this indicates the $p_\star$ is $O(\mathsf{poly}(d))$.

3.5 VQE Convergence under Noisy Gradients

So far we have assumed perfect access to the exact gradient $\nabla L = (\frac{\partial L}{\partial \theta_1}, \frac{\partial L}{\partial \theta_2}, \cdots, \frac{\partial L}{\partial \theta_p})^T$. In practical NISQ settings, the estimation of gradients is noisy either due to the finite number of measurements, or to the noisy implementation of circuits. In this section, we extend the convergence theorem and show that for sufficiently small amount of noise, the efficient convergence remains. We comment that, while the noise level required in the following theorem depends polynomially on $1/d$ and is still not practical for NISQ settings, our result is the first rigorous result to establish the convergence of VQE in the noisy setting. In addition, the result in this section suggest that the convergence theorem is robust, and reveal the dependency of the noise level on the approximation error $1 - |\langle\Psi(t)|\Psi^\star\rangle|^2$.

In the continuous-time setting we consider the following definition for noisy gradient flow:

Definition 3.4 (Noisy gradient flow)**.** For loss function $L : \mathbb{R}^p \to \mathbb{R}$, the noisy gradient flow on the parameters $\boldsymbol{\theta} \in \mathbb{R}^p$ with learning rate η is defined as

$$\frac{d\boldsymbol{\theta}}{dt} = -\eta(\nabla L + \boldsymbol{\varepsilon}) \quad \text{or} \quad \frac{d\theta_l}{dt} = -\eta(\frac{\partial L}{\partial \theta_l} + \varepsilon_l(t)) \quad \forall l \in [p] \tag{3.15}$$

where $\boldsymbol{\varepsilon}(t) := (\varepsilon_1(t), \cdots, \varepsilon_p(t))^T$ is the noise to the gradient estimation.

The following noisy version of the convergence theorem states that when the ℓ_∞-norm of $\boldsymbol{\varepsilon}(t)$ is sufficiently small, the convergence result still holds:

Corollary 3.4 (Convergence theorem with noisy gradient)**.** *Consider training a p-parameter d-dimensional VQE instance $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ with learning rate $\eta = \frac{1}{pZ(\mathbf{H},d)}$, where the ansatz $\mathbf{U}$ is generated by $\mathbf{H}$ as described in Definition 3.3. Let $\lambda_1 < \lambda_2 \leq \cdots \leq \lambda_d$ denote the eigenvalues of $\mathbf{M}$, and $|\Psi^\star\rangle$ denote the ground state. If*

- *the number of parameters p greater than a threshold of order $O\big((\frac{\lambda_d-\lambda_1}{\lambda_2-\lambda_1})^4, \frac{d^4}{Z(\mathbf{H},d)^3}, \log(d)\big)$,*
- *the gradient estimation error $\|\boldsymbol{\varepsilon}(t)\|_\infty \leq c' \cdot \frac{Z}{\|\mathbf{H}\|_{\mathsf{op}}}(\lambda_2 - \lambda_1)\sqrt{1-|\langle\Psi(t)|\Psi^\star\rangle|^2}|\langle\Psi(t)|\Psi^\star\rangle|$, for some constant c',*

then with probability ≥ 0.99, the output state $|\Psi(t)\rangle$ converges under noisy gradient to the ground state with error $\epsilon := 1 - |\langle\Psi(T_\epsilon)|\Psi^\star\rangle|^2$ in time $T_\epsilon = O\big(\frac{\log d}{\lambda_2-\lambda_1}\log\frac{1}{\epsilon}\big)$. The success probability can be boosted to $1-\delta$ for any $0 < \delta < 1$ using $O\left(\log\left(\frac{1}{\delta}\right)\right)$ random restarts.

Remark 3.1. To interpret the upper bound on $\|\boldsymbol{\varepsilon}\|_\infty$, notice that

$$\sqrt{1-|\langle\Psi(t)|\Psi^\star\rangle|^2}|\langle\Psi(t)|\Psi^\star\rangle| \leq \max\{|\langle\Psi(t)|\Psi^\star\rangle|^2, 1-|\langle\Psi(t)|\Psi^\star\rangle|^2\}. \tag{3.16}$$

At the initial stage of training, $\|\boldsymbol{\varepsilon}\|_\infty$ need to be $O(|\langle\Psi(t)|\Psi^\star\rangle|^2)$ so that the worst-case perturbation in the gradient does not eliminate the overlap between $|\Psi(t)\rangle$ and $|\Psi^\star\rangle$; at the final stage of training $\|\boldsymbol{\varepsilon}\|_\infty$ need to be $O(1-|\langle\Psi(t)|\Psi^\star\rangle|^2)$ to obtain solutions with high quality.

Remark 3.2. The premise of Corollary 3.4 requires $\|\boldsymbol{\varepsilon}(t)\|_\infty / \|\mathbf{H}\|_{\mathsf{op}}$ to be of order $Z/\|\mathbf{H}\|_{\mathsf{op}}^2$, which depends polynomially on $1/d$. We highlight that our analysis here considers the worst-case

(or adversarial) perturbation on the gradient. It is possible that the requirement on $\|\varepsilon\|_\infty$ can be further relaxed in the practical scenerio. For example, when the noise is purely due to the finite measurements, we can further assume ε to be stochastic and unbiased.

The proof of Corollary 3.4 follows directly from the following Lemma 3.5, which calculates the dynamics at the presence of gradient noise, and Lemma 3.6, which states the convergence of the noisy dynamics. The proofs for the lemmas are based on the proofs for Lemma 3.11 and 3.12 and are postponed to Section 3.12.

Lemma 3.5 (Output-state dynamics with noisy gradient estimation)**.** *Consider VQE instance* $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$*, with* $\mathbf{U}$ *being the ansatz defined in Definition 3.3. Under gradient flow with learning rate* η *and noisy gradient estimation* $\nabla L + \varepsilon(t) = \left(\frac{\partial L}{\partial \theta_l} + \varepsilon_l(t)\right)_{l\in[p]}$*, the output state* $|\Psi(t)\rangle$ *follow the dynamics*

$$\frac{d}{dt}|\Psi(t)\rangle = -(\eta \cdot p \cdot Z(\mathbf{H}, d)) \operatorname{tr}_1(\mathbf{Y}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|] \otimes \mathbf{I}_{d\times d}))|\Psi(t)\rangle + \eta \sum_{l=1}^{p} i\varepsilon_l \mathbf{H}_l |\Psi(t)\rangle. \tag{3.17}$$

Here $\mathbf{H}_l$ *are function of* $\boldsymbol{\theta}(t)$*, defined as* $\mathbf{U}_{l:p}(\boldsymbol{\theta}(t))\mathbf{H}\mathbf{U}_{l:p}^\dagger(\boldsymbol{\theta}(t))$ *for all* $l \in [p]$*, and* $\mathbf{Y}$ *is defined as* $\frac{1}{pZ(\mathbf{H},d)} \sum_{l=1}^{p} \mathbf{H}_l^{\otimes 2}$.

The following modified version of Lemma 3.12 implies that the main theorem holds with noisy gradient estimation:

Lemma 3.6 (VQE perturbation lemma under noisy gradients)**.** *If*

- *the output state at initialization* $|\Psi(0)\rangle$ *has non-negligible overlap with the ground state* $|\Psi^\star\rangle$*:* $|\langle\Psi(0)|\Psi^\star\rangle|^2 \geq \Omega(\frac{1}{d})$,

- *for all* $0 \le t \le T$, $\|\mathbf{Y}(t) - \mathbf{Y}^\star(t)\|_{\mathsf{op}} \le O\big(\frac{\lambda_2-\lambda_1}{\lambda_d-\lambda_1} \cdot \frac{1}{d}\big)$,
- *for all* $0 \le t \le T$, $\|\boldsymbol{\varepsilon}(t)\|_\infty \le c' \frac{Z}{\|\mathbf{H}\|_{\mathsf{op}}}(\lambda_2 - \lambda_1)\sqrt{1 - |\langle\Psi(t)|\Psi^\star\rangle|^2}|\langle\Psi(t)|\Psi^\star\rangle|$ *for some positive constant* c',

then under the dynamics

$$\frac{d}{dt}|\Psi(t)\rangle = -\operatorname{tr}_1(\mathbf{Y}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|] \otimes \mathbf{I}_{d\times d}))|\Psi(t)\rangle + \frac{1}{pZ}\sum_{l=1}^{p} i\varepsilon_l \mathbf{H}_l|\Psi(t)\rangle, \tag{3.18}$$

the output states converges to the ground state such that for all $0 \le t \le T$*:*

$$1 - |\langle\Psi(t)|\Psi^\star\rangle|^2 \le \exp(-c\frac{\lambda_2 - \lambda_1}{\log d}t), \text{ for some constant } c. \tag{3.19}$$

3.6 Ansatz-dependent Result

Theorem 3.3 provides a sufficient condition on the number of classical parameters to ensure a VQE instance to converge with a linear rate. The bound depends on the system dimension d as well as the spectral ratio $\kappa = \frac{\lambda_d-\lambda_1}{\lambda_2-\lambda_1}$. In this section, we develop a tighter ansatz-dependent bound on the over-parameterization threshold. We connect the over-parameterization threshold to the effective dimension d_{eff} and effective spectral ratio κ_{eff}, two ansatz-dependent quantities that will defined later in Definition 3.6.

Given an ansatz design $\mathcal{A}$, recall that $G_{\mathcal{A}}$ is a subgroup of $SU(d)$ associated with $\mathcal{A}$ defined in Section 3.2, containing all the realizable unitary matrices by $\mathcal{A}$ with varying number of layers $L = 0, 1, 2, \cdots$. Fixing the input state $|\Phi\rangle$, if $G_{\mathcal{A}}$ is a proper subgroup of $SU(d)$, output state $|\Psi\rangle = \mathbf{U}|\Phi\rangle$ is restricted to a subspace of $\mathbb{C}^d$, leading to a tighter bound on the number of

parameters for convergence. We now formalize the intuition using the group theory language.

A finite-dimensional representation (W, Π) of a group G is specified by a vector space W and a group homomorphism $\Pi\colon G \to GL(W)$, such that $\Pi(g_1)\Pi(g_2) = \Pi(g_1 g_2)$ for all $g_1, g_2 \in G$. And the representation is said to be *unitary* if $\Pi(g)$ is unitary for all $g \in G$. Apparently, as $G_{\mathcal{A}}$ are composed of unitary matrices, the identity map furnishes a unitary representation of $G_{\mathcal{A}}$ (which we will refer to as the *natural representation*).

An important concept in the group representation theory is *irreducibility*. Given a representation (W, Π) of $G_{\mathcal{A}}$, a subspace $V \subseteq W$ is said to be *invariant* if $\Pi(g)v \in V$ for all $v \in V$ and $g \in G_{\mathcal{A}}$. A representation is further said to be *irreducible* if it has no invariant subspaces other than the trivial subspaces consisting of the empty set $\emptyset$ and the whole space W. We are especially interested in the setting where $G_{\mathcal{A}}$ is reducible, as the reducibility induces a decomposition of the ambient space $\mathcal{H} = \mathbb{C}^d$:

Proposition 3.6.1 (Adapted from [70, Proposition 4.27])**.** *Let G be a group with* unitary *representation Π acting on a vector space W. Then this representation is* completely reducible *i.e. W is isomorphic to a direct sum $V_1 \oplus \cdots \oplus V_m$ where each V_j is an invariant subspace which itself has no non-trivial invariant subspaces.*

By Proposition 3.6.1, the natural representation of $G_{\mathcal{A}}$ induces a decomposition of the state space $\mathcal{H} = V_1 \oplus \cdots \oplus V_m$. We now define the ansatz compatibility and the key quantities d_{eff} and κ_{eff} for a VQE instance $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ using this decomposition.

Definition 3.5 (Compatibility of ansatz)**.** Consider a VQE instance $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ with ansatz design $\mathcal{A}$. Let $\mathcal{H} = V_1 \oplus \cdots \oplus V_m$ be the completely-reduced decomposition induced by the ansatz design $\mathcal{A}$ through the natural representation of $G_{\mathcal{A}}$ and let $|\Psi^\star\rangle$ denote the ground state of

$\mathbf{M}$. The ansatz design $\mathcal{A}$ is said to be compatible with the VQE problem if there exists $j \in [m]$ such that both the input state $|\Phi\rangle$ and the target ground state $|\Psi^\star\rangle$ lie within the invariant subspace V_j. We will drop the subscript j and refer to this subspace as V when there is no ambiguity.

The effective quantities for compatible ansatz can be defined using the invariant subspace:

Definition 3.6 (Effective dimension d_{eff} and effective ratio κ_{eff})**.** Consider a VQE instance $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ with compatible ansatz design $\mathcal{A}$. And let V denote the invariant subspace where the input and the ground state lies with projection $\mathbf{\Pi} = \mathbf{Q}\mathbf{Q}^\dagger$ (here $\mathbf{Q} \in \mathbb{C}^{d \times d_{\mathsf{eff}}}$ is an arbitrary set of orthonormal basis). The effective dimension d_{eff} is defined as the dimension of V. The effective spectrum is defined as the ordered eigenvalues $(\lambda'_1, \cdots, \lambda'_{d_{\mathsf{eff}}})$ of the Hermitian $\mathbf{Q}^\dagger \mathbf{M} \mathbf{Q}$. The effective spectral ratio κ_{eff} is defined as $\frac{\lambda'_{d_{\mathsf{eff}}} - \lambda'_1}{\lambda'_2 - \lambda'_1}$. The effective generating Hamiltonian $\mathbf{H}_{\mathsf{eff}}$ is defined as $\mathbf{Q}^\dagger \mathbf{H} \mathbf{Q}$.

Given the projection $\mathbf{\Pi}$ onto V, the basis $\mathbf{Q}$ is not unique, but allow a $d_{\mathsf{eff}} \times d_{\mathsf{eff}}$ unitary transformation. This does not introduce any ambiguity in the definition of the effective spectrum as unitary transformations does not change the eigenvalues of $\mathbf{Q}^\dagger \mathbf{M} \mathbf{Q}$.

The Killing-Cartan classification indicates that the subgroup $G_{\mathcal{A}}$ restricted on the invariant subspace V must be one of the simple lie groups. Here we focus on the case where the subgroup $G_{\mathcal{A}}$ restricted on the invariant subspace V is a special unitary group $SU(d_{\mathsf{eff}})$. Similar results can be proved for special orthogonal, symplectic group by replacing the integral forumla, which can be found for example in [59]). By definition V is invariant under the action of any operator represented by ansatz $\mathcal{A}$, indicating the dynamics of the output state is restricted to the subspace V spanned by $\mathbf{Q}$. By transforming all the Hamiltonians and the input state by $\mathbf{Q}$ in the proof of Theorem 3.3, we have the following corollary:

Corollary 3.7. *Let* $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ *be a VQE instance using compatible ansatz design* $\mathcal{A}$ *with* d_{eff}*,* κ_{eff}*,* $\mathbf{H}_{\mathsf{eff}}$ *and* $(\lambda'_1, \cdots, \lambda'_{d_{\mathsf{eff}}})$ *as defined in Definition 3.5 and that the distribution of the subgroup* $G_{\mathcal{A}}$ *restricted to the subspace is Haar measure over special unitary matrices. Let* $|\Psi^\star\rangle$ *denote the ground state of* $\mathbf{M}$ *and* $|\Psi(t)\rangle$ *the output state at time* t*. If the number of parameters* p *is greater than or equal to an over-parameterization threshold* $p^\star$ *of order* $O\big(\kappa_{\mathsf{eff}}^4, \frac{d_{\mathsf{eff}}^4}{Z(\mathbf{H}_{\mathsf{eff}}, d_{\mathsf{eff}})^3}, \log(d_{\mathsf{eff}})\big)$*, then with probability* ≥ 0.99*, under gradient flow with learning rate of* $\eta = \frac{1}{pZ(\mathbf{H}_{\mathsf{eff}}, d_{\mathsf{eff}})}$*, the output state converges to the ground state with error* $\epsilon = 1 - |\langle\Psi(t)|\Psi^\star\rangle|^2$ *in time* $T_\epsilon = O\big(\frac{\log d_{\mathsf{eff}}}{\lambda'_2 - \lambda'_1} \log \frac{1}{\epsilon}\big)$*. The success probability may be boosted to* $1 - \delta$ *for any* $0 \leq \delta \leq 1$ *using* $O\left(\log \frac{1}{\delta}\right)$ *random restarts.*

The proof for Corollary 3.7 is postponed to Section 3.13. For general ansatz design $\mathcal{A}$ including HEA with $G_{\mathcal{A}} = SU(d)$, the effective dimension d_{eff} (resp. effective ratio κ_{eff}) is the same as the system dimension d (resp. the ratio $\kappa = \frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}$). In fact this is the case for fully-trainable ansatz that contain universal gate sets and satisfy the premise of [45]. On the other hand, a problem-specific compatible ansatz design can have much smaller d_{eff} and κ_{eff} and achieve reasonable performance with much fewer number of parameters. As we see in Section 5.1, for physical problems like transverse field Ising models and Heinsenberg models, certain HVA designs can have d_{eff} and κ_{eff} orders of magnitudes smaller than d and κ.

3.7 Conclusion

Our results establish a starting point for the rigorous analysis of training in VQAs and show how such analysis provides a principled way for their designs. In this section, we discuss possible related directions as well as extensions of the work that could lead to a deeper understanding

of the VQE training, or lead to applications where the quantum advantage is better motivated theoretically. We study VQEs using a specialized parameterization (the partially-trainable ansatz). This ansatz can be seen theoretically and empirically to effectively mimic the convergence behavior of common practical parameterization. Exploring other equivalent parameterizations may be of interest to directly establish tighter over-parameterization bounds. Our current analysis does not yield a direct lower bound on the minimum number of parameters required to ensure convergence so we cannot be sure if our bounds are tight. Our empirical analysis indicates that the theoretical bounds could be improved, leading to more practically feasible thresholds for over-parameterization. Obtaining a critical point between the over and under parameterized regimes would be ideal, to obtain a complete theoretical characterization. In particular, the current sufficient over-parameterization threshold is exponential in the effective dimension and spectral ratio, and reduction. This is not surprising, since a universal polynomial bound would yield polynomial time quantum solutions to some combinatorial optimization problems that are not expected to be efficiently solvable in general. It is thus very important to study *structured* ansatz for particular problems arising from physics or quantum chemistry, where it is possible that the effective dimension itself could be polynomial in the number of qubits, leading to polynomial time quantum algorithms in these settings.

3.8 Proof: Technical Lemmas

We start our proof by laying out three technical helper lemmas:

Lemma 3.8. *Let* $\mathbf{x}, \mathbf{v}$ *be two unit vectors in* $\mathbb{C}^d$*, the commutator* $i[\mathbf{x}\mathbf{x}^\dagger, \mathbf{v}\mathbf{v}^\dagger]$ *has a pair of eigenvalues* $\pm|\langle \mathbf{x}, \mathbf{v}\rangle|\sqrt{1-|\langle \mathbf{x}, \mathbf{v}\rangle|^2}$.

Proof. Express $\mathbf{x}$ as $\alpha\mathbf{v} + \beta\mathbf{w}$ with unit vectors $\mathbf{w}$ and $\mathbf{v}$ such that $\mathbf{w}^\dagger\mathbf{v} = 0$:

$$i[\mathbf{x}\mathbf{x}^\dagger, \mathbf{v}\mathbf{v}^\dagger] = i\alpha^*\beta\mathbf{w}\mathbf{v}^\dagger - i\alpha\beta^*\mathbf{v}\mathbf{w}^\dagger. \tag{3.20}$$

This rank-2 Hermitian has two real eigenvalues λ_+ and λ_- such that $\lambda_+ + \lambda_- = 0$ and $\lambda_+\lambda_- = -|\alpha|^2|\beta|^2$. □

Lemma 3.9 (Bounding commutator norms)**.** *Let* $\mathbf{M} := \sum_{j=1}^d \lambda_j \mathbf{v}_j\mathbf{v}_j^\dagger$ *be a* $d\times d$*-Hermitian matrix with eigenvectors* $\{\mathbf{v}_j\}_{j=1}^d$ *and eigenvalues* $\lambda_1 \leq \cdots \leq \lambda_d$*. Let* $\mathbf{x}$ *be a* d*-dimensional unit vector. The Frobenius norm of the commutator* $[\mathbf{M}, \mathbf{x}\mathbf{x}^\dagger]$ *can be bounded in terms of* $|\langle \mathbf{x}, \mathbf{v}_1\rangle|$ *as:*

$$\|[\mathbf{M}, \mathbf{x}\mathbf{x}^\dagger]\|_F \leq \sqrt{2}(\lambda_d - \lambda_1)\sqrt{1 - |\langle \mathbf{x}, \mathbf{v}_1\rangle|^2}. \tag{3.21}$$

Proof. First observe that, for any real value λ, $[\mathbf{M} - \lambda\mathbf{I}, \mathbf{x}\mathbf{x}^\dagger] = [\mathbf{M}, \mathbf{x}\mathbf{x}^\dagger] - \lambda[\mathbf{I}, \mathbf{x}\mathbf{x}^\dagger] = [\mathbf{M}, \mathbf{x}\mathbf{x}^\dagger]$.

Express $\mathbf{x}$ as $\alpha\mathbf{v_1} + \beta\mathbf{w}$, where $\mathbf{v}_1$ is the eigenvector of $\mathbf{M}$ with eigenvalue λ_1 and $\mathbf{w}$ is a

unit vector orthogonal to $\mathbf{v}_1$. We have $|\alpha|^2 = |\langle \mathbf{x}, \mathbf{v}_1 \rangle|^2$ and $|\beta|^2 = 1 - |\langle \mathbf{x}, \mathbf{v}_1 \rangle|^2$.

$$
\begin{aligned}
& \left\| [\mathbf{M}, \mathbf{x}\mathbf{x}^\dagger] \right\|_F^2 && (3.22)\\
=& \left\| [\mathbf{M} - \lambda \mathbf{I}, \mathbf{x}\mathbf{x}^\dagger] \right\|_F^2 && (3.23)\\
=& \operatorname{tr}\left(- [\mathbf{M} - \lambda \mathbf{I}, \mathbf{x}\mathbf{x}^\dagger][\mathbf{M} - \lambda \mathbf{I}, \mathbf{x}\mathbf{x}^\dagger] \right) && (3.24)\\
=& 2\mathbf{x}^\dagger (\mathbf{M} - \lambda \mathbf{I})^2 \mathbf{x} - 2(\mathbf{x}^\dagger (\mathbf{M} - \lambda \mathbf{I}) \mathbf{x})^2 && (3.25)\\
\leq& 2\mathbf{x}^\dagger (\mathbf{M} - \lambda \mathbf{I})^2 \mathbf{x} && (3.26)\\
=& 2|\alpha|^2 \mathbf{v}_1^\dagger (\mathbf{M} - \lambda \mathbf{I})^2 \mathbf{v}_1 + 2|\beta|^2 \mathbf{w}^\dagger (\mathbf{M} - \lambda \mathbf{I})^2 \mathbf{w} \\
& + 2\alpha^* \beta \mathbf{v}_1^\dagger (\mathbf{M} - \lambda \mathbf{I})^2 \mathbf{w} + 2\alpha \beta^* \mathbf{w}^\dagger (\mathbf{M} - \lambda \mathbf{I})^2 \mathbf{v}_1 && (3.27)\\
=& 2|\alpha|^2 \mathbf{v}_1^\dagger (\mathbf{M} - \lambda \mathbf{I})^2 \mathbf{v}_1 + 2|\beta|^2 \mathbf{w}^\dagger (\mathbf{M} - \lambda \mathbf{I})^2 \mathbf{w} && (3.28)\\
=& 2|\alpha|^2 (\lambda_1 - \lambda)^2 + 2|\beta|^2 \mathbf{w}^\dagger (\mathbf{M} - \lambda \mathbf{I})^2 \mathbf{w} && (3.29)
\end{aligned}
$$

Line 3.28 follows from the orthogonality of $\mathbf{w}$ and $\mathbf{v}_1$.

Taking $\lambda = \lambda_1$, the right-hand side is bounded by $2|\beta|^2(\lambda_d - \lambda_1)^2 = 2(\lambda_d - \lambda_1)^2(1 - |\langle \mathbf{x}, \mathbf{v}_1 \rangle|^2)$. □

Lemma 3.10 (Estimation with Taylor expansion)**.** *Let* $\mathbf{V}$ *be a unitary matrix generated by Hermitian* $\mathbf{H}$ *as* $\mathbf{V} = \exp(-i\theta\mathbf{H})$. *For any Hermitian* $\mathbf{K}$

$$
\left\| (\mathbf{V}\mathbf{K}\mathbf{V}^\dagger)^{\otimes 2} - \mathbf{K}^{\otimes 2} \right\|_{\mathsf{op}} \leq 4|\theta| \left\| \mathbf{H} \right\|_{\mathsf{op}} \left\| \mathbf{K} \right\|_{\mathsf{op}}^2 . \tag{3.30}
$$

Proof. The first- and second-order derivatives of $(\mathbf{V}\mathbf{K}\mathbf{V}^\dagger)^{\otimes 2}$ are:

$$\frac{d}{d\theta}(\mathbf{V}\mathbf{K}\mathbf{V}^\dagger)^{\otimes 2} = \mathbf{V}^{\otimes 2}([-i\mathbf{H},\mathbf{K}]\otimes\mathbf{K}+\mathbf{K}\otimes[-i\mathbf{H},\mathbf{K}])(\mathbf{V}^\dagger)^{\otimes 2}, \tag{3.31}$$

$$\frac{d^2}{d\theta^2}(\mathbf{V}\mathbf{K}\mathbf{V}^\dagger)^{\otimes 2} = -\mathbf{V}^{\otimes 2}(2[\mathbf{H},\mathbf{K}]\otimes[\mathbf{H},\mathbf{K}]+[\mathbf{H},[\mathbf{H},\mathbf{K}]]\otimes\mathbf{K}+\mathbf{K}\otimes[\mathbf{H},[\mathbf{H},\mathbf{K}]])(\mathbf{V}^\dagger)^{\otimes 2}. \tag{3.32}$$

Hence

$$\left\|(\mathbf{V}\mathbf{K}\mathbf{V}^\dagger)^{\otimes 2}-\mathbf{K}^{\otimes 2}\right\|_{\mathsf{op}} \tag{3.33}$$

$$=\left\|\int_0^\theta d\theta'(e^{-i(\theta-\theta')\mathbf{H}})^{\otimes 2}([-i\mathbf{H},\mathbf{K}]\otimes\mathbf{K}+\mathbf{K}\otimes[-i\mathbf{H},\mathbf{K}])(e^{i(\theta-\theta')\mathbf{H}})^{\otimes 2}\right\|_{\mathsf{op}} \tag{3.34}$$

$$\leq 4|\theta|\left\|\mathbf{H}\right\|_{\mathsf{op}}\left\|\mathbf{K}\right\|_{\mathsf{op}}^2. \tag{3.35}$$

□

3.9 Proof: Lemma 3.11 and 3.12

In this section, we prove Lemma 3.11 and 3.12 on the characterization of the VQE dynamics.

Lemma 3.11 (VQE dynamics under gradient flow)**.** *Consider a d-dimensional VQE instance $(\mathbf{M},|\Phi\rangle,\mathbf{U})$, with arbitrary input state $|\Phi\rangle$ and initial parameters $\boldsymbol{\theta}(0)$. Let $\mathbf{U}$ be parameterized by $\{\mathbf{H}_1,\cdots,\mathbf{H}_p\}$ as defined in Equation 3.8. Under gradient flow with learning rate $\eta = \frac{d^2-1}{\sum_{l=1}^p \mathrm{tr}(\mathbf{H}_l^2)}$, the output state $|\Psi(t)\rangle = \mathbf{U}(\boldsymbol{\theta}(t))|\Phi\rangle$ follow the dynamics*

$$\frac{d}{dt}|\Psi(t)\rangle = -\Xi_t\big([\mathbf{M},|\Psi(t)\rangle\langle\Psi(t)|]\big)|\Psi(t)\rangle. \tag{3.36}$$

Here $\Xi_t(\cdot)$ is an endomorphism on skew Hermitians defined as

$$\Xi_t(\mathbf{A}) = \frac{d^2-1}{\sum_{l=1}^{p}\operatorname{tr}(\mathbf{H}_l^2)}\sum_{l=1}^{p}\operatorname{tr}(\mathbf{H}_{l:p}(\boldsymbol{\theta}(t))\mathbf{A})\mathbf{H}_{l:p}(\boldsymbol{\theta}(t)) \tag{3.37}$$

with $\mathbf{H}_l(\boldsymbol{\theta}) := \mathbf{U}_{l:p}(\boldsymbol{\theta})\mathbf{H}_l\mathbf{U}_{l:p}(\boldsymbol{\theta})$ defined as $\mathbf{H}_l$ conjugated by

$$\mathbf{U}_{l:p}(\boldsymbol{\theta}) := \exp(-i\theta_p\mathbf{H}_p)\cdots\exp(-i\theta_l\mathbf{H}_l). \tag{3.38}$$

Proof. We start by calculating the gradient of $\mathbf{U}_{r:p}(\boldsymbol{\theta})$ with respect to θ_l. For $r > l$, $\mathbf{U}_{r:p}$ is independent of θ_l, therefore $\partial\mathbf{U}_{r:p}/\partial\theta_l = 0$. For $r \leq l$,

$$\frac{\partial\mathbf{U}_{r:p}(\boldsymbol{\theta})}{\partial\theta_l} = \mathbf{U}_{l:p}(\boldsymbol{\theta})(-i\mathbf{H}_l)\mathbf{U}_{r:l}(\boldsymbol{\theta}) = -i\mathbf{U}_{l:p}(\boldsymbol{\theta})\mathbf{H}_l\mathbf{U}_{l:p}(\boldsymbol{\theta})^\dagger\mathbf{U}_{r:p}(\boldsymbol{\theta}). \tag{3.39}$$

For the rest of the proof we use $\mathbf{U}_{l:p}$ in abbreviation for $\mathbf{U}_{l:p}(\boldsymbol{\theta})$ when there are no ambiguity. The objective function L can be expressed as $\langle\Phi|\mathbf{U}_{1:p}^\dagger\mathbf{M}\mathbf{U}_{l:p}|\Phi\rangle$:

$$\frac{\partial L(\boldsymbol{\theta})}{\partial\theta_l} = \langle\Phi|\frac{\partial}{\partial\theta_l}\mathbf{U}_{1:p}^\dagger\mathbf{M}\mathbf{U}_{1:p}|\Phi\rangle + \langle\Phi|\mathbf{U}_{1:p}^\dagger\mathbf{M}\frac{\partial}{\partial\theta_l}\mathbf{U}_{1:p}|\Phi\rangle \tag{3.40}$$

$$= \langle\Phi|\mathbf{U}_{1:p}^\dagger i[\mathbf{U}_{l:p}\mathbf{H}_l\mathbf{U}_{l:p}^\dagger,\mathbf{M}]\mathbf{U}_{1:p}|\Phi\rangle \tag{3.41}$$

$$= \langle\Psi(t)|i[\mathbf{U}_{l:p}\mathbf{H}_l\mathbf{U}_{l:p}^\dagger,\mathbf{M}]|\Psi(t)\rangle \tag{3.42}$$

$$= i\operatorname{tr}([\mathbf{M},|\Psi(t)\rangle\langle\Psi(t)|]\mathbf{U}_{l:p}\mathbf{H}_l\mathbf{U}_{l:p}^\dagger). \tag{3.43}$$

The third equality follows from that fact that $\mathbf{U}_{1:p}|\Phi\rangle$ is exactly the output state $|\Psi(t)\rangle$. Following

the dynamics of gradient flow with learning rate η:

$$\frac{d\theta_l}{dt} = -\eta \frac{\partial}{\partial \theta_l} L(\boldsymbol{\theta}) = -i\eta \operatorname{tr}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\mathbf{U}_{l:p}\mathbf{H}_l\mathbf{U}_{l:p}^\dagger). \tag{3.44}$$

The dynamics for $\mathbf{U}_{l:p}$ and $|\Psi(t)\rangle$ as functions of $\boldsymbol{\theta}(t)$ are therefore

$$\frac{d}{dt}\mathbf{U}_{l:p} = \sum_{r=l}^{p} \frac{d\theta_r}{dt}\frac{\partial}{\partial\theta_r}\mathbf{U}_{l:p} = -\eta \sum_{r=l}^{p} \operatorname{tr}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\mathbf{U}_{l:p}\mathbf{H}_l\mathbf{U}_{l:p}^\dagger)\mathbf{U}_{l:p}\mathbf{H}_l\mathbf{U}_{l:p}^\dagger\mathbf{U}_{l:p} \tag{3.45}$$

and

$$\begin{aligned}
\frac{d}{dt}|\Psi(t)\rangle &= \frac{d}{dt}\big(\mathbf{U}_{1:p}|\Phi\rangle\big) && (3.46)\\
&= -\eta\Big(\sum_{l=1}^{p} \operatorname{tr}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\mathbf{U}_{l:p}\mathbf{H}_l\mathbf{U}_{l:p}^\dagger)\mathbf{U}_{l:p}\mathbf{H}_l\mathbf{U}_{l:p}^\dagger\Big)\mathbf{U}_{1:p}|\Phi\rangle && (3.47)\\
&= -\Xi_t([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|])|\Psi(t)\rangle && (3.48)
\end{aligned}$$

□

Lemma 3.12 (Robustness of RGF convergence for VQE)**.** *Consider a d-dimensional VQE instance $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ with $\mathbf{U}$ parameterized by $\{\mathbf{H}_1, \cdots, \mathbf{H}_p\}$ as defined in Equation 3.8. Let $\lambda_1 < \lambda_2 \leq \cdots \leq \lambda_d$ be the eigenvalues of $\mathbf{M}$.*

If (1) the output state at initialization $|\Psi(0)\rangle$ has non-negligible overlap with the target ground state $|\Psi^\star\rangle$, such that $|\langle\Psi(0)|\Psi^\star\rangle|^2 \geq \Omega(\frac{1}{d})$, and (2) for all $t \in [0, T]$, $\|\Xi_t - \mathsf{id}\|_{\infty,1} \leq O(\frac{\lambda_2-\lambda_1}{\lambda_d-\lambda_1} \cdot \frac{1}{d})$, then under the dynamics $\frac{d}{dt}|\Psi(t)\rangle = -\Xi_t([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|])|\Psi(t)\rangle$, the output state converges to the ground state efficiently as $1 - |\langle\Psi(t)|\Psi^\star\rangle|^2 \leq \exp(-c\frac{\lambda_2-\lambda_1}{\log d}t)$ for some constant c and $t \in [0, T]$.

Proof for Lemma 3.12. Define the potential h as $|\langle\Psi^\star|\Psi(t)\rangle|^2$. The time derivative of h is $\frac{dh}{dt} = \mathrm{tr}\left(|\Psi^\star\rangle\langle\Psi^\star|(\frac{d}{dt}|\Psi\rangle\langle\Psi| + |\Psi\rangle(\frac{d}{dt}|\Psi\rangle)^\dagger)\right)$.

Plug in $\frac{d}{dt}|\Psi\rangle = -\Xi_t([\mathbf{M}, |\Psi\rangle\langle\Psi|])|\Psi\rangle = -[\mathbf{M}, |\Psi\rangle\langle\Psi|]|\Psi\rangle - (\Xi_t - \mathsf{id})([\mathbf{M}, |\Psi\rangle\langle\Psi|])|\Psi\rangle$:

$$\frac{d}{dt}h = (\frac{d}{dt}|\Psi(t)\rangle)^\dagger|\Psi^\star\rangle\langle\Psi^\star|\Psi(t)\rangle + \langle\Psi(t)||\Psi^\star\rangle\langle\Psi^\star|\frac{d}{dt}|\Psi(t)\rangle \tag{3.49}$$

$$= 2(\langle\Psi(t)|\mathbf{M}|\Psi(t)\rangle - \lambda_1)|\langle\Psi^\star|\Psi(t)\rangle|^2 + \mathrm{tr}(E(t)[|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]). \tag{3.50}$$

where $E_t := (\Xi_t - \mathsf{id})([\mathbf{M}, |\Psi\rangle\langle\Psi|])$. The first term in Line (3.50) corresponds to the exact Riemannian gradient flow on the unit sphere:

$$2(\langle\Psi(t)|\mathbf{M}|\Psi(t)\rangle - \lambda_1)|\langle\Psi^\star|\Psi(t)\rangle|^2 = 2(\langle\Psi(t)|\mathbf{M}|\Psi(t)\rangle - \lambda_1)h \tag{3.51}$$

$$\geq 2((1-h)\lambda_2 + h\lambda_1 - \lambda_1)h \tag{3.52}$$

$$= 2(\lambda_2 - \lambda_1)(1-h)h \tag{3.53}$$

The second term in Line (3.50) stems from the deviation of Ξ_t from the identity map id:

$$\mathrm{tr}(E_t[|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]) \tag{3.54}$$

$$\geq -\|E_t\|_{\mathsf{op}}\,\|[|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]\|_{\mathrm{tr}} \tag{3.55}$$

$$\geq -\|\Xi_t - \mathsf{id}\|_{\infty,1}\,\|[\mathbf{M}, |\Psi\rangle\langle\Psi|]\|_{\mathrm{tr}}\,\|[|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]\|_{\mathrm{tr}} \tag{3.56}$$

$$\geq -\sqrt{d}\|\Xi_t - \mathsf{id}\|_{\infty,1}\,\|[\mathbf{M}, |\Psi\rangle\langle\Psi|]\|_F\,\|[|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]\|_{\mathrm{tr}} \tag{3.57}$$

$$\geq -2\sqrt{2}\sqrt{d}\sqrt{h}(1-h)(\lambda_d - \lambda_1)\|\Xi_t - \mathsf{id}\|_{\infty,1}. \tag{3.58}$$

For the last inequality, we apply Lemma 3.8 and 3.9.

Combining the two terms, we can lower bound the time derivative of h as

$$\frac{d}{dt}h \geq 2(\lambda_2 - \lambda_1)(1-h)h(1 - \sqrt{2d}\frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}\|\Xi_t - \mathsf{id}\|_{\infty,1}\frac{1}{\sqrt{h}})), \tag{3.59}$$

or by dividing both sides by a negative number $-h$:

$$\frac{d}{dt}(-\ln h) \leq -2(\lambda_2 - \lambda_1)(1-h)(1 - \sqrt{2d}\frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}\|\Xi_t - \mathsf{id}\|_{\infty,1}\frac{1}{\sqrt{h}})). \tag{3.60}$$

Dividing both sides by a positive number $-\ln h$:

$$\begin{aligned}\frac{d}{dt}\ln(-\ln h) &\leq -2(\lambda_2 - \lambda_1)\frac{1-h}{-\ln h}\cdot(1 - \sqrt{2d}\frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}\|\Xi_t - \mathsf{id}\|_{\infty,1}\frac{1}{\sqrt{h}})) &(3.61)\\ &\leq -2(\lambda_2 - \lambda_1)\frac{1}{1-\ln h}\cdot(1 - \sqrt{2d}\frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}\|\Xi_t - \mathsf{id}\|_{\infty,1}\frac{1}{\sqrt{h}})) &(3.62)\end{aligned}$$

where the second inequality follows from the fact that $\frac{1-h}{-\ln h} \geq \frac{1}{1-\ln h}$ for $h \in (0,1)$ (adaptive from the technical Lemma 4 in [67]). For Ξ_t such that $1 - \sqrt{2d}\frac{\lambda_d-\lambda_1}{\lambda_2-\lambda_1}\|\Xi_t - \mathsf{id}\|_{\infty,1}\frac{1}{\sqrt{h}}$ is positive, h is non-decreasing, meaning that $h(t) \geq h(0)$ for all t. Conditioned on $h(0) \geq \Omega(\frac{1}{d})$ at initialization, there exists a pair of constants C_0, c such that if $\|\Xi_t - \mathsf{id}\|_{\infty,1} \leq \frac{C_0}{d}\frac{\lambda_2-\lambda_1}{\lambda_d-\lambda_1}$ for all t, $1 - h(t) \leq -\ln h(t) \leq \exp(-c\frac{\lambda_2-\lambda_1}{\log d}t)$. □

3.10 Proof: Characterization of Ξ_t

In this section, we derive an upperbound on the deviation of $\Xi_t(\cdot)$ from the identity map $\mathsf{id}(\cdot)$, which can be computed without solving the constrained maximization problems in the definition of the operator norm. We start by repeating some notations. Let $\otimes$ denote the Kronecker

product for Hilbert spaces, matrices and vectors. For any Hilbert space $\mathcal{H}$ with dimension d, let $\mathbf{W}_{d^2\otimes d^2}$ and $\mathbf{I}_{d^2\times d^2}$ denote the swap matrix and identity matrix acting on the Hilbert space $\mathcal{H}\otimes\mathcal{H}$. More concretely, let $\{\mathbf{e}_a\}_{a\in[d]}$ be any orthonormal basis of $\mathcal{H}$. $\{\mathbf{e}_a\otimes\mathbf{e}_b\}_{a,b\in[d]}$ then forms a basis of $\mathcal{H}\otimes\mathcal{H}$. The swap matrix and identity matrix are defined as $\mathbf{W}_{d^2\times d^2} := \sum_{a,b\in[d]} \mathbf{e}_a\mathbf{e}_b^\dagger \otimes \mathbf{e}_b\mathbf{e}_a^\dagger$ and $\mathbf{I}_{d^2\times d^2} := \sum_{a,b\in[d]} \mathbf{e}_a\mathbf{e}_a^\dagger \otimes \mathbf{e}_b\mathbf{e}_b^\dagger$. The subscripts are dropped when there is no ambiguity. For any operator on $\mathcal{H}\otimes\mathcal{H}$, the partial trace $\mathrm{tr}_1(\cdot)$ is a linear function defined such that $\mathrm{tr}_1(\mathbf{A}\otimes\mathbf{B}) = \mathrm{tr}(\mathbf{A})\mathbf{B}$ for any operators $\mathbf{A}$ and $\mathbf{B}$ on $\mathcal{H}$.

Recall that Ξ_t is a function of time t through the parameters $\boldsymbol{\theta}(t)$, defined as

$$\Xi_t(\mathbf{A}) := \frac{d^2-1}{\sum_{l=1}^p \mathrm{tr}(\mathbf{H}_l^2)} \sum_{l=1}^p \mathrm{tr}(\mathbf{H}_{l:p}(\boldsymbol{\theta}(t))\mathbf{A})\mathbf{H}_{l:p}(\boldsymbol{\theta}(t)) \tag{3.63}$$

with $\mathbf{H}_l(\boldsymbol{\theta}) := \mathbf{U}_{l:p}(\boldsymbol{\theta})\mathbf{H}_l\mathbf{U}_{l:p}(\boldsymbol{\theta})$, which is $\mathbf{H}_l$ conjugated by

$$\mathbf{U}_{l:p}(\boldsymbol{\theta}) := \exp(-i\theta_p\mathbf{H}_p)\cdots\exp(-i\theta_{l+1}\mathbf{H}_{l+1}). \tag{3.64}$$

The following lemma shows that it suffices to bound $\|\mathbf{Y}_t - \mathbf{Y}^\star\|_{\mathsf{op}}$, with

$$\mathbf{Y}_t := \frac{d^2-1}{\sum_{l=1}^p \mathrm{tr}(\mathbf{H}_l^2)} \sum_{l=1}^p \mathbf{H}_{l:p}(\boldsymbol{\theta}(t))\otimes\mathbf{H}_{l:p}(\boldsymbol{\theta}(t)), \quad \mathbf{Y}^\star = \mathbf{W}_{d^2\times d^2} - \frac{1}{d}\mathbf{I}_{d^2\times d^2}. \tag{3.65}$$

Lemma 3.13 (Helper lemma: characterization of $\Xi_t(\cdot)$)**.** *Let $\{\mathbf{H}_1,\cdots,\mathbf{H}_p\}$ be a set of $d\times d$ traceless Hermitians, and let $\Xi(\cdot)$ be an endomorphism on skew Hermitians defined as $\Xi(\mathbf{A}) =$*

$\frac{d^2-1}{\sum_{l=1}^p \mathrm{tr}(\mathbf{H}_l^2)} \sum_{l=1}^p \mathrm{tr}(\mathbf{H}_l\mathbf{A})\mathbf{H}_l$:

$$\|\Xi - \mathsf{id}\|_{\infty,1} \leq \left\| \frac{d^2-1}{\sum_{l=1}^p \mathrm{tr}(\mathbf{H}_l)} \sum_{l=1}^p \mathbf{H}_l \otimes \mathbf{H}_l - \left(\mathbf{W} - \frac{1}{d}\mathbf{I}_{d\times d}\right) \right\|_{\mathsf{op}}. \tag{3.66}$$

Proof. First observe that, for any skew Hermitian (and therefore traceless) $\mathbf{A}$,

$$\begin{aligned}
&\mathrm{tr}_1\left((\mathbf{W}_{d^2\times d^2} - \frac{1}{d}\mathbf{I}_{d^2\otimes d^2})(\mathbf{A}\otimes \mathbf{I}_{d\times d})\right) && (3.67)\\
=&\mathrm{tr}_1\left((\sum_{a,b\in[d]} \mathbf{e}_a\mathbf{e}_b^\dagger \otimes \mathbf{e}_b\mathbf{e}_a^\dagger)(\mathbf{A}\otimes \mathbf{I}_{d\times d})\right) - \frac{1}{d}\mathrm{tr}_1\left((\sum_{a,b\in[d]} \mathbf{e}_a\mathbf{e}_a^\dagger \otimes \mathbf{e}_b\mathbf{e}_b^\dagger)(\mathbf{A}\otimes \mathbf{I}_{d\times d})\right) && (3.68)\\
=&\sum_{a,b\in[d]} \mathrm{tr}_1\left((\mathbf{e}_a\mathbf{e}_b^\dagger\mathbf{A})\otimes(\mathbf{e}_b\mathbf{e}_a^\dagger)\right) - \frac{1}{d}\sum_{a,b\in[d]} \mathrm{tr}_1\left((\mathbf{e}_a\mathbf{e}_a^\dagger\mathbf{A})\otimes(\mathbf{e}_b\mathbf{e}_b^\dagger)\right) && (3.69)\\
=&\sum_{a,b\in[d]} (\mathbf{A})_{ba}\mathbf{e}_b\mathbf{e}_a^\dagger - \frac{1}{d}\sum_{a,b\in[d]} (\mathbf{A})_{aa}\mathbf{e}_b\mathbf{e}_b^\dagger && (3.70)\\
=&\mathbf{A} - \frac{\mathrm{tr}(\mathbf{A})}{d}\mathbf{I}_{d\otimes d} && (3.71)\\
=&\mathbf{A}. && (3.72)
\end{aligned}$$

The last equality follows from the fact that $\mathbf{A}$ is traceless. Similar calculation shows that:

$$\mathrm{tr}_1\left((\frac{d^2-1}{\sum_{l=1}^p \mathrm{tr}(\mathbf{H}_l)}\sum_{l=1}^p \mathbf{H}_l\otimes\mathbf{H}_l)(\mathbf{A}\otimes\mathbf{I}_{d\times d})\right) = \Xi(\mathbf{A}). \tag{3.73}$$

Let $\mathbf{Y}$ and $\mathbf{Y}^\star$ denote $\frac{d^2-1}{\sum_{l=1}^p \mathrm{tr}(\mathbf{H}_l)}\sum_{l=1}^p \mathbf{H}_l\otimes\mathbf{H}_l$ and $\mathbf{W} - \frac{1}{d}\mathbf{I}_{d\times d}$ respectively. For any skew

Hermitian $\mathbf{A}$:

$$\|\Xi(\mathbf{A}) - \mathsf{id}(\mathbf{A})\|_{\mathsf{op}} \tag{3.74}$$

$$= \left\| \mathrm{tr}_1\left((\mathbf{A}\otimes \mathbf{I}_{d\times d})(\mathbf{Y}-\mathbf{Y}^\star)\right)\right\|_{\mathsf{op}} \tag{3.75}$$

$$= \max_{\mathbf{x}\in\mathcal{H},\mathbf{x}^\dagger\mathbf{x}=1} \mathbf{x}^\dagger \,\mathrm{tr}_1\left((\mathbf{A}\otimes \mathbf{I}_{d\times d})(\mathbf{Y}-\mathbf{Y}^\star)\right)\mathbf{x} \tag{3.76}$$

$$= \max_{\mathbf{x}\in\mathcal{H},\mathbf{x}^\dagger\mathbf{x}=1} \mathrm{tr}\left(\mathbf{x}\mathbf{x}^\dagger \,\mathrm{tr}_1\left((\mathbf{A}\otimes \mathbf{I}_{d\times d})(\mathbf{Y}-\mathbf{Y}^\star)\right)\right) \tag{3.77}$$

$$= \max_{\mathbf{x}\in\mathcal{H},\mathbf{x}^\dagger\mathbf{x}=1} \mathrm{tr}\left((\mathbf{I}_{d\times d}\otimes \mathbf{x}\mathbf{x}^\dagger)(\mathbf{A}\otimes \mathbf{I}_{d\times d})(\mathbf{Y}-\mathbf{Y}^\star)\right) \tag{3.78}$$

$$= \max_{\mathbf{x}\in\mathcal{H},\mathbf{x}^\dagger\mathbf{x}=1} \mathrm{tr}\left((\mathbf{A}\otimes \mathbf{x}\mathbf{x}^\dagger)(\mathbf{Y}-\mathbf{Y}^\star)\right) \tag{3.79}$$

$$\leq \max_{\mathbf{x}\in\mathcal{H},\mathbf{x}^\dagger\mathbf{x}=1} \left\|\mathbf{A}\otimes \mathbf{x}\mathbf{x}^\dagger\right\|_{\mathrm{tr}} \|\mathbf{Y}-\mathbf{Y}^\star\|_{\mathsf{op}} \tag{3.80}$$

$$= \|\mathbf{A}\|_{\mathrm{tr}} \|\mathbf{Y}-\mathbf{Y}^\star\|_{\mathsf{op}} \tag{3.81}$$

Hence $\|\Xi - \mathsf{id}\|_{\infty,1} := \max_{\mathbf{A}:\|\mathbf{A}\|_{\mathrm{tr}}=1,\mathbf{A}=-\mathbf{A}^\dagger} \|\Xi(\mathbf{A}) - \mathsf{id}(\mathbf{A})\|_{\mathsf{op}} \leq \|\mathbf{Y}-\mathbf{Y}^\star\|_{\mathsf{op}}$.

□

3.11 Proof: Theorem 3.3

In this section, we prove Theorem 3.3 for partially-trainbale ansatze by establishing the premises of Lemma 3.12, namely, for $p \geq p^\star = O(\mathsf{poly}(d,\kappa))$, with high probability over random initializations $\|\Xi_t - \mathsf{id}\|_{\infty,1} \leq O(\frac{\lambda_2-\lambda_1}{\lambda_d-\lambda_1}\cdot\frac{1}{d})$.

Recall that $\mathbf{Y}^\star = \mathbf{W}_{d^2\times d^2} - \frac{1}{d}\mathbf{I}_{d^2\times d^2}$. For partially-trainable ansatze in Definition 3.3:

$$\mathbf{Y}_t := \frac{d^2-1}{p\sum_{l=1}^p \mathrm{tr}(\mathbf{H}^2)} \sum_{l=1}^p \mathbf{H}_{l:p}(\boldsymbol{\theta}(t)) \otimes \mathbf{H}_{l:p}(\boldsymbol{\theta}(t)), \tag{3.82}$$

$$\mathbf{H}_{l:p}(\boldsymbol{\theta}(t)) = \mathbf{U}_p \exp(-i\theta_p(t)) \cdots \mathbf{U}_{l+1} \exp(-i\theta_{l+1}(t)) \mathbf{U}_l \mathbf{H} \mathbf{U}_l^\dagger \cdots \exp(i\theta_p(t)) \mathbf{U}_p^\dagger. \tag{3.83}$$

To highlight the time-dependency of $\mathbf{Y}_t$ through the parameters $\boldsymbol{\theta}(t)$, we use $\mathbf{Y}_t$ and $\mathbf{Y}(\boldsymbol{\theta}(t))$ interchangeably. Lemma 3.14 shows that at initialization, $\mathbf{Y}(\boldsymbol{\theta}(0))$ concentrates around $\mathbf{Y}^\star$. Lemma 3.15 shows that during training, $\mathbf{Y}(\boldsymbol{\theta}(t))$ remains close to $\mathbf{Y}(\boldsymbol{\theta}(0))$.

3.11.1 Concentration at initialization

Lemma 3.14 (Concentration at initialization for VQE)**.** *Over the randomness of ansatz initialization (i.e. for $\{\mathbf{U}_l\}_{l=1}^p$ sampled $i.i.d.$ with respect to the Haar measure), for any initial $\boldsymbol{\theta}(0)$, with probability $1-\delta$:*

$$\|\mathbf{Y}(\boldsymbol{\theta}(0)) - \mathbf{Y}^\star\|_{\mathsf{op}} \le \frac{c_i}{\sqrt{p}} \cdot \frac{\|\mathbf{H}\|_{\mathsf{op}}^2}{Z} \sqrt{\log \frac{d^2}{\delta}}, \tag{3.84}$$

with c_i being a positive constant.

Proof. Define

$$\mathbf{X}_l := \frac{1}{Z(\mathbf{H}, d)} \big(\mathbf{U}_{l:p}(\boldsymbol{\theta}(0)) \mathbf{H} \mathbf{U}_{l:p}^\dagger(\boldsymbol{\theta}(0))\big)^{\otimes 2} - \mathbf{Y}^\star. \tag{3.85}$$

By straight-forward calculation, we know that X_l is centered (i.e $\mathbb{E}[X_l] = 0$). The set $\{\mathbf{X}_l\}$ can be viewed as independent random matrices as the Haar random unitary removes all the

correlation. The matrix on the left-hand side can therefore be expressed as the arithmetic average of p independent random matrices. The square of $\mathbf{X}_l$ is bounded in operator norm:

$$\left\|\mathbf{X}_l^2\right\|_{\mathsf{op}} = \|\mathbf{X}_l\|_{\mathsf{op}}^2 \leq (\frac{\|\mathbf{H}\|_{\mathsf{op}}^2}{Z} + \frac{d+1}{d})^2 \leq (\frac{2\,\|\mathbf{H}\|_{\mathsf{op}}^2}{Z(\mathbf{H},d)})^2 \tag{3.86}$$

where the second inequality follows from the fact that the ratio $g_1 = \|\mathbf{H}\|_{\mathsf{op}}^2 / \operatorname{tr}(\mathbf{H}^2)$ satisfies that $1 \geq g_1 \geq 1/d$. By Hoeffding's inequality([71], Thm 1.3), with probability $\geq 1-\delta$,

$$\|\mathbf{Y}(\boldsymbol{\theta}(0)) - \mathbf{Y}^\star\|_{\mathsf{op}} \leq \frac{c_i}{\sqrt{p}} \cdot \frac{\|\mathbf{H}\|_{\mathsf{op}}^2}{Z(\mathbf{H},d)} \sqrt{\log \frac{d^2}{\delta}}. \tag{3.87}$$

for some constant c_i. □

3.11.2 Concentration during training

We next show that the concentration is maintained throughout the evolution of the dynamics as long as exponential convergence holds.

Lemma 3.15 (Concentration during training (time dependent))**.** *Suppose that under learning rate* $\eta = \frac{1}{pZ(\mathbf{H},d)}$*, for all* $0 \leq t \leq T$*,* $1 - |\langle\Psi|\Psi^\star\rangle|^2 \leq \exp(-c\frac{(\lambda_2-\lambda_1)}{\log d}t)$*, then with probability* $\geq 1-\delta$*, for all* $0 \leq t \leq T$*:*

$$\|\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}(\boldsymbol{\theta}(0))\|_{\mathsf{op}} \leq C_3 \left(\frac{T}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)^3}} \left(1 + \sqrt{\log\left(\frac{2d}{\delta}\right)}\right)\right), \tag{3.88}$$

where C_3 *is a constant.*

To prove that $\mathbf{Y}$ concentrates to its expected value throughout training upto any point in time until which the linear convergence condition holds on the gradient flow dynamics. The proof will be based on two main ideas:

1. The linear convergence of the gradient flow dynamics allows the deviation of the parameters $\boldsymbol{\theta}$ from their initial values to be bounded in terms of the evolution time (See Lemma 3.16).

2. The random variables $\mathbf{Y}(\boldsymbol{\theta}(t))$ for different times t form a *random field*, whose deviations $\mathbf{Y}(\boldsymbol{\theta}(t_1)) - \mathbf{Y}(\boldsymbol{\theta}(t_1))$ we show to be bounded by a quantity proportional to $|t_1 - t_2|/\sqrt{p}$. We then use Dudley's lemma [72, Theorem 8.1.6] on the concentration of random fields to bound the supremum of the deviation from initialization over time.

We first show a result connecting the evolution time to the corresponding deviation in $\boldsymbol{\theta}$.

Lemma 3.16 (Slow-varying θ)**.** *Suppose that under learning rate* $\eta = \frac{1}{pZ(\mathbf{H},d)}$*, for all* $0 \leq t \leq T$*,* $1 - |\langle \Psi(t)|\Psi^\star\rangle|^2 \leq \exp(-c\frac{(\lambda_2-\lambda_1)}{\log d}t)$*, then for all* $0 \leq t_1, t_2 \leq T$*:*

$$\|\boldsymbol{\theta}(t_2) - \boldsymbol{\theta}(t_1)\|_\infty \leq \frac{1}{p} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)}} \cdot |t_2 - t_1|, \tag{3.89}$$

$$\|\boldsymbol{\theta}(t_2) - \boldsymbol{\theta}(t_1)\|_2 \leq \frac{1}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)}} \cdot |t_2 - t_1|. \tag{3.90}$$

Proof. Let $\mathbf{J}(t) := [\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]$ and $\mathbf{H}_l := \mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger$. Recall that

$$\frac{d\theta_l}{dt} = -\frac{1}{pZ(\mathbf{H},d)} \operatorname{tr}(i\mathbf{J}(t)\mathbf{H}_l(t)), \tag{3.91}$$

$$|\theta_l(t_2) - \theta_l(t_1)| = |\int_{t_1}^{t_2} \frac{d\theta_l(t)}{dt} dt| = \frac{1}{pZ} |\int_{t_1}^{t_2} \mathrm{tr}(\mathbf{H}_l(t)\mathbf{J}(t)dt| \tag{3.92}$$

$$\leq \frac{1}{pZ} \|\mathbf{H}(t)\|_F \int_{t_1}^{t_2} \|\mathbf{J}(t)\|_F dt \tag{3.93}$$

$$\leq \frac{1}{pZ} \sqrt{\mathrm{tr}(\mathbf{H}^2)} \int_{t_1}^{t_2} \sqrt{2}(\lambda_d - \lambda_1) e^{-\frac{c}{2}\frac{\lambda_2 - \lambda_1}{\log d} t} dt \tag{3.94}$$

$$= 2\sqrt{2} \frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1} \sqrt{\frac{d^2 - 1}{p^2 Z}} (\log d/c) \left(e^{-\frac{c}{2}\frac{\lambda_2 - \lambda_1}{\log d} t_1} - e^{-\frac{c}{2}\frac{\lambda_2 - \lambda_1}{4 \log d} t_2} \right) \tag{3.95}$$

$$\leq \frac{1}{p} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2 - 1}{Z(\mathbf{H}, d)}} \cdot |t_2 - t_1| \cdot e^{-\frac{c}{2}\frac{\lambda_2 - \lambda_1}{\log d} t_1} \tag{3.96}$$

$$\leq \frac{1}{p} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2 - 1}{Z(\mathbf{H}, d)}} \cdot |t_2 - t_1|. \tag{3.97}$$

Here we use the fact that $\|\mathbf{J}\|_F \leq \sqrt{2}(\lambda_d - \lambda_1)\sqrt{1 - |\langle \Psi(t) | \Psi^\star \rangle|^2}$, following technical Lemma 3.9.

□

We next consider the random variables $\mathbf{Y}(\boldsymbol{\theta}(t))$ for t in some interval $[0, T]$. These variables form a random field and we show a concentration inequality on the expected deviation in $\mathbf{Y}(\boldsymbol{\theta}(t))$ over different intervals. A random variable $\mathbf{X}$ is said to be *sub-gaussian* [72, Proposition 2.5.2] if its tails satisfy $\mathbf{Pr}[\mathbf{X} \geq t] \leq 2\exp(-t^2/K_1^2)$ for some K_1. The largest K_1 satifying this relation is defined to be the second Orlicz norm, or ψ_2-norm of $\mathbf{X}$.

Lemma 3.17 (Concentration of deviations in $\mathbf{Y}(\boldsymbol{\theta}(t))$)**.**

$$\mathbf{Pr}[\|\mathbf{Y}(\boldsymbol{\theta}(t_2)) - \mathbf{Y}(\boldsymbol{\theta}(t_1))\|_{\mathsf{op}} > t] \leq 2\exp\left(-\frac{-t^2 Z(\mathbf{H}, d)^2}{2C_1 \|\boldsymbol{\theta}(t_2) - \boldsymbol{\theta}(t_1)\|_2^2} \right) \tag{3.98}$$

for some constant C_1.

Proof. We first observe that due to the Haar distribution of the unitaries U_l, $\mathbf{Y}(\boldsymbol{\theta}(t_2)) - \mathbf{Y}(\boldsymbol{\theta}(t_1))$ is distributed identically to $\mathbf{Y}(\boldsymbol{\theta}(t_2) - \boldsymbol{\theta}(t_1)) - \mathbf{Y}(0)$. For convenience we define $\delta\boldsymbol{\theta} = \boldsymbol{\theta}(t_2) -$

$\boldsymbol{\theta}(t_1)$ in the remainder of the proof.

Define $\mathbf{Y}_l(\boldsymbol{\theta}) = \mathbf{H}_l^{\otimes 2}$; then $\mathbf{Y}(\boldsymbol{\theta}) = \frac{1}{pZ(\mathbf{H},d)} \sum_{l=1}^{p} \mathbf{Y}_l$. We consider a re-parameterization of the random variables $\mathbf{H}_l(\theta)$ by constructing random variables that are identically distributed, but are functions on a different latent probability space. Defining $\mathbf{H}_l$ as $\mathbf{U}_p \cdots \mathbf{U}_l \mathbf{H} \mathbf{U}_l^\dagger \cdots \mathbf{U}_p^\dagger$, $\mathbf{Y}$ can be rewritten as:

$$\mathbf{Y}(\boldsymbol{\theta}) = \frac{1}{pZ} \sum_{l=1}^{p} \left(e^{-i\theta_p \mathbf{H}_p} \cdots e^{-i\theta_{l+1} \mathbf{H}_{l+1}} \mathbf{H}_l e^{i\theta_{l+1} \mathbf{H}_{l+1}} \cdots e^{i\theta_p \mathbf{H}_p} \right)^{\otimes 2}. \tag{3.99}$$

By the Haar randomness of $\{\mathbf{U}_l\}_{l=1}^p$, we can view $\{\mathbf{H}_l\}_{l=1}^p$ as random Hermitians generated by $\{\mathbf{V}_l \mathbf{H} \mathbf{V}_l^\dagger\}$ for *i.i.d.* Haar random $\{\mathbf{V}_l\}_{l=1}^p$. This variable is identically distributed to $\mathbf{Y}$ and $\mathbf{Y}_l$ can be defined as each term in the sum.

We will apply the well-known McDiarmid inequality [72, Theorem 2.9.1] that can be stated as follows: Consider independent random variables $X_1, \ldots, X_k \in \mathcal{X}$. Suppose a random variable $\phi \colon \mathcal{X}^k \to \mathbb{R}$ satisfies the condition that for all $1 \leq j \leq k$ and for all $x_1, \ldots, x_j, \ldots, x_k, x_j' \in \mathcal{X}$,

$$|\phi(x_1, \ldots, x_j, \ldots, x_k) - \phi(x_1, \ldots, x_j', \ldots, x_k)| \leq c_j, \tag{3.100}$$

then the tails of the distribution satisfy

$$\mathbf{Pr}[|\phi(X_1, \ldots, X_k) - \mathbb{E}\phi| \geq t] \leq \exp\left(\frac{-2t^2}{\sum_{i=1}^k c_i^2} \right). \tag{3.101}$$

With our earlier re-parameterization we can consider $\mathbf{Y}$ and consequently $\mathbf{Y}_l$ as functions of the randomly sampled Hermitian operators $\mathbf{H}_l$. Define the variable $\mathbf{Y}^{(k)}$ as that obtained by

resampling $\mathbf{H}_k$ independently, and $\mathbf{Y}_l^{(k)}$ correspondingly. Finally we define

$$\Delta^{(k)}\mathbf{Y} = \left\| (\mathbf{Y}(\delta\boldsymbol{\theta}) - \mathbf{Y}(0)) - (\mathbf{Y}^{(k)}(\delta\boldsymbol{\theta}) - \mathbf{Y}^{(k)}(0)) \right\|_{\mathsf{op}} = \left\| \mathbf{Y}(\delta\boldsymbol{\theta}) - \mathbf{Y}^{(k)}(\delta\boldsymbol{\theta}) \right\|_{\mathsf{op}}. \quad (3.102)$$

Via the triangle inequality,

$$\Delta^{(k)}\mathbf{Y} = \|\mathbf{Y}(\delta\boldsymbol{\theta}) - \mathbf{Y}^{(k)}(\delta\boldsymbol{\theta})\| = \frac{1}{pZ}\|\sum_{l=1}^{k} \mathbf{Y}_l(\delta\boldsymbol{\theta}) - \mathbf{Y}_l^{(k)}(\delta\boldsymbol{\theta})\| \quad (3.103)$$

$$\leq \frac{1}{pZ}\sum_{l=1}^{k}\|\mathbf{Y}_l(\delta\boldsymbol{\theta}) - \mathbf{Y}_l^{(k)}(\delta\boldsymbol{\theta})\|. \quad (3.104)$$

Then by definition,

$$\begin{aligned} &\|\mathbf{Y}_l(\delta\boldsymbol{\theta}) - \mathbf{Y}_l^{(k)}(\delta\boldsymbol{\theta})\| \\ =&(e^{-i\delta\boldsymbol{\theta}_p\mathbf{H}_p}\cdots e^{-i\delta\boldsymbol{\theta}_{k+1}\mathbf{H}_{k+1}})^{\otimes 2}\big((e^{-i\delta\boldsymbol{\theta}_k\mathbf{H}_k}\mathbf{K}e^{i\delta\boldsymbol{\theta}_k\mathbf{H}_k})^{\otimes 2} \\ -&(e^{-i\delta\boldsymbol{\theta}_k\mathbf{H}'_k}\mathbf{K}e^{i\delta\boldsymbol{\theta}_k\mathbf{H}'_k})^{\otimes 2}\big)(e^{i\delta\boldsymbol{\theta}_{l+1}\mathbf{H}_{l+1}}\cdots e^{i\delta\boldsymbol{\theta}_p\mathbf{H}_p})^{\otimes 2}. \end{aligned} \quad (3.105)$$

where $\mathbf{K} := e^{-i\delta\boldsymbol{\theta}_{k-1}\mathbf{H}_{k-1}}\cdots e^{-i\delta\boldsymbol{\theta}_{l+1}\mathbf{H}_{l+1}}\mathbf{H}_l e^{i\delta\boldsymbol{\theta}_{l+1}\mathbf{H}_{l+1}}\cdots e^{i\delta\boldsymbol{\theta}_{k-1}\mathbf{H}_{k-1}}$. By Lemma 3.10,

$$\|\big(\mathbf{Y}_l(\delta\boldsymbol{\theta}) - \mathbf{Y}_l(\mathbf{0})\big) - \big(\mathbf{Y}_l^{(k)}(\delta\boldsymbol{\theta}) - \mathbf{Y}_l^{(k)}(\mathbf{0})\big)\| \leq 8\,|\delta\boldsymbol{\theta}_k|\,\|\mathbf{H}\|_{\mathsf{op}}\,\|\mathbf{K}\|_{\mathsf{op}}^2 = 8\,|\delta\boldsymbol{\theta}_k|\,\|\mathbf{H}\|_{\mathsf{op}}^3. \quad (3.106)$$

We finally have $\Delta^{(k)}(y) \leq \frac{8|\delta\boldsymbol{\theta}_k|\|\mathbf{H}\|_{\mathsf{op}}^3}{Z}$. By the McDiarmid inequality, the result follows. □

To bound the supremum of the deviation over an entire time interval, we employ Dudley's integral inequality (stated below in it's matrix form).

Lemma 3.18 (Dudley's integral inequality: subgaussian matrix version (Adapted from Theorem 8.1.6 in [72])**.** *Let $\boldsymbol{\mathcal{R}}$ be a metric space equipped with a metric $\mathbf{d}(\cdot,\cdot)$, and $\mathbf{X} : \boldsymbol{\mathcal{R}} \mapsto \mathbb{R}^{D\times D}$ with subgaussian increments ie. it satisfies*

$$\mathbf{Pr}[\|\mathbf{X}(r_1) - \mathbf{X}(r_2)\|_{\mathsf{op}} > t] \leq 2D \exp\left(-\frac{t^2}{C_\sigma^2 \mathbf{d}(r_1, r_2)^2}\right), \tag{3.107}$$

Then with probability at least $1 - 2D\exp(-u^2)$ for any subset $\mathcal{S} \subseteq \boldsymbol{\mathcal{R}}$:

$$\sup_{(r_1,r_2)\in\mathcal{S}} \|\mathbf{X}(r_1) - \mathbf{X}(r_2)\|_{\mathsf{op}} \leq C \cdot C_\sigma \left[\int_0^{\mathrm{diam}(\mathcal{S})} \sqrt{\mathcal{N}(\mathcal{S}, \mathbf{d}, \epsilon)}\, d\epsilon + u \cdot \mathrm{diam}(\mathcal{S})\right]. \tag{3.108}$$

for some constant C, where $\mathcal{N}(\mathcal{S}, \mathbf{d}, \epsilon)$ is the metric entropy defined as the logarithm of the ϵ-covering number of $\mathcal{S}$ using metric d.

We then have the following main result:

Lemma 3.19 (Concentration during training (time dependent))**.** *Suppose that under learning rate $\eta = \frac{1}{pZ(\mathbf{H},d)}$, for all $0 \leq t \leq T$, $1 - |\langle\Psi|\Psi^\star\rangle|^2 \leq \exp(-c\frac{(\lambda_2-\lambda_1)}{\log d}t)$, then with probability $\geq 1-\delta$, for all $0 \leq t \leq T$:*

$$\|\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}(\boldsymbol{\theta}(0))\|_{\mathsf{op}} \leq C_3 \left(\frac{T}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)^3}} \left(1 + \sqrt{\log\left(\frac{2d}{\delta}\right)}\right)\right), \tag{3.88}$$

where C_3 is a constant.

Proof. Via Lemma 3.17,

$$\mathbf{Pr}[\|\mathbf{Y}(\boldsymbol{\theta}(t_2)) - \mathbf{Y}(\boldsymbol{\theta}(t_1))\|_{\mathsf{op}} > t] \leq 2\exp\left(-\frac{-t^2 Z(\mathbf{H},d)^2}{2C_1\|\boldsymbol{\theta}(t_2)-\boldsymbol{\theta}(t_1)\|^2}\right), \tag{3.109}$$

By Lemma 3.16 $\|\boldsymbol{\theta}(t_2) - \boldsymbol{\theta}(t_1)\|_2 \leq \frac{1}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)}} \cdot |t_2 - t_1|$. Thus, $\mathbf{Y}$ has sub-gaussian increments if we define the metric $\mathbf{d}(t_2, t_1) = \frac{1}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)^3}} \cdot |t_2 - t_1|$, thereby satisfying the conditions for Lemma 3.18. Under this metric, the diameter of the interval $[0, T]$ is $\frac{T}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)^3}}$. Applying Lemma 3.18, with $u = \sqrt{\log(2d/\delta)}$ to ensure a failure probability at most δ we have

$$\sup_{t\in[0,T]} \|\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}(\boldsymbol{\theta}(0))\|_{\mathsf{op}} \leq C_2\left(\int_0^{\mathrm{diam}([0,T])} \epsilon^{-1}\, d\epsilon + \mathrm{diam}([0,T])\sqrt{\log\left(\frac{2d}{\delta}\right)}\right). \tag{3.110}$$

We assume without loss of generality that p is large enough such that $\mathrm{diam}([0, T]) < 1$. Then,

$$\sup_{t\in[0,T]} \|\mathbf{Y}(\theta(t)) - \mathbf{Y}(\theta(0))\|_{\mathsf{op}} \leq C_2\left(\mathrm{diam}([0,T])\left(1 + \sqrt{\log\left(\frac{2d}{\delta}\right)}\right)\right). \tag{3.111}$$

By the previous consideration,

$$\sup_{t\in[0,T]} \|\mathbf{Y}(\theta(t)) - \mathbf{Y}(\theta(0))\|_{\mathsf{op}} \leq C_3\left(\frac{T}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)^3}}\left(1 + \sqrt{\log\left(\frac{2d}{\delta}\right)}\right)\right), \tag{3.112}$$

where C_2, C_3 are constants. □

Given the lemmas above, we now state the proof of Lemma 3.15:

Proof. Via Lemma 3.17,

$$\mathbf{Pr}[\|\mathbf{Y}(\boldsymbol{\theta}(t_2)) - \mathbf{Y}(\boldsymbol{\theta}(t_1))\|_{\mathsf{op}} > t] \leq 2\exp\left(-\frac{-t^2 Z(\mathbf{H},d)^2}{2C_1\|\boldsymbol{\theta}(t_2)-\boldsymbol{\theta}(t_1)\|^2}\right), \tag{3.113}$$

By Lemma 3.16 $\|\boldsymbol{\theta}(t_2) - \boldsymbol{\theta}(t_1)\|_2 \leq \frac{1}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)}} \cdot |t_2 - t_1|$. Thus, $\mathbf{Y}$ has sub-gaussian increments if we define the metric $\mathbf{d}(t_2, t_1) = \frac{1}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)^3}} \cdot |t_2 - t_1|$, thereby satisfying the conditions for Lemma 3.18. Under this metric, the diameter of the interval $[0, T]$ is $\frac{T}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)^3}}$. Applying Lemma 3.18, with $u = \sqrt{\log(2d/\delta)}$ to ensure a failure probability at most δ we have

$$\sup_{t\in[0,T]} \|\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}(\boldsymbol{\theta}(0))\|_{\mathsf{op}} \leq C_2\left(\int_0^{\mathrm{diam}([0,T])} \epsilon^{-1}\, d\epsilon + \mathrm{diam}([0,T])\sqrt{\log\left(\frac{2d}{\delta}\right)}\right). \tag{3.114}$$

We assume without loss of generality that p is large enough such that $\mathrm{diam}([0, T]) < 1$. Then,

$$\sup_{t\in[0,T]} \|\mathbf{Y}(\theta(t)) - \mathbf{Y}(\theta(0))\|_{\mathsf{op}} \leq C_2\left(\mathrm{diam}([0,T])\left(1 + \sqrt{\log\left(\frac{2d}{\delta}\right)}\right)\right). \tag{3.115}$$

By the previous consideration,

$$\sup_{t\in[0,T]} \|\mathbf{Y}(\theta(t)) - \mathbf{Y}(\theta(0))\|_{\mathsf{op}} \leq C_3\left(\frac{T}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2-1}{Z(\mathbf{H},d)^3}}\left(1 + \sqrt{\log\left(\frac{2d}{\delta}\right)}\right)\right), \tag{3.116}$$

where C_2, C_3 are constants. □

3.11.3 Proof of Theorem 3.3

Finally, we can combine our previous results to show that with sufficient over-parameterization, the VQE dynamics can be made to exponentially converge to the ground state

Theorem 3.3 (Exponential convergence of VQE)**.** *Consider a d-dimensional VQE instance $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ with $\mathbf{U}$ generated by $\mathbf{H}$ as in Definition 3.3. Let $\lambda_1 < \lambda_2 \leq \cdots \leq \lambda_d$ be the eigenvalues of $\mathbf{M}$ and $|\Phi^\star\rangle$ be the ground state. If the number of parameters p of order $\mathsf{poly}(d, \kappa)$ with $\kappa := \frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}$, then under gradient flow on $\boldsymbol{\theta}$ with learning rate $\eta = \frac{d^2-1}{p\,\mathrm{tr}(\mathbf{H^2})}$, the output state $|\Psi(t)\rangle$ converges to an ϵ-approximation of the ground state $|\Psi^*\rangle$ such that $\epsilon = 1 - |\langle\Psi(T_\epsilon)|\Psi^\star\rangle|^2$ in time $T_\epsilon = O\big(\frac{\log d}{\lambda_2 - \lambda_1} \log \frac{1}{\epsilon}\big)$, with success probability $2/3$.*

Proof. We first show that the initial output state $|\Psi(0)\rangle$ satisfies the condition $|\langle\Psi(0)|\Psi^\star\rangle|^2 \geq \Omega(1/d)$, required by Lemma 3.12, with high probability. To see this, observe that $|\Psi(0)\rangle$ is obtained by applying a Haar uniform unitary operator to an input vector, therefore, $|\Psi(0)\rangle$ obeys the uniform Haar distribution over the space of quantum states $S(\mathbb{C}^d)$. Due to this uniformity, $|\langle\Psi(0)|\Psi^\star\rangle|^2$ is equidistributed to $|\langle\Psi(0)|1\rangle|^2$. Furthermore, $|\Psi(0)\rangle$ follows the same distribution as a state vector $|w\rangle = \frac{1}{\sum_{j=1}^d w_{j,\mathrm{re}}^2 + w_{j,\mathrm{im}}^2} \left(\sum_{k=1}^d w_{k,\mathrm{re}} + i w_{k,\mathrm{im}}\right)$, where each $w_{j,\mathrm{im}}, w_{j,\mathrm{re}}$ are drawn from independent standard normal distributions. The distribution of $|\langle\Psi(0)|\Psi^\star\rangle|^2$ is therefore identical to that of the quantity $\frac{w_{1,\mathrm{re}}^2}{\sum_{j=1}^d w_{j,\mathrm{re}}^2 + w_{j,\mathrm{im}}^2}$. By standard concentration of normal variables, the numerator is $\Theta(1)$ and the denominator is $\Theta(d)$, with any constant failure probability. Choosing both the failure probabilities to be 0.0025, we have that the condition $|\langle\Psi(0)|\Psi^\star\rangle|^2 \geq \Omega(1/d)$ is satisfied with probability at least 0.995.

Once the above condition is satisfied, Lemma 3.12 states that if the closeness condition on $\mathbf{Y}$ is maintained for time $T_\epsilon = \frac{1}{c} \frac{\log d}{\lambda_2 - \lambda_1} \log\left(\frac{1}{\epsilon}\right)$ the obtained error is less than or equal to ϵ.

Therefore, by Lemma 3.15 and Lemma 3.14, in order to ensure with failure probability at most 0.005, that $\|\mathbf{Y}(t) - \mathbf{Y}(0)\| \leq \frac{C_0}{d} \cdot \frac{\lambda_2 - \lambda_1}{\lambda_d - \lambda_1}$ for constant C_0 up to any time t such that $0 < t \leq T_\epsilon$ and $1 - |\langle \Psi | \Psi^\star \rangle|^2 \leq \exp(-c\frac{(\lambda_2 - \lambda_1)}{\log d} t')$ for all $t' \leq t$, it suffices to choose p such that

$$C_3 \left(\frac{T_\epsilon}{\sqrt{p}} \cdot \sqrt{2}(\lambda_d - \lambda_1) \cdot \sqrt{\frac{d^2 - 1}{Z(\mathbf{H}, d)^3}} \left(1 + \sqrt{\log\left(\frac{2d}{0.005} \right)} \right) \right) \leq \frac{\lambda_2 - \lambda_1}{\lambda_d - \lambda_1} \cdot \frac{C_0}{d}. \tag{3.117}$$

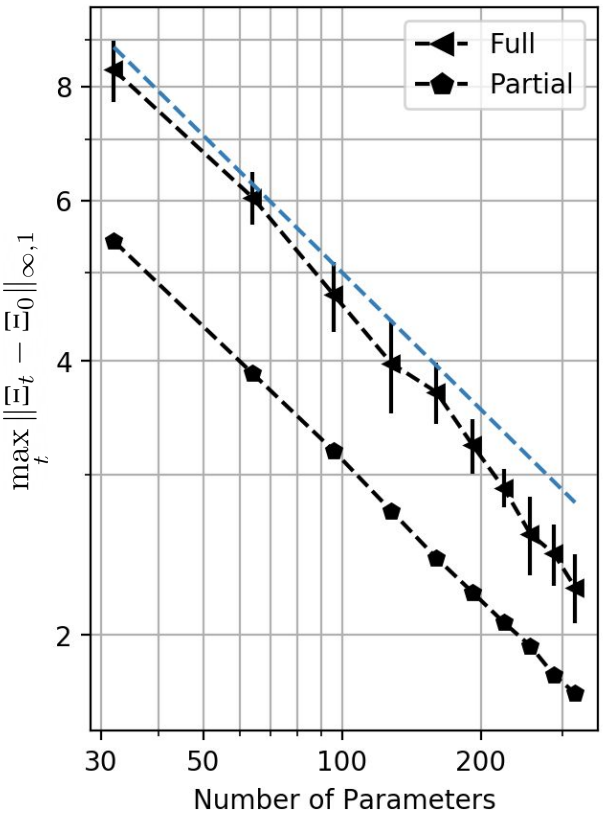
By simple algebra, it can be verified from the above that it suffices to choose any p greater than or equal to some over-parameterization threshold $p^\star$, where $p^\star = O\left(\left(\frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1} \right)^4 \frac{d^4}{Z(\mathbf{H}, d)^3} \log(d) \right)$.

The above argument shows that if the dynamics exhibits linear convergence upto some time t, the closeness condition on $\mathbf{Y}$ will also be satisified with failure probability at most δ, if the number of parameters is chosen appropriately. We now argue that *both* these conditions must hold until time T_ϵ. Let t_0 be the minimum time such that either $1 - |\langle \Psi_{t_0} | \Psi^\star \rangle|^2 > \exp(-c\frac{(\lambda_2 - \lambda_1)}{\log d} t_0)$ or $\|\mathbf{Y}(t_0) - \mathbf{Y}(0)\| > \frac{\lambda_2 - \lambda_1}{\lambda_d - \lambda_1} \cdot \frac{C_0}{d}$. If $1 - |\langle \Psi | \Psi^\star \rangle|^2 > \exp(-c\frac{(\lambda_2 - \lambda_1)}{\log d} t_0)$, we must have $\|\mathbf{Y}(t_0') - \mathbf{Y}(0)\| > \frac{\lambda_2 - \lambda_1}{\lambda_d - \lambda_1} \cdot \frac{C_0}{d}$ at some earlier time t_0' (Lemma 3.12). Similarly, if $\|\mathbf{Y}(t_0) - \mathbf{Y}(0)\| > \frac{\lambda_2 - \lambda_1}{\lambda_d - \lambda_1} \cdot \frac{C_0}{d}$, we must have $1 - |\langle \Psi(t_0') | \Psi^\star \rangle|^2 > \exp(-c\frac{(\lambda_2 - \lambda_1)}{\log d} t_0')$ at some earlier time t_0' (Lemma 3.15). Therefore by contradiction, both conditions must be realized for all times $t \leq T_\epsilon$, yielding the result. □

3.11.4 Convergence for Fully-trainable Ansatz

In this section we demonstrate that the over-parameterization threshold for partially-trainable ansatze extends to fully-trainable ansatze. First notice that, the concentration of Ξ_t at the random initialization holds following a similar proof for Lemma 3.14. To see that the concentration during training time also holds for the fully-trainable ansatze mirroring Lemma 3.15, we empirically

study the maximal deviation of $\|\Xi_t - \mathsf{id}\|_{\infty,1}$ by evaluating $\mathbf{Y}_t$ during training as a function of p (see Lemma 3.13 for the justification of the upper bound).



(a) $1/\sqrt{p}$-scaling of $\|\Xi_t - \Xi_0\|_{\infty,1}$

Figure 3.2: An upperbound on the maximal deviation of the channel Ξ_t during training from Ξ_0 at initialization for a 4-qubit hardware-efficient ansatz (HEA) with CZ entanglement. The standard deviations are calculated over 10 random initializations for each data point. The reference line in blue depends on the number of parameters p as $50/\sqrt{p}$. Both the fully and partially trainable ansatze has the channel distance scaling as $O(1/\sqrt{p})$.

In Figure 3.2, we plot $\max_{t>0} \|\mathbf{Y}_t - \mathbf{Y}_0\|_{\mathsf{op}}$ versus the number of parameters p as a proxy for $\max_{t>0} \|\Xi_t - \Xi_0\|_{\infty,1}$ over $10,000$ training steps: for both the partially- and fully-trainable ansatze, the maximal deviation scales with the inverse of square-root of p, suggesting that the fully-trainable ansatze have a similar $\mathsf{poly}(d, \kappa)$ *trainability* threshold.

3.12 Proof of Corollary 3.4

Lemma 3.20 (Output-state dynamics with noisy gradient estimation)**.** *Consider VQE instance* $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$*, with* $\mathbf{U}$ *being the ansatz defined in Definition 3.3. Under gradient flow with learning rate* η *and noisy gradient estimation* $\nabla L + \boldsymbol{\varepsilon}(t) = \left(\frac{\partial L}{\partial \theta_l} + \varepsilon_l(t)\right)_{l\in[p]}$*, the output state* $|\Psi(t)\rangle$ *follow the dynamics*

$$\frac{d}{dt}|\Psi(t)\rangle = -(\eta \cdot p \cdot Z(\mathbf{H}, d)) \operatorname{tr}_1(\mathbf{Y}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|] \otimes \mathbf{I}_{d\times d}))|\Psi(t)\rangle + \eta \sum_{l=1}^{p} i\varepsilon_l \mathbf{H}_l |\Psi(t)\rangle. \tag{3.17}$$

Here $\mathbf{H}_l$ *are function of* $\boldsymbol{\theta}(t)$*, defined as* $\mathbf{U}_{l:p}(\boldsymbol{\theta}(t))\mathbf{H}\mathbf{U}_{l:p}^\dagger(\boldsymbol{\theta}(t))$ *for all* $l \in [p]$*, and* $\mathbf{Y}$ *is defined as* $\frac{1}{pZ(\mathbf{H},d)} \sum_{l=1}^{p} \mathbf{H}_l^{\otimes 2}$.

Proof. We start by calculating the gradient of $\mathbf{U}_{r:p}(\boldsymbol{\theta})$ with respect to θ_l. For $r > l$, $\mathbf{U}_{r:p}(\boldsymbol{\theta})$ is independent of θ_l; for $r \le l$,

$$\frac{\partial \mathbf{U}_{r:p}}{\partial \theta_l} = \mathbf{U}_{l:p}(\boldsymbol{\theta})(-i\mathbf{H})\mathbf{U}_{r:l-1}(\boldsymbol{\theta}) = -i\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger \mathbf{U}_{r:p}. \tag{3.118}$$

Therefore

$$\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_l} = \langle\Phi|\mathbf{U}_0^\dagger \frac{\partial}{\partial \theta_l}\mathbf{U}_{1:p}^\dagger \mathbf{M}\mathbf{U}_{1:p}\mathbf{U}_0|\Phi\rangle + \langle\Phi|\mathbf{U}_0^\dagger\mathbf{U}_{1:p}^\dagger \mathbf{M}\frac{\partial}{\partial \theta_l}\mathbf{U}_{1:p}\mathbf{U}_0|\Phi\rangle \tag{3.119}$$

$$= \langle\Phi|\mathbf{U}_0^\dagger\mathbf{U}_{1:p}^\dagger i[\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger, \mathbf{M}]\mathbf{U}_{1:p}\mathbf{U}_0|\Phi\rangle \tag{3.120}$$

$$= i \operatorname{tr}([\mathbf{M}, |\Psi\rangle\langle\Psi|]\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger). \tag{3.121}$$

Following gradient flow with learning rate η:

$$\frac{d\theta_l}{dt} = -\eta\big(\frac{\partial}{\partial\theta_l}L(\boldsymbol{\theta}) + \varepsilon_l\big) = -i\eta\,\mathrm{tr}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger) - \eta\varepsilon_l. \tag{3.122}$$

The dynamics for $\mathbf{U}_{l:p}(\boldsymbol{\theta}(t))$ and $|\Psi(t)\rangle$ are therefore:

$$\frac{d}{dt}\mathbf{U}_{l:p}(t) \tag{3.123}$$

$$= \sum_{r=l}^{p} \frac{d\theta_r}{dt}\frac{\partial}{\partial\theta_r}\mathbf{U}_{l:p} \tag{3.124}$$

$$= -\eta\sum_{r=l}^{p}\mathrm{tr}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\mathbf{U}_{r:p}\mathbf{H}\mathbf{U}_{r:p}^\dagger)\mathbf{U}_{r:p}\mathbf{H}\mathbf{U}_{r:p}^\dagger\mathbf{U}_{l:p} + i\eta\sum_{r=1}^{p}\varepsilon_r\mathbf{U}_{r:p}\mathbf{H}\mathbf{U}_{r:p}^\dagger\mathbf{U}_{l:p}, \tag{3.125}$$

and

$$\frac{d}{dt}|\Psi(t)\rangle = \frac{d}{dt}\mathbf{U}_{1:p}\mathbf{U}_0|\Phi\rangle \tag{3.126}$$

$$= -(\eta\cdot pZ)\frac{1}{pZ}\big(\sum_{l=1}^{p}\mathrm{tr}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger)\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger\big)\mathbf{U}_{1:p}\mathbf{U}_0|\Phi\rangle \tag{3.127}$$

$$+ i\eta\sum_{l=1}^{p}\varepsilon_l\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger\mathbf{U}_{1:p}\mathbf{U}_0|\Phi\rangle \tag{3.128}$$

$$= -(\eta\cdot pZ)\,\mathrm{tr}_1(\mathbf{Y}[\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\otimes\mathbf{I})|\Psi(t)\rangle + \eta\sum_{l=1}^{p} i\varepsilon_l\mathbf{H}_l|\Psi(t)\rangle. \tag{3.129}$$

□

Lemma 3.21 (VQE perturbation lemma under noisy gradients)**.** *If*

- *the output state at initialization* $|\Psi(0)\rangle$ *has non-negligible overlap with the ground state* $|\Psi^\star\rangle$*:* $|\langle\Psi(0)|\Psi^\star\rangle|^2 \geq \Omega(\frac{1}{d})$,
- *for all* $0 \leq t \leq T$, $\|\mathbf{Y}(t) - \mathbf{Y}^\star(t)\|_{\mathsf{op}} \leq O\big(\frac{\lambda_2-\lambda_1}{\lambda_d-\lambda_1}\cdot\frac{1}{d}\big)$,

- *for all* $0 \le t \le T$, $\|\boldsymbol{\varepsilon}(t)\|_\infty \le c' \frac{Z}{\|\mathbf{H}\|_{\mathsf{op}}}(\lambda_2 - \lambda_1)\sqrt{1 - |\langle\Psi(t)|\Psi^\star\rangle|^2}|\langle\Psi(t)|\Psi^\star\rangle|$ *for some positive constant* c'*,*

then under the dynamics

$$\frac{d}{dt}|\Psi(t)\rangle = -\operatorname{tr}_1(\mathbf{Y}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|] \otimes \mathbf{I}_{d\times d}))|\Psi(t)\rangle + \frac{1}{pZ}\sum_{l=1}^{p} i\varepsilon_l \mathbf{H}_l|\Psi(t)\rangle, \tag{3.18}$$

the output states converges to the ground state such that for all $0 \le t \le T$*:*

$$1 - |\langle\Psi(t)|\Psi^\star\rangle|^2 \le \exp(-c\frac{\lambda_2 - \lambda_1}{\log d}t), \text{ for some constant } c. \tag{3.19}$$

Proof for Lemma 3.6. Let $\mathcal{E}(t) := \mathbf{Y}(t) - (\mathbf{W} - \frac{1}{d}\mathbf{I}_{d^2\times d^2})$ denote the deviation of $\mathbf{Y}(t)$ from its expected value. The matrix that governs the dynamics can be expressed as

$$\operatorname{tr}_1(\mathbf{Y}(t)([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|] \otimes \mathbf{I}_{d\times d})) = [\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|] + E(t) \tag{3.130}$$

where

$$E(t) := \operatorname{tr}_1\left(\mathcal{E}(t)([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|] \otimes \mathbf{I}_{d\times d})\right). \tag{3.131}$$

Define h as $|\langle\Psi^\star|\Psi(t)\rangle|^2$, the time derivative of h

$$\frac{d}{dt}h = (\frac{d}{dt}|\Psi(t)\rangle)^\dagger|\Psi^\star\rangle\langle\Psi^\star|\Psi(t)\rangle + \langle\Psi(t)|\Psi^\star\rangle \cdot \langle\Psi^\star|\frac{d}{dt}|\Psi(t)\rangle \tag{3.132}$$

$$= 2(\langle\Psi(t)|\mathbf{M}|\Psi(t)\rangle - \lambda_1)|\langle\Psi^\star|\Psi(t)\rangle|^2 \tag{3.133}$$

$$+ \mathrm{tr}(E(t)[|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]) \tag{3.134}$$

$$+ \mathrm{tr}(N(t)[|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]) \tag{3.135}$$

with $N(t)$ defined as $-\frac{1}{pZ}\sum i\varepsilon_{lt}\mathbf{H}_l$. The first term corresponds to the actual Riemannian gradient flow on the sphere:

$$2(\langle\Psi(t)|\mathbf{M}|\Psi(t)\rangle - \lambda_1)|\langle\Psi^\star|\Psi(t)\rangle|^2 = 2(\langle\Psi(t)|\mathbf{M}|\Psi(t)\rangle - \lambda_1)h \tag{3.136}$$

$$\geq 2((1-h)\lambda_2 + h\lambda_1 - \lambda_1)h \tag{3.137}$$

$$= 2(\lambda_2 - \lambda_1)(1-h)h. \tag{3.138}$$

The second term stems from the deviation of $\mathbf{Y}$ from its expectation:

$$\begin{aligned}
&\operatorname{tr}(E(t)[|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]) && (3.139)\\
=&\operatorname{tr}\left(\operatorname{tr}_1\left(\mathcal{E}(t)([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|] \otimes \mathbf{I}_{d\times d})\right)[|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]\right) && (3.140)\\
=&\operatorname{tr}\left(\mathcal{E}(t)([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|] \otimes [|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|])\right) && (3.141)\\
\geq& -\|\mathcal{E}(t)\|_{\mathsf{op}} \|[\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|] \otimes [|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]\|_{\mathsf{tr}} && (3.142)\\
=&-2\sqrt{h(1-h)}\,\|\mathcal{E}(t)\|_{\mathsf{op}} \|[\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\|_{\mathsf{tr}} && (3.143)\\
\geq& -2\sqrt{d}\sqrt{h(1-h)}\,\|\mathcal{E}(t)\|_{\mathsf{op}} \|[\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\|_F && (3.144)\\
\geq& -2\sqrt{2}\sqrt{d}\sqrt{h}(1-h)(\lambda_d-\lambda_1)\,\|\mathcal{E}(t)\|_{\mathsf{op}} && (3.145)\\
\geq& -C_4 h(1-h)(\lambda_2-\lambda_1)\frac{1}{\sqrt{hd}}. && (3.146)
\end{aligned}$$

Here we use technical Lemma 3.8 and 3.9 and the fact that $\|\mathcal{E}(t)\|_{\mathsf{op}}$ is $O(\frac{\lambda_2-\lambda_1}{\lambda_d-\lambda_1}\cdot\frac{1}{d})$.

The third term is a result of inaccurate estimation of gradients:

$$\operatorname{tr}(N(t)[|\Psi^\star\rangle\langle\Psi^\star|, |\Psi(t)\rangle\langle\Psi(t)|]) \geq -2\,\|N(t)\|_{\mathsf{op}}\sqrt{h(1-h)} \geq -C_5(\lambda_2-\lambda_1)h(1-h) \tag{3.147}$$

Here we use the fact that $\|N(t)\|_{\mathsf{op}} \leq O\big((\lambda_2-\lambda_1)\sqrt{h(1-h)}\big)$ if $\|\boldsymbol{\varepsilon}\|_\infty$ is $O\big(\frac{Z}{\|\mathbf{H}\|_{\mathsf{op}}}(\lambda_2-\lambda_1)\sqrt{h(1-h)}\big)$.

Combining all three terms, we have

$$\frac{d}{dt}h \geq C_6(\lambda_2-\lambda_1)(1-h)h(1-\frac{C_7}{\sqrt{hd}}). \tag{3.148}$$

Following the same calculation in the proof of Lemma 3.12, we have $1 - h(t) \leq -\ln h(t) \leq \exp(-c\frac{\lambda_2-\lambda_1}{\log d}t)$ for some constant c if $h(0)$ is $\Omega(1/d)$. □

3.13 Proof of Corollary 3.7

The proof of Corollary 3.7 involves replacing the integration formula in the proof to the main theorem with integration over subgroups. We start by presenting a basic fact about block-diagonal matrices (Lemma 3.22) and the integration formula for subgroups of $SU(d)$ (Lemma 3.23).

Lemma 3.22 (Basic fact)**.** *Let G be a matrix subgroup of $SU(d)$ inducing a decomposition of invariant subspace $V = \oplus_{j=1}^{m} V_j$ with projections $\{\mathbf{\Pi}_j\}_{j=1}^{m}$. Without loss of generality, assume V_1 to be the subspace of interest. Then for any Hermitian $\mathbf{A}$ and unitary matrix $\mathbf{U}$ in group G:*

$$\mathbf{\Pi}_1\mathbf{U}\mathbf{A}\mathbf{U}^\dagger\mathbf{\Pi}_1 = \mathbf{\Pi}_1\mathbf{U}\mathbf{\Pi}_1\ \mathbf{\Pi}_1\mathbf{A}\mathbf{\Pi}_1\ \mathbf{\Pi}_1\mathbf{U}^\dagger\mathbf{\Pi}_1 \tag{3.149}$$

Proof. The decomposition of invariant subspaces dictates that any $\mathbf{U} \in G$ is block-diagonal under $\{\mathbf{\Pi}_j\}_{j=1}^{m}$, namely $\forall \mathbf{U} \in G, \forall j \neq j', \mathbf{\Pi}_{j'}\mathbf{U}\mathbf{\Pi}_j = 0$.

$$\mathbf{\Pi}_1\mathbf{U}\mathbf{A}\mathbf{U}^\dagger\mathbf{\Pi}_1 \tag{3.150}$$

$$= \mathbf{\Pi}_1\mathbf{U}\sum_{j=1}^{m}\mathbf{\Pi}_j\mathbf{A}\sum_{j'=1}^{m}\mathbf{\Pi}_{j'}\mathbf{U}^\dagger\mathbf{\Pi}_1 \tag{3.151}$$

$$= \sum_{j,j'\in[m]}(\mathbf{\Pi}_1\mathbf{U}\mathbf{\Pi}_j)\mathbf{A}(\mathbf{\Pi}_{j'}\mathbf{U}^\dagger\mathbf{\Pi}_1) \tag{3.152}$$

$$= \mathbf{\Pi}_1\mathbf{U}\mathbf{\Pi}_1\mathbf{A}\mathbf{\Pi}_1\mathbf{U}^\dagger\mathbf{\Pi}_1 \tag{3.153}$$

$$= \mathbf{\Pi}_1\mathbf{U}\mathbf{\Pi}_1\mathbf{\Pi}_1\mathbf{A}\mathbf{\Pi}_1\mathbf{\Pi}_1\mathbf{U}^\dagger\mathbf{\Pi}_1. \tag{3.154}$$

The last equation uses the property of projections $\mathbf{\Pi}_j^2 = \mathbf{\Pi}_j$. $\square$

As a direct result, we have the following generic integral formula for $\mathbf{U}$ sampled from any $\mathcal{D}$ supported on the subgroup G:

Lemma 3.23 (Integration formula on subgroup restricted to an invariant subspace)**.** *Let G be a matrix subgroup of $SU(d)$ inducing a decomposition of invariant subspace $V = \oplus_{j=1}^m V_j$ with projections $\{\mathbf{\Pi}_j\}_{j=1}^m$. Without loss of generality, assume V_1 to be the subspace of interest and let $\mathbf{Q} \in \mathbb{C}^{d \times d_{\text{eff}}}$ be an arbitrary orthonormal basis for V_1. For any Hermitians $\{\mathbf{A}_r\}_{r=1}^R$ and measure $\mathcal{D}$ over G:*

$$(\mathbf{Q}^\dagger)^{\otimes R} \mathbb{E}_{\mathbf{U} \sim \mathcal{D}} [\otimes_{r=1}^R \mathbf{U} \mathbf{A}_r \mathbf{U}^\dagger] \mathbf{Q}^{\otimes R} = \mathbb{E}_{\mathbf{U}^{(1)} \sim \mathcal{D}^{(1)}} [\otimes_{r=1}^R \mathbf{U}^{(1)} \mathbf{A}_r^{(1)} (\mathbf{U}^{(1)})^\dagger] \tag{3.155}$$

where $\mathcal{D}^{(1)}$ is the distribution of $\mathbf{Q}^\dagger \mathbf{U} \mathbf{Q}$ for $\mathbf{U}$ sampled with respect to $\mathcal{D}$, and $\mathbf{A}_r^{(1)} := \mathbf{Q}^\dagger \mathbf{A}_r \mathbf{Q}$ is the Hermitian $\mathbf{A}_r$ restricted to the subspace V_1.

Lemma 3.23 allows using the integration formula in [59] when $\mathcal{D}^{(1)}$ is the Haar measure over a special unitary, special orthogonal or symplectic group. We are now ready to present the proof of Corollary 3.7.

Without loss of generality, we assume V_1 to be the relevant subspace with projection $\mathbf{\Pi}_1 = \mathbf{Q}\mathbf{Q}^\dagger$. For concise notations, define $\mathbf{U}^{(1)} = \mathbf{Q}^\dagger \mathbf{U} \mathbf{Q}$, $\mathbf{A}^{(1)} = \mathbf{Q}^\dagger \mathbf{A} \mathbf{Q}$ and $|\Psi^{(1)}\rangle = \mathbf{Q}^\dagger |\Psi\rangle$ for any unitary $\mathbf{U}$, Hermitian $\mathbf{A}$ and vector $|\Psi\rangle$.

Note that the potential function we track in the proof of Theorem 3.3 $|\langle \Psi^\star | \Psi(t) \rangle|^2$ is equal to $|\langle \Psi^{(1),\star} | \Psi^{(1)}(t) \rangle|^2$ if both $|\Psi\rangle^\star$ and $|\Psi(t)\rangle \in V_1$. Therefore for the purpose of the proof it suffices to track the dynamics of $|\Psi^{(1)}(t)\rangle$. Below we (1) first establish that $|\Psi(t)\rangle \in V_1$ through

out the training and (2) then show that the dynamics of $|\Psi^{(1)}(t)\rangle$ takes the same form as stated in Lemma 3.11 by replacing $\mathbf{M}$ and $\mathbf{H}$ with $\mathbf{M}^{(1)} = \mathbf{Q}^\dagger \mathbf{M} \mathbf{Q}$ and $\mathbf{H}^{(1)} = \mathbf{Q}^\dagger \mathbf{H} \mathbf{Q}$.

By Lemma 3.11, the dynamics of $|\Psi\rangle$ takes the form

$$\frac{d}{dt}|\Psi\rangle \propto -\frac{1}{p}\sum_{l=1}^{p} \mathrm{tr}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger)\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger|\Psi(t)\rangle. \tag{3.156}$$

We first show that $|\Psi(t)\rangle$ remains in V_1 for all t (i.e. $|\Psi(t)\rangle = \mathbf{\Pi}_1|\Psi(t)\rangle$). It suffices to show the time derivate $\frac{d|\Psi\rangle}{dt}$ stays in V_1 for $|\Psi\rangle \in V_1$ by noticing that for all $l \in [p]$,

$$\mathbf{\Pi}_1\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger|\Psi(t)\rangle \tag{3.157}$$

$$=\mathbf{U}_{l:p}\mathbf{\Pi}_1\mathbf{H}\mathbf{U}_{l:p}^\dagger|\Psi(t)\rangle \tag{3.158}$$

$$=\mathbf{U}_{l:p}\mathbf{H}\mathbf{\Pi}_1\mathbf{U}_{l:p}^\dagger|\Psi(t)\rangle \tag{3.159}$$

$$=\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger\mathbf{\Pi}_1|\Psi(t)\rangle \tag{3.160}$$

$$=\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger|\Psi(t)\rangle. \tag{3.161}$$

The first and the third equality is because $\mathbf{U}_{l:p} \in G_{\mathcal{A}}$ for all $l \in [p]$ and therefore block-diagoanl under $\{\mathbf{\Pi}_j\}_{j=1}^m$; The second equality is because and $\mathbf{H}$ is block-diagoanl under $\{\mathbf{\Pi}_j\}_{j=1}^m$; The last equality follows from $|\Psi\rangle \in V_j$.

We now calculate the dynamics of $|\Psi^{(1)}(t)\rangle$. For the trace operation in each term,

$$\mathrm{tr}([\mathbf{M}, |\Psi(t)\rangle\langle\Psi(t)|]\mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger) \tag{3.162}$$

$$= \mathrm{tr}([|\Psi(t)\rangle\langle\Psi(t)|, \mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger]\mathbf{M}) \tag{3.163}$$

$$= \mathrm{tr}([\mathbf{\Pi}_1|\Psi(t)\rangle\langle\Psi(t)|\mathbf{\Pi}_1, \mathbf{U}_{l:p}\mathbf{H}\mathbf{U}_{l:p}^\dagger]\mathbf{M}) \tag{3.164}$$

$$= \mathrm{tr}([\mathbf{\Pi}_1|\Psi(t)\rangle\langle\Psi(t)|\mathbf{\Pi}_1, \mathbf{\Pi}_1\mathbf{U}_{l:p}\mathbf{\Pi}_1\mathbf{H}\mathbf{\Pi}_1\mathbf{U}_{l:p}^\dagger\mathbf{\Pi}_1]\mathbf{M}) \tag{3.165}$$

$$= \mathrm{tr}([\mathbf{\Pi}_1|\Psi(t)\rangle\langle\Psi(t)|\mathbf{\Pi}_1, \mathbf{\Pi}_1\mathbf{U}_{l:p}\mathbf{\Pi}_1\mathbf{H}\mathbf{\Pi}_1\mathbf{U}_{l:p}^\dagger\mathbf{\Pi}_1]\mathbf{\Pi}_1\mathbf{M}\mathbf{\Pi}_1) \tag{3.166}$$

$$= \mathrm{tr}([\mathbf{Q}^\dagger|\Psi(t)\rangle\langle\Psi(t)|\mathbf{Q}, \mathbf{Q}^\dagger\mathbf{M}\mathbf{Q}]\mathbf{Q}^\dagger\mathbf{U}_{l:p}\mathbf{Q}\mathbf{Q}^\dagger\mathbf{H}\mathbf{Q}\mathbf{Q}^\dagger\mathbf{U}_{l:p}^\dagger\mathbf{Q}). \tag{3.167}$$

The first, fourth and the fifth equation follow from basic properties of trace operators; the second equality uses the fact that $|\Psi(t)\rangle$ stays in V_j; the third equality uses the fact that $\mathbf{U}_{l:p}$ and $\mathbf{H}$ are block-diagonal. Therefore we can rewrite Equation (3.156) as

$$\frac{d}{dt}|\Psi^{(1)}(t)\rangle \propto -\frac{1}{p}\sum_{l=1}^{p}\mathrm{tr}([\mathbf{M}^{(1)}, |\Psi^{(1)}(t)\rangle\langle\Psi^{(1)}(t)|]\mathbf{U}_{l:p}^{(1)}\mathbf{H}^{(1)}(\mathbf{U}_{l:p}^{(1)})^\dagger)\mathbf{U}_{l:p}^{(1)}\mathbf{H}^{(1)}(\mathbf{U}_{l:p}^{(1)})^\dagger|\Psi^{(1)}(t)\rangle. \tag{3.168}$$

The dynamics of $|\Psi^{(1)}(t)\rangle$ depends on $\mathbf{Q}^\dagger\mathbf{M}\mathbf{Q}$, $\mathbf{Q}^\dagger\mathbf{H}\mathbf{Q}$ and $\mathbf{Q}^\dagger\mathbf{U}\mathbf{Q}$. Corollary 3.7 follows trivially by using the integration formula specified in Lemma 3.23.

3.14 More on Experiments

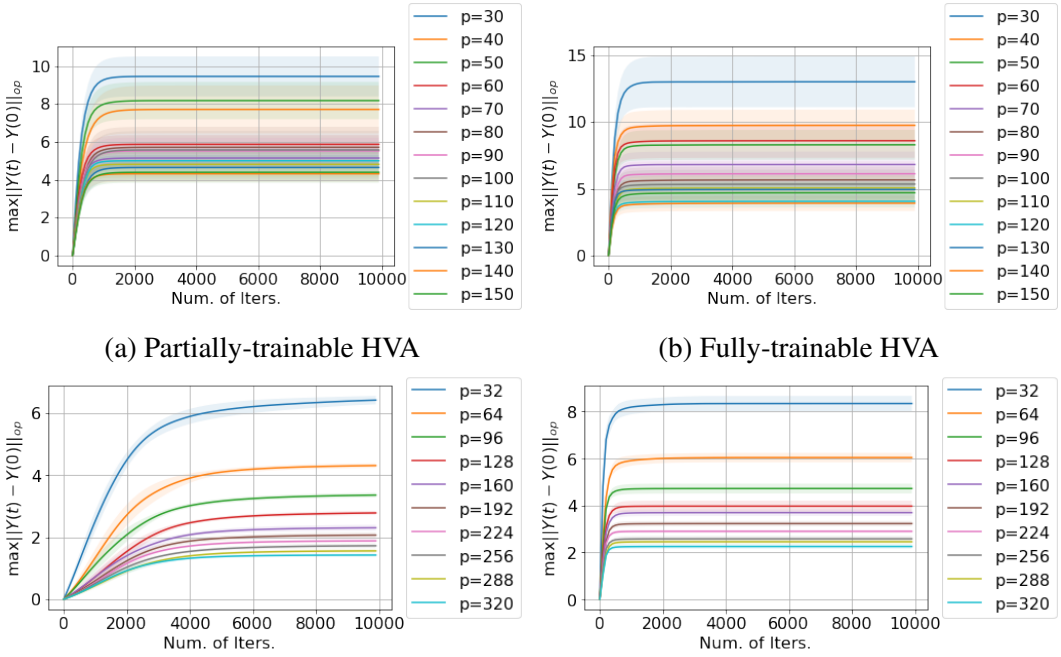
Implementation of Partially-Trainable Ansatze. We implement the partially-trainable ansatz

(Definition 3.3) by approximating the Haar measure over $G_{\mathcal{A}}$ by calculating

$$\mathbf{U}(\boldsymbol{\phi}) = \prod_{l'=1}^{L_{\text{sample}}} \prod_{k=1}^{K} \exp(-i\phi_{l',k}\mathbf{H}_k) \tag{3.169}$$

for $L_{\text{sample}} = 20$ and randomly initialized $\{\phi_{l',k}\}_{k\in[K],l'\in[L_{\text{sample}}]}$.

Deviation of $\mathbf{Y}$ and $\boldsymbol{\theta}$ as Functions of Time t. In Figure 3.3 and Figure 3.4, we plot the deviation of $\mathbf{Y}$ and $\boldsymbol{\theta}$ as functions of time steps t for both the partially- and fully-trainable settings. The mean values are plotted in solid lines and the shaded areas represent the standard deviation over random initializations. The maximum time steps is set to be $10,000$. As observed in Figure 3.3 and 3.4, the deviation of$\mathbf{Y}$ and $\boldsymbol{\theta}$ saturates quickly after a few time steps.



(a) Partially-trainable HVA

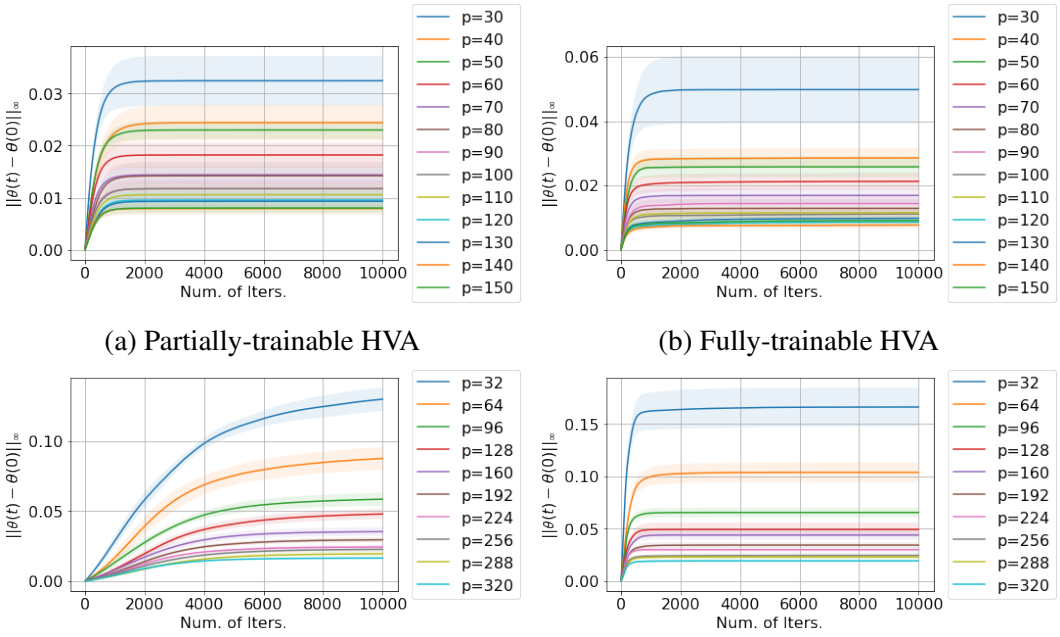
(b) Fully-trainable HVA

(c) Partially-trainable HEA

(d) Fully-trainable HEA

Figure 3.3: Deviation of $\mathbf{Y}$ during training for HVA and HEA

Definition of the Synthetic Problems. For the synthetic problem with system dimension d, effective dimension d_{eff} and the effective spectral ratio κ_{eff}, we embed a $d_{\text{eff}} \times d_{\text{eff}}$ problem



(a) Partially-trainable HVA

(b) Fully-trainable HVA

(c) Partially-trainable HEA

(d) Fully-trainable HVA

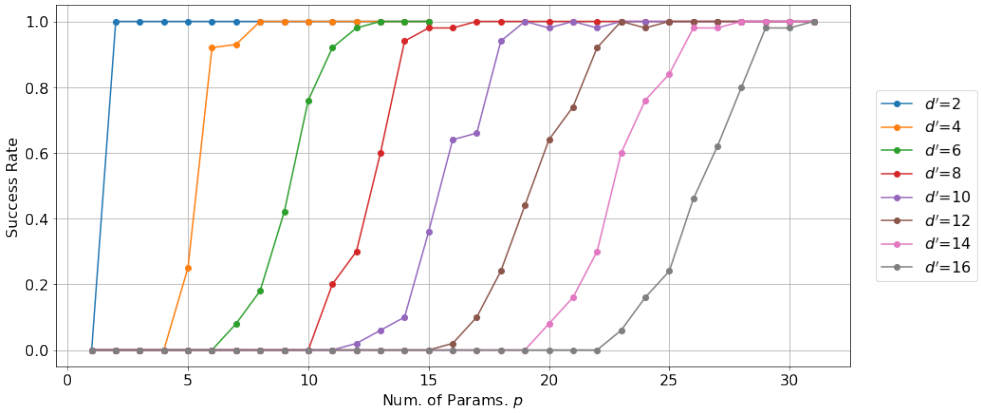
Figure 3.4: Deviation of $\boldsymbol{\theta}$ during training for HVA and HEA

Hamiltonian $\mathbf{M}^{(1)} = \mathbf{Q}^\dagger \mathbf{M} \mathbf{Q}$ with eigenvalues $(0, \frac{1}{\kappa_{\text{eff}}}, 1, \cdots, 1)$, generators $\mathbf{H}^{(1)} = \mathbf{Q}^\dagger \mathbf{H} \mathbf{Q}$ and unitaries $\{\mathbf{U}_l^{(1)} = \mathbf{Q}^\dagger \mathbf{U}_l \mathbf{Q}\}_{l=1}^p$ into a d-dimensional space using arbitrary $d \times d$ unitary $\mathbf{U}_{\text{embed}} = \begin{bmatrix} \mathbf{Q} & \mathbf{Q}^\perp \end{bmatrix}$ with $\mathbf{Q}^\perp$ being arbitrary complementary columns of $\mathbf{Q}$:

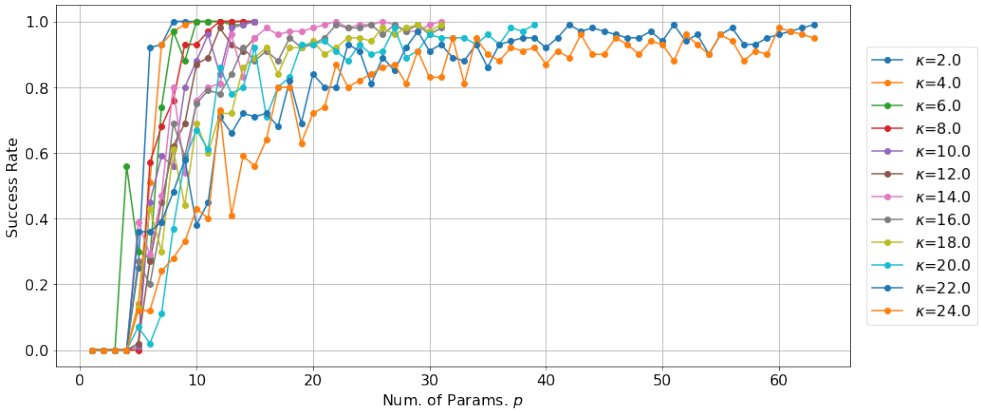
$$\mathbf{M} = \mathbf{U}_{\text{embed}} \begin{bmatrix} \mathbf{M}^{(1)} & 0 \\ 0 & \mathbf{I}_{d-d_{\text{eff}} \times d-d_{\text{eff}}} \end{bmatrix} \mathbf{U}_{\text{embed}}^\dagger \tag{3.170}$$

$$\mathbf{H} = \mathbf{U}_{\text{embed}} \begin{bmatrix} \mathbf{H}^{(1)} & 0 \\ 0 & 0 \end{bmatrix} \mathbf{U}_{\text{embed}}^\dagger \tag{3.171}$$

$$\mathbf{U}_l = \mathbf{U}_{\text{embed}} \begin{bmatrix} \mathbf{U}_l^{(1)} & 0 \\ 0 & \mathbf{I}_{d-d_{\text{eff}} \times d-d_{\text{eff}}} \end{bmatrix} \mathbf{U}_{\text{embed}}^\dagger, \quad \forall l \in [p] \tag{3.172}$$



(a) Varying effective dimension d_{eff}



(b) Varying effective ratio κ_{eff}

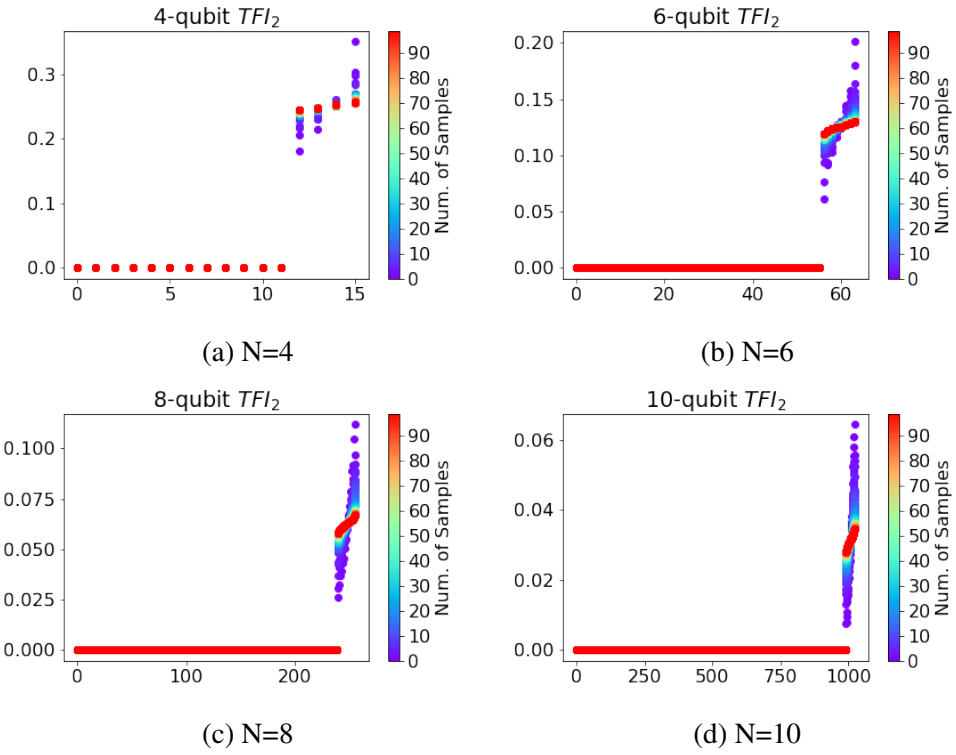
Figure 3.5: The success rate for achieving a 0.01-approximation for the ground state as a function of number of parameters. Each curve corresponds to a synthetic instance with dimension 16 and with varying $(d_{\text{eff}}, \kappa_{\text{eff}})$. Success rates are estimated over 100 random initializations. Top: Fixing $d = 16, \kappa_{\text{eff}} = 4.0$ for $d_{\text{eff}} = 2, 4, 6, \cdots, 16$. The threshold increases as the system dimension increases. Bottom: Fixing $d = 16, d_{\text{eff}} = 4$ for $\kappa_{\text{eff}} = 2.0, 4.0, 6.0, \cdots, 24.0$. The threshold is positively correlated to the spectral ratio of the system.

And the ansatz takes the form

$$\mathbf{U}(\boldsymbol{\theta}) = \big(\prod_{l=1}^{p} \mathbf{U}_l \exp(-i\theta_l \mathbf{H})\big)\mathbf{U}_0 \tag{3.173}$$

where $d_{\text{eff}} \times d_{\text{eff}}$ unitaries $\{\mathbf{U}_l^{(1)}\}$ are sampled i.i.d from the Haar measure over $SU(d_{\text{eff}})$. In Figure 3.5, we plot the success rates versus the number of parameters for various d_{eff} and κ_{eff} that are similar to Figure 3.1.

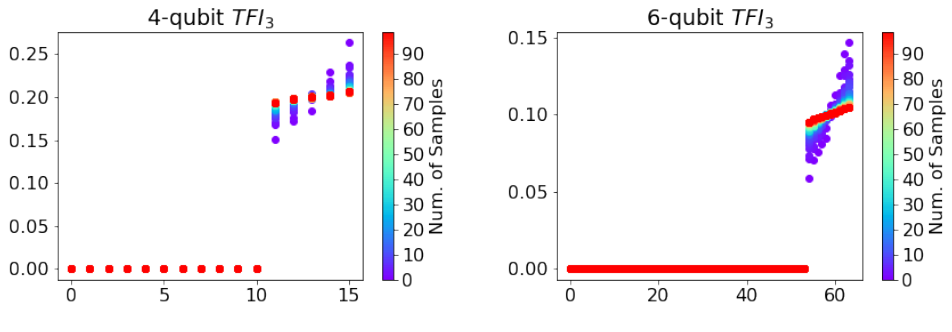
Estimating Invariant Subspaces for TFI and XXZ Models. Similar to the Kitaev model in Section 5.1.1, we numerically confirm that the TFI and XXZ models involved are all compatible. The convergences of the empirical estimatino of projection $\hat{\mathbf{\Pi}}$ are summarized in Figure 3.6, Figure 3.7, Figure 3.8 and Figure 3.9. For each of the plots, the x-axes corresponds to the indexes of the eigenvalues sorted in the ascending orders. The value of R in Equation 5.1 ranges from 0 to 100 and is color-coded, increasing from blue to red.



(a) N=4 (b) N=6

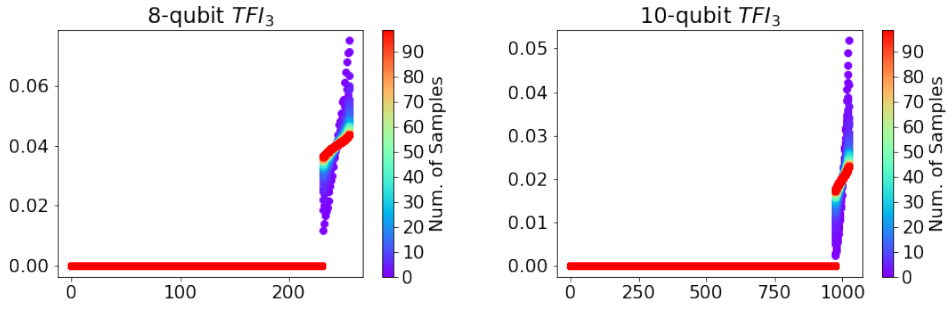
(c) N=8 (d) N=10

Figure 3.6: Specturm of $\hat{\Pi}$ for $TFI_{2\text{alt}}$ model with $4, 6, 8, 10$ qubits



(a) N=4

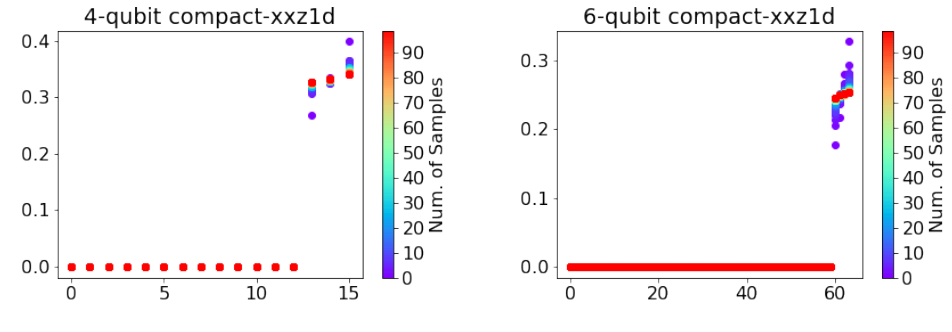
(b) N=6



(c) N=8

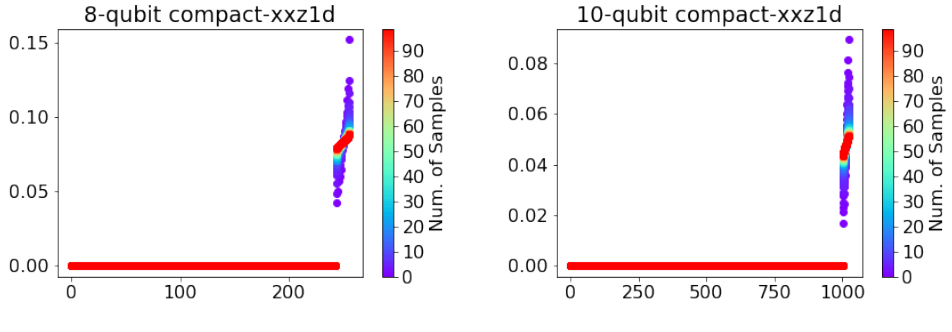
(d) N=10

Figure 3.7: Specturm of $\hat{\Pi}$ for $TFI_{3\text{alt}}$ model with $4, 6, 8, 10$ qubits



(a) N=4

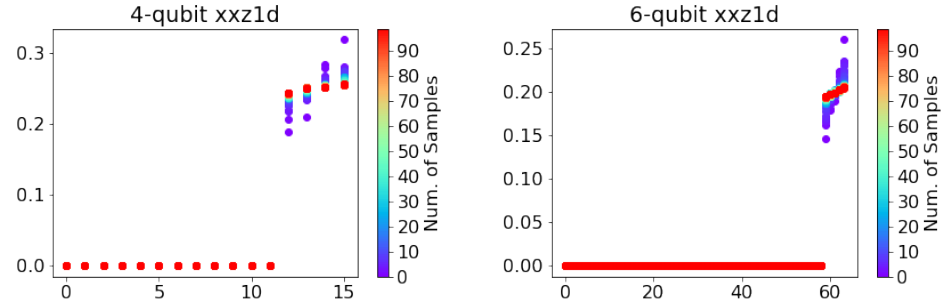
(b) N=6



(c) N=8

(d) N=10

Figure 3.8: Specturm of $\hat{\Pi}$ for $XXZ_{4\text{alt}}$ model with $4, 6, 8, 10$ qubits



(a) N=4 (b) N=6

(c) N=8 (d) N=10

Figure 3.9: Specturm of $\hat{\Pi}$ for XXZ_{6alt} model with $4, 6, 8, 10$ qubits

Chapter 4: Analyzing Convergence in Quantum Neural Networks: Deviations from Neural Tangent Kernels

In Chapter 2, we saw that typical under-parameterized QNNs with $p = O(\log d)$ parameters can have many local minima, In this chapter, we apply the tools developed in the last chapter to examine the convergence of quantum neural networks. We attempt to answer, what is the behaviour of QNNs in the limit of over-parameterization.

Despite the existing empirical and theoretical investigations, the convergence of QNN training is not fully understood. Inspired by the success of the neural tangent kernels (NTKs) in probing into the dynamics of classical neural networks, a recent line of works proposes to study over-parameterized QNNs by examining a quantum version of tangent kernels. In this chapter, we study the dynamics of QNNs and show that contrary to popular belief it is qualitatively different from that of any kernel regression: due to the unitarity of quantum operations, there is a non-negligible deviation from the tangent kernel regression derived at the random initialization. As a result of the deviation, we prove the at-most sublinear convergence for QNNs with Pauli measurements, which is beyond the explanatory power of any kernel regression dynamics. We then present the actual dynamics of QNNs in the limit of over-parameterization. The new dynamics capture the change of convergence rate during training, and implies that the range of measurements is crucial to the fast QNN convergence.

As we will see in Section 4.11, the analysis is applicable to all variational algorithms, providing a general lens for studying the over-parameterization in VQAs.

4.1 Introduction

Analogous to the classical logic gates, quantum gates are the basic building blocks for quantum computing. A variational quantum circuit (also referred to as an ansatz) is composed of parameterized quantum gates. Recall that a quantum neural network (QNN) is nothing but an instantiation of learning with parametric models using variational quantum circuits and quantum measurements: A p-parameter d-dimensional QNN for a dataset $\{\mathbf{x}_i, y_i\}$ is specified by an encoding $\mathbf{x}_i \mapsto \boldsymbol{\rho}_i$ of the feature vectors into quantum states in an underlying d-dimensional Hilbert space $\mathcal{H}$, a variational circuit $\mathbf{U}(\boldsymbol{\theta})$ with real parameters $\boldsymbol{\theta} \in \mathbb{R}^p$, and a quantum measurement $\mathbf{M}_0$. The predicted output $\hat{y}_i$ is obtained by measuring $\mathbf{M}_0$ on the output $\mathbf{U}(\boldsymbol{\theta})\boldsymbol{\rho}_i\mathbf{U}^\dagger(\boldsymbol{\theta})$. (unlike in Chapter 2, we use $\mathbf{M}_0$ instead of $\mathbf{M}$ to highlight the parameter-dependency of the measurements in the Heisenberg's picture). Like deep neural networks, the parameters $\boldsymbol{\theta}$ in the variational circuits are optimized by gradient-based methods to minimize an objective function that measures the misalignments of the predicted outputs and the ground truth labels.

Despite their potential there are challenges in the practical deployment of QNNs. Most notably, the optimization problem for training QNNs can be highly non-convex. The landscape of QNN training may be swarmed with spurious local minima and saddle points that can trap gradient-based optimization methods (see Chapter 2 and [64, 73]). QNNs with large dimensions also suffer from a phenomenon called the *barren plateau* [40], where the gradients of the parameters vanish at random intializations, making convergence slow even in a trap-free landscape. These

difficulties in training QNNs, together with the challenge of classically simulating QNNs at a decent scale, calls for a theoretical understanding of the convergence of QNNs.

Neural Tangent Kernels. Many of the theoretical difficulties in understanding QNNs have also been encountered in the study of classical deep neural networks: despite the landscape of neural networks being non-convex and susceptible to spurious local minima and saddle points, it has been empirically observed that the training errors decays exponentially in the training time [74, 75] in the highly *over-parameterized* regime with sufficiently many number of trainable parameters. This phenomenon is theoretically explained by connecting the training dynamics of neural networks to the kernel regression: the kernel regression model generalizes the linear regression by equipping the linear model with non-linear feature maps. Given a training set $\{\mathbf{x}_j, y_j\}_{j=1}^m \subset \mathcal{X} \times \mathcal{Y}$ and a non-linear feature map $\phi : \mathcal{X} \to \mathcal{X}'$ mapping the features to a potentially high-dimensional feature space $\mathcal{X}'$. The kernel regression solves for the optimal weight $\mathbf{w}$ that minimizes the mean-square loss $\frac{1}{2m}\sum_{j=1}^m(\mathbf{w}^T\phi(\mathbf{x}_j) - y_j)^2$. The name of kernel regression stems from the fact that the optimal hypothesis $\mathbf{w}$ depends on the high-dimensional feature vectors $\{\phi(\mathbf{x}_j)\}_{j=1}^m$ through a $m \times m$ *kernel* matrix $\mathbf{K}$, such that $K_{ij} = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$. The kernel regression enjoys a linear convergence (i.e. the mean square loss decaying exponentially over time) when $\mathbf{K}$ is positive definite.

The kernel matrix associated with a neural network is determined by tracking how the predictions for each training sample evolve jointly at random initialization. The study of the neural network convergence then reduces to characterizing the corresponding kernel matrices (the neural tangent kernel, or the NTK). In addition to the convergence results, NTK also serves as a tool for studying other aspect of neural networks including generalization [76, 77] and stability [78].

The key observation that justifies the study of neural networks with neural tangent kernels, is that the NTK becomes a constant (over time) during training in the limit of infinite layer widths. This has been theoretically established starting with the analysis of wide fully-connected neural networks [31, 32, 66] and later generalized to a variety of architectures (e.g. [79]).

Quantum NTKs. Inspired by the success of NTKs, recent years have witnessed multiple works attempting to associate over-parameterized QNNs to kernel regression. Along the line there are two types of studies. The first category investigates and compares the properties of the "quantum" kernel induced by the quantum encoding of classical features, where K_{ij} associated with the i-th and j-th feature vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ equals $\mathrm{tr}(\boldsymbol{\rho}_\mathbf{i}\boldsymbol{\rho}_\mathbf{j})$ with $\boldsymbol{\rho}_i$ and $\boldsymbol{\rho}_j$ being the quantum state encodings, without referring to the dynamics of training [80, 81, 82]. The second category seeks to directly establish the quantum version of NTK for QNNs by examining the evolution of the model predictions at random initialization, which is the recipe for calculating the classical NTK in [32]: [83] empirically evaluates the direct training of the quantum NTK instead of the original QNN formulation. On the other hand, by analyzing the time derivative of the quantum NTK at initialization, [84] conjectures that in the limit of over-parameterization, the quantum NTK is a constant over time and therefore the dynamics reduces to a kernel regression.

Despite recent efforts, a rigorous answer remains evasive whether the quantum NTK is a constant during training for over-parameterized QNNs. We show that the answer to this question is indeed, surprisingly negative: as a result of the unitarity of quantum circuits, there is a finite change in the conjectured quantum NTK as the training error decreases, even in the the limit of over-parameterization.

Contributions. In this work, we focus on QNNs equipped with the mean square loss, trained using gradient flow, following [32]. In Section 4.3, we show that, despite the formal resemblance

to kernel regression dynamics, the over-parameterized QNN does not follow the dynamics of *any* kernel regression due to the unitarity: for the widely-considered setting of classifications with Pauli measurements, we show that the objective function at time t decays at most as a polynomial function of $1/t$ (Theorem 4.2). This contradicts the dynamics of any kernel regression with a positive definite kernel, which exhibits convergence with $L(t) \leq L(0)\exp(-ct)$ for some positive constant c. We also identify the true asymptotic dynamics of QNN training as regression with a time-varying Gram matrix $\mathbf{K}_{\mathsf{asym}}$ (Lemma 4.3), and show rigorously that the real dynamics concentrates to the asymptotic one in the limit $p \to \infty$ (Theorem 4.4). This reduces the problem of investigating QNN convergence to studying the convergence of the asymptotic dynamics governed by $\mathbf{K}_{\mathsf{asym}}$.

We also consider a model of QNNs where the final measurement is post-processed by a linear scaling. In this setting, we provide a complete analysis of the convergence of the asymptotic dynamics in the case of 1 training sample (Corollary 4.5), and provide further theoretical evidence of convergence in the neighborhood of most global minima when the number of samples $m > 1$ (Theorem 4.6). These theoretical evidences are supplemented with an empirical study that demonstrates in generality, the convergence of the asymptotic dynamics when $m \geq 1$. Coupled with our proof of convergence, these form the strongest concrete evidences of the convergence of training for over-parameterized QNNs.

Connections to previous works. Our result extends the existing literature on QNN landscapes (e.g. [43, 57]) and looks into the training dynamics, which allows us to characterize the rate of convergence and to show how the range of the measurements affects the convergence to global minima. The dynamics for over-parameterized QNNs proposed by us can be reconciled with the existing calculations of quantum NTK as follows: in the regime of over-parameterization,

the QNN dynamics coincides with the quantum NTK dynamics conjectured in [84] at random initialization; yet it deviates from quantum NTK dynamics during training, and the deviation does not vanish in the limit of $p \to \infty$.

4.2 Preliminaries

Classical neural networks A popular choice of the hypothesis set $\mathcal{F}$ in modern-day machine learning is the *classical neural networks*. A vanilla version of the L-layer feed-forward neural network takes the form $f(x; \mathbf{W}_1, \cdots, \mathbf{W}_L) = \mathbf{W}_L \sigma(\cdots \mathbf{W}_2 \sigma(\mathbf{W}_1 \sigma(x)) \cdots)$, where $\sigma(\cdot)$ is a non-linear activation function, and for all $l \in [L]$, $\mathbf{W}_l \in \mathbb{R}^{d_l \times d_{l-1}}$ is the weights in the l-th layer, with $d_L = 1$ and d_0 the same as the dimension of the feature space $\mathcal{X}$. It has been shown that, in the limit $\min_{l=1}^{L-1} d_l \to \infty$, the training of neural networks with square loss is close to kernel learning, and therefore enjoys a linear convergence rate [31, 32, 79, 85].

Quantum neural networks. Quantum neural networks is a family of parameterized hypothesis set analogous to its classical counterpart. At a high level, it has the layered-structure like a classical neural network. At each layer, a linear transformation acts on the output from the last layer. A quantum neural network is different from its classical counterpart in the following three aspects. Similar to Chapter 2, we consider the general p-parameter ansatz $\mathbf{U}(\boldsymbol{\theta})$ in a d-dimensional Hilbert space can be specified by a set of $d \times d$ unitaries $\{\mathbf{U}_0, \mathbf{U}_1, \cdots, \mathbf{U}_p\}$ and a set of non-zero $d \times d$ Hermitians $\{\mathbf{H}^{(1)}, \mathbf{H}^{(2)}, \cdots, \mathbf{H}^{(p)}\}$ as

$$\mathbf{U}_p \exp(-i\theta_p \mathbf{H}^{(p)}) \mathbf{U}_{p-1} \exp(-i\theta_p \mathbf{H}^{(p-1)}) \cdots \exp(-i\theta_2 \mathbf{H}^{(2)}) \mathbf{U}_1 \exp(-i\theta_1 \mathbf{H}^{(1)}) \mathbf{U}_0. \tag{4.1}$$

Without loss of generality, we assume that $\mathrm{tr}(\mathbf{H}^{(l)}) = 0$. Notice that most p-parameter ansatze

$\mathbf{U} : \mathbb{R}^p \to \mathbb{C}^{d\times d}$ can be expressed as Equation 4.1. One exception may be the anastz design with intermediate measurements (e.g. [86]). In Section 4.4, we will also consider the periodic anastz:

Definition 4.1 (Periodic ansatz)**.** A d-dimensional p-parameter periodic anasatz $\mathbf{U}(\boldsymbol{\theta})$ is defined as

$$\mathbf{U}_p \exp(-i\theta_p \mathbf{H}) \cdot \cdots \cdot \mathbf{U}_1 \exp(-i\theta_1 \mathbf{H})\mathbf{U}_0, \tag{4.2}$$

where $\mathbf{U}_l$ are sampled $i.i.d.$ with respect to the Haar measure over the special unitary group $SU(d)$, and $\mathbf{H}$ is a non-zero trace-0 Hermitian.

Up to a unitary transformation, the periodic ansatz is equivalent to an ansatz in Line 4.1 where $\{\mathbf{H}^{(l)}\}_{l=1}^p$ sampled as $\mathbf{V}_l\mathbf{H}\mathbf{V}_l^\dagger$ with $\mathbf{V}_l$ being haar random $d\times d$ unitary matrices. Moreover, training the periodic ansatz is equivalent to training a subset of parameters for ansatze generated by periodically stacking the unitaries in Line 4.1 (see e.g. [62, 63]) at random initialization (See also the discussion in Chapter 3). Similar ansatze have been considered in [40, 57, 64, 87].

ERM of quantum neural network. Like in Chapter 2, we consider the empirical risk minimization with the common choice of the *square loss* $\ell(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$. Solving the ERM for a dataset $\mathcal{S} := \{(\boldsymbol{\rho}_j, y_j)\}_{j=1}^m \subseteq (\mathbb{C}^{d\times d} \times \mathbb{R})^m$ involves optimizing the objective function $\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) := \frac{1}{2m}\sum_{j=1}^m \left(\hat{y}_j(\boldsymbol{\theta}) - y_j\right)^2$, where $\hat{y}_j(\boldsymbol{\theta}) = \mathrm{tr}(\boldsymbol{\rho}_j \mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}_0\mathbf{U}(\boldsymbol{\theta}))$ for all $j \in [m]$ with $\mathbf{M}_0$ being the quantum measurement and $\mathbf{U}(\boldsymbol{\theta})$ being the variational ansatz. Typically, a QNN is trained by optimizing the ERM objective function by gradient descent: at the t-th iteration, the parameters are updated as $\boldsymbol{\theta}(t+1) \leftarrow \boldsymbol{\theta}(t) - \eta\nabla L(\boldsymbol{\theta}(t))$, where η is the learning rate; for sufficiently small η, the dynamics of gradient descent reduces to that of the gradient flow: $d\boldsymbol{\theta}(t)/dt = -\eta\nabla L(\boldsymbol{\theta}(t))$. Here we focus on the gradient flow setting following [32].

Rate of convergence In the optimization literature, the rate of convergence describes how fast an iterative algorithm approaches an (approximate) solution. For a general function L with variables $\boldsymbol{\theta}$, let $\boldsymbol{\theta}(t)$ be the solution maintained at the time step t and $\boldsymbol{\theta}^\star$ be the optimal solution. The algorithm is said to be converging *exponentially fast* or at a *linear rate* if $L(\boldsymbol{\theta}(t)) - L(\boldsymbol{\theta}^\star) \leq \alpha \exp(-ct)$ for some constants c and α. In contrast, algorithms with the sub-optimal gap $L(\boldsymbol{\theta}(t)) - L(\boldsymbol{\theta}^\star)$ decreasing slower than exponential are said to be converging with a *sublinear* rate (e.g. $L(\boldsymbol{\theta}(t)) - L(\boldsymbol{\theta}^\star)$ decaying with t as a polynomial of $1/t$). We will mainly consider the setting where $L(\boldsymbol{\theta}^\star) = 0$ (i.e. the *realizable* case) with continuous time t.

4.3 Deviations of QNN Dynamics from NTK

Consider a regression model on an m-sample training set: for all $j \in [m]$, let y_j and $\hat{y}_j$ be the label and the model prediction of the j-th sample. The *residual* vector $\mathbf{r}$ is a m-dimensional vector with $r_j := y_j - \hat{y}_j$. The dynamics of the kernel regression is signatured by the first-order linear dynamics of the residual vectors: let $\mathbf{w}$ be the learned model parameter, and let $\phi(\cdot)$ be the fixed non-linear map. Recall that the kernel regression minimizes $L(\mathbf{w}) = \frac{1}{2m}\sum_{j=1}^m (\mathbf{w}^T\phi(\mathbf{x}_j) - y_j)^2$ for a training set $\mathcal{S} = \{(\mathbf{x}_j, y_j)\}_{j=1}^m$, and the gradient with respect to $\mathbf{w}$ is $\frac{1}{m}\sum_{j=1}^m (\mathbf{w}^T\phi(\mathbf{x}_j) - y_j)\phi(\mathbf{x}_j) = -\frac{1}{m}\sum_{j=1}^m r_j\phi(\mathbf{x}_j)$. Under the gradient flow with learning rate η, the weight $\mathbf{w}$ updates as $\frac{d\mathbf{w}}{dt} = \frac{\eta}{m}\sum_{j=1}^m r_j\phi(\mathbf{x}_j)$, and the i-th entry of the residual vector updates as $dr_i/dt = -\phi(\mathbf{x}_i)^T\frac{d\mathbf{w}}{dt} = -\frac{\eta}{m}\sum_{j=1}^m \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)r_j$, or more succinctly $d\mathbf{r}/dt = -\frac{\eta}{m}\mathbf{K}\mathbf{r}$ with $\mathbf{K}$ being the kernel/Gram matrix defined as $K_{ij} = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$ (see also [32]). Notice that the kernel matrix $\mathbf{K}$ is a constant of time and is independent of the weight $\mathbf{w}$ or the labels.

Dynamics of residual vectors. We start by characterizing the dynamics of the residual vectors for the general form of p-parameter QNNs and highlight the limitation of viewing the over-parameterized QNNs as kernel regressions. Similar to the kernel regression, $\frac{dr_j}{dt} = -\frac{d\hat{y}_j}{dt} = -\operatorname{tr}(\boldsymbol{\rho}_j \frac{d}{dt}\mathbf{U}^\dagger(\boldsymbol{\theta}(t))\mathbf{M}_0\mathbf{U}(\boldsymbol{\theta}(t)))$ in QNNs. We derive the following dynamics of $\mathbf{r}$ by tracking the parameterized measurement $\mathbf{M}(\boldsymbol{\theta}) = \mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}_0\mathbf{U}(\boldsymbol{\theta})$ as a function of time t.

Lemma 4.1 (Dynamics of the residual vector)**.** *Consider a QNN instance with an ansatz* $\mathbf{U}(\boldsymbol{\theta})$ *defined as in Line 4.1, a training dataset* $\mathcal{S} = \{(\boldsymbol{\rho}_j, y_j)\}_{j=1}^m$*, and a measurement* $\mathbf{M}_0$*. Under the gradient flow for the objective function* $L(\boldsymbol{\theta}) = \frac{1}{2m}\sum_{j=1}^m \left(\operatorname{tr}(\boldsymbol{\rho}_j\mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}_0\mathbf{U}(\boldsymbol{\theta})) - y_j\right)^2$ *with learning rate* η*, the residual vector* $\mathbf{r}$ *satisfies the differential equation*

$$\frac{d\mathbf{r}(\boldsymbol{\theta}(t))}{dt} = -\frac{\eta}{m}\mathbf{K}(\mathbf{M}(\boldsymbol{\theta}(t)))\mathbf{r}(\boldsymbol{\theta}(t)), \tag{4.3}$$

where $\mathbf{K}$ *is a positive semi-definite matrix-valued function of the parameterized measurement. The* (i,j)*-th element of* $\mathbf{K}$ *is defined as*

$$\sum_{l=1}^{p}\left(\operatorname{tr}\left(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_i]\mathbf{H}_l\right)\operatorname{tr}\left(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_j]\mathbf{H}_l\right)\right). \tag{4.4}$$

Here $\mathbf{H}_l := \mathbf{U}_0^\dagger\mathbf{U}_{1:l-1}^\dagger(\boldsymbol{\theta})\mathbf{H}^{(l)}\mathbf{U}_{1:l-1}(\boldsymbol{\theta})\mathbf{U}_0$*, is a function of* $\boldsymbol{\theta}$ *with* $\mathbf{U}_{1:r}(\boldsymbol{\theta})$ *being the shorthand for* $\mathbf{U}_r\exp(-i\theta_r\mathbf{H}^{(r)})\cdots\mathbf{U}_1\exp(-i\theta_1\mathbf{H}^{(1)})$.

While Equation 4.3 takes a similar form to that of the kernel regression, the matrix $\mathbf{K}$ is *dependent* on the parameterized measurement $\mathbf{M}(\boldsymbol{\theta})$. This is a consequence of the unitarity: consider an alternative parameterization, where the objective function $\mathbf{L}(\mathbf{M}) = \frac{1}{2m}\sum_{j=1}^m \left(\operatorname{tr}(\boldsymbol{\rho}_j\mathbf{M}) - y_j\right)^2$ is optimized over all Hermitian matrices $\mathbf{M}$. It can be easily verified that the corresponding

dynamics is exactly the kernel regression with $K_{ij} = \mathrm{tr}(\boldsymbol{\rho}_i \boldsymbol{\rho}_j)$.

Due to the unitarity of the evolution of quantum states, the spectrum of eigenvalues of the parameterized measurement $\mathbf{M}(\boldsymbol{\theta})$ is required to remain the same throughout training. In the proof of Lemma 4.1 (deferred to Section 4.6.1), we see that the derivative of $\mathbf{M}(\boldsymbol{\theta})$ takes the form of a linear combination of commutators $i[\mathbf{A}, \mathbf{M}(\boldsymbol{\theta})]$ for some Hermitian $\mathbf{A}$. As a result, the traces of the k-th matrix powers $\mathrm{tr}(\mathbf{M}^k(\boldsymbol{\theta}))$ are constants of time for any integer k, since $d\,\mathrm{tr}(\mathbf{M}^k(\boldsymbol{\theta}))/dt = k\,\mathrm{tr}(\mathbf{M}^{k-1}(\boldsymbol{\theta}) d\mathbf{M}(\boldsymbol{\theta})/dt) = k\,\mathrm{tr}(\mathbf{M}^{k-1}(\boldsymbol{\theta}) i[\mathbf{A}, \mathbf{M}(\boldsymbol{\theta})]) = 0$ for any Hermitian $\mathbf{A}$. The spectrum of eigenvalues remains unchanged because the coefficients of the characteristic polynomials of $\mathbf{M}(\boldsymbol{\theta})$ is completely determined by the traces of matrix powers. On the contrary, the eigenvalues are in general not preserved for $\mathbf{M}$ evolving under the kernel regression.

Another consequence of the unitarity constraint is that a QNN can not make predictions outside the range of the eigenvalues of $\mathbf{M}_0$, while for the kernel regression with a strictly positive definite kernel, the model can (over-)fit training sets with arbitrary label assignments. Here we further show that the unitarity is pronounced in a typical QNN instance where the predictions are within the range of the measurement.

Sublinear convergence in QNNs. One of the most common choices for designing QNNs is to use a (tensor product of) Pauli matrices as the measurement (see e.g. [9, 88]). Such a choice features a measurement $\mathbf{M}_0$ with eigenvalues $\{\pm 1\}$ and trace zero. Here we show that in the setting of supervised learning on pure states with Pauli measurements, the (neural tangent) kernel regression is insufficient to capture the convergence of QNN training. For the kernel regression with a positive definite kernel $\mathbf{K}$, the objective function L can be expressed as $\frac{1}{2m}\sum_{j=1}^{m}(\hat{y}_j - y_j)^2 = \frac{1}{2m}\mathbf{r}^T\mathbf{r}$; under the kernel dynamics of $\frac{d\mathbf{r}}{dt} = -\frac{\eta}{m}\mathbf{K}\mathbf{r}$, it is easy to verify that $\frac{d\ln L}{dt} =$

$-\frac{2\eta}{m}\frac{\mathbf{r}^T\mathbf{K}\mathbf{r}}{\mathbf{r}^T\mathbf{r}} \leq -\frac{2\eta}{m}\lambda_{\min}(\mathbf{K})$ with $\lambda_{\min}(\mathbf{K})$ being the smallest eigenvalue of $\mathbf{K}$. This indicates that L decays at a linear rate, i.e. $L(T) \leq L(0)\exp(-\frac{2\eta}{m}\lambda_{\min}(\mathbf{K})T)$. In contrast, we show that the rate of convergence of the QNN dynamics *must* be sublinear, slower than the linear convergence rate predicted by the kernel regression model with a positive definite kernel.

Theorem 4.2 (No faster than sublinear convergence)**.** *Consider a QNN instance with a training set* $\mathcal{S} = \{(\boldsymbol{\rho}_j, y_j)\}$ *such that* $\boldsymbol{\rho}_j$ *are pure states and* $y_j \in \{\pm 1\}$*, and a measurement* $\mathbf{M}_0$ *with eigenvalues in* $\{\pm 1\}$*. Under the gradient flow for the objective function* $L(\boldsymbol{\theta}) = \frac{1}{2m}\sum_{j=1}^{m}\mathrm{tr}(\boldsymbol{\rho}_j\mathbf{M}(\boldsymbol{\theta}) - y_j)^2$*, for any ansatz* $\mathbf{U}(\boldsymbol{\theta})$ *defined in Line 4.1,* L *converges to* $zero$ *at most at a sublinear convergence rate. More concretely, for* $\mathbf{U}(\boldsymbol{\theta})$ *generated by* $\{\mathbf{H}^{(l)}\}_{l=1}^{p}$*, let* η *be the learning rate and* m *be the sample size, the objective function at time* t*:*

$$L(\boldsymbol{\theta}(t)) \geq 1/(c_0 + c_1 t)^2. \tag{4.5}$$

Here $c_0 = 1/\sqrt{L(\boldsymbol{\theta}(0))}$ *depends on the objective function at initialization, and* $c_1 = 12\eta\sum_{l=1}^{p}\left\|\mathbf{H}^{(l)}\right\|_{\mathsf{op}}^2$.

The constant c_1 in the theorem depends on the number of parameters p through $\sum_{l=1}^{p}\left\|\mathbf{H}^{(l)}\right\|_{\mathsf{op}}^2$ if the operator norm of $\mathbf{H}^{(l)}$ is a constant of p. We can get rid of the dependency on p by scaling the learning rate η or changing the time scale, which does not affect the sublinearity of convergence.

By expressing the objective function $L(\boldsymbol{\theta}(t))$ as $\frac{1}{2m}\mathbf{r}(\boldsymbol{\theta}(t))^T\mathbf{r}(\boldsymbol{\theta}(t))$, Lemma 4.1 indicates that the decay of $\frac{dL(\boldsymbol{\theta}(t))}{dt}$ is lower-bounded by $\frac{-2\eta}{m}\lambda_{\max}(\mathbf{K}(\boldsymbol{\theta}(t)))L(\boldsymbol{\theta}(t))$, where $\lambda_{\max}(\cdot)$ is the largest eigenvalue of a Hermitian matrix. The full proof of Theorem 4.2 is deferred to Section 4.6.2, and follows from the fact that when the QNN prediction for an input state $\boldsymbol{\rho}_j$ is close to the ground truth $y_j = 1$ or -1, the diagonal entry $K_{jj}(\boldsymbol{\theta}(t))$ vanishes. As a result the largest eigenvalue $\lambda_{\max}(\mathbf{K}(\boldsymbol{\theta}(t)))$ also vanishes as the objective function $L(\boldsymbol{\theta}(t))$ approaches 0 (which is the global

minima). Notice the sublinearity of convergence is independent of the system dimension d, the choices of $\{\mathbf{H}^{(l)}\}_{l=1}^{p}$ in $\mathbf{U}(\boldsymbol{\theta})$ or the number of parameters p. This means that the dynamics of QNN training is completely different from kernel regression even in the limit where d and/or $p \to \infty$.

Experiments: sublinear QNN convergence To support Theorem 4.2, we simulate the training of QNNs using $\mathbf{M}_0$ with eigenvalues ± 1. For dimension $d = 32$ and 64, we randomly sample four d-dimensional pure states that are orthogonal, with two of samples labeled $+1$ and the other two labeled -1. The training curves (plotted under the log scale) in Figure 4.1 flattens as L approaches 0, suggesting the rate of convergence $-d \ln L/dt$ vanishes around global minima, which is a signature of the sublinear convergence. Note that the sublinearity of convergence is independent of the number of parameters p. For gradient flow or gradient descent with sufficiently small step-size, the scaling of a constant learning rate η leads to a scaling of time t and does not fundamentally change the (sub)linearity of the convergence. For the purpose of visual comparison, we scale η with p by choosing the learning rate as $10^{-3}/p$. For more details on the experiments, please refer to Section 4.10.

4.4 Asymptotic Dynamics of QNNs

As demonstrated in the previous section, the dynamics of the QNN training deviates from the kernel regression for any choices of the number of parameters p and the dimension d in the setting of Pauli measurements for classification. This calls for a new characterization of the QNN dynamics in the regime of over-parameterization. For a concrete definition of over-parameterization, we consider the family of the periodic ansatze in Definition 4.1, and refer to the

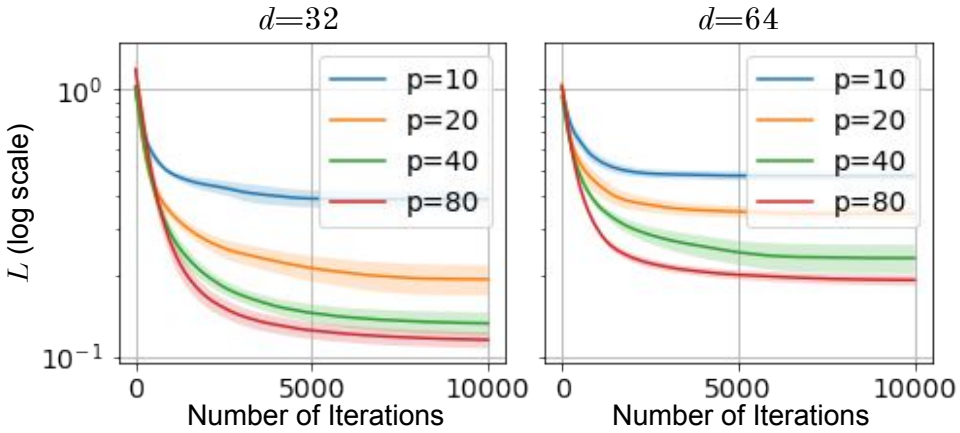


Figure 4.1: Sublinear convergence of QNN training. For QNNs with Pauli measurements for a classification task, the (log-scaled) training curves flatten as the number of iterations increases, indicating a sublinear convergence. The flattening of training curves remains for increasing numbers of parameters $p = 10, 20, 40, 80$. The training curves are averaged over 10 random initialization, and the error bars are the halves of standard deviations.

limit of $p \to \infty$ with a fixed generating Hamiltonian $\mathbf{H}$ as the regime of over-parameterization.

In this section, we derive the asymptotic dynamics of QNN training when number of parameters p in the periodic ansatze goes to infinity. We start by decomposing the dynamics of the residual $\mathbf{r}(\boldsymbol{\theta}(t))$ into a term corresponding to the asymptotic dynamics, and a term of perturbation that vanishes as $p \to \infty$. As mentioned before, in the context of the gradient flow, the choice of η is merely a scaling of the time and therefore arbitrary. For a QNN instance with m training samples and a p-parameter ansatz generated by a Hermitian $\mathbf{H}$ as defined in Line 4.2, we choose η to be $\frac{m}{p}\frac{d^2-1}{\mathrm{tr}(\mathbf{H}^2)}$ to facilitate the presentation:

Lemma 4.3 (Decomposition of the residual dynamics)**.** *Let $\mathcal{S}$ be a training set with m samples $\{(\boldsymbol{\rho}_j, y_j)\}_{j=1}^m$, and let $\mathbf{U}(\boldsymbol{\theta})$ be a p-parameter ansatz generated by a non-zero $\mathbf{H}$ as in Line 4.2. Consider a QNN instance with a training set $\mathcal{S}$, ansatz $\mathbf{U}(\boldsymbol{\theta})$ and a measurement $\mathbf{M}_0$. Under the gradient flow with $\eta = \frac{m}{p}\frac{d^2-1}{\mathrm{tr}(\mathbf{H}^2)}$, the residual vector $\mathbf{r}(t)$ as a function of time t through $\boldsymbol{\theta}(t)$*

evolves as

$$\frac{d\mathbf{r}(t)}{dt} = -(\mathbf{K}_{\mathsf{asym}}(t) + \mathbf{K}_{\mathsf{pert}}(t))\mathbf{r}(t) \tag{4.6}$$

where both $\mathbf{K}_{\mathsf{asym}}$ *and* $\mathbf{K}_{\mathsf{pert}}$ *are functions of time through the parameterized measurement* $\mathbf{M}(\boldsymbol{\theta}(t))$*, such that*

$$(\mathbf{K}_{\mathsf{asym}}(t))_{ij} := \mathrm{tr}\left(\mathrm{i}[\mathbf{M}(t), \boldsymbol{\rho}_i]\ \mathrm{i}[\mathbf{M}(t), \boldsymbol{\rho}_j]\right), \tag{4.7}$$

$$(\mathbf{K}_{\mathsf{pert}}(t))_{ij} := \mathrm{tr}\left(\mathrm{i}[\mathbf{M}(t), \boldsymbol{\rho}_i] \otimes \mathrm{i}[\mathbf{M}(t), \boldsymbol{\rho}_j]\Delta(t)\right). \tag{4.8}$$

Here $\Delta(t)$ *is a* $d^2 \times d^2$ *Hermitian as a function of* t *through* $\boldsymbol{\theta}(t)$*.*

The proof is deferred to Section 4.7. Under the random initialization by sampling $\{\mathbf{U}_l\}_{l=1}^p$ i.i.d. from the haar measure over the special unitary group $SU(d)$, $\Delta(0)$ concentrates at zero as p increases. We further show that $\Delta(t) - \Delta(0)$ has a bounded operator norm decreasing with number of parameters. This allows us to associate the convergence of the over-parameterized QNN with the properties of $\mathbf{K}_{\mathsf{asym}}(t)$:

Theorem 4.4 (Linear convergence of QNN with mean-square loss)**.** *Let* $\mathcal{S}$ *be a training set with* m *samples* $\{(\boldsymbol{\rho}_j, y_j)\}_{j=1}^m$*, and let* $\mathbf{U}(\boldsymbol{\theta})$ *be a* p*-parameter ansatz generated by a non-zero* $\mathbf{H}$ *as in Line 4.2. Consider a QNN instance with the training set* $\mathcal{S}$*, ansatz* $\mathbf{U}(\boldsymbol{\theta})$ *and a measurement* $\mathbf{M}_0$*, trained by gradient flow with* $\eta = \frac{m}{p}\frac{d^2-1}{\mathrm{tr}(\mathbf{H}^2)}$*. Then for sufficiently large number of parameters* p*, if the smallest eigenvalue of* $\mathbf{K}_{\mathsf{asym}}(t)$ *is greater than a constant* C_0*, then with high probability over the random initialization of the periodic ansatz, the loss function converges to zero at a linear*

rate

$$L(t) \leq L(0) \exp(-\frac{C_0 t}{2}). \tag{4.9}$$

We defer the proof to Section 4.8. Similar to $\mathbf{r}(t)$, the evolution of $\mathbf{M}(t)$ decomposes into an asymptotic term

$$\frac{d}{dt}\mathbf{M}(t) = \sum_{j=1}^{m} r_j [\mathbf{M}(t), [\mathbf{M}(t), \boldsymbol{\rho}_j]] \tag{4.10}$$

and a perturbative term depending on $\Delta(t)$. Theorem 4.4 allows us to study the behavior of an over-parameterized QNN by simulating/characterizing the asymptotic dynamics of $\mathbf{M}(t)$, which is significantly more accessible.

Application: QNN with one training sample. To demonstrate the proposed asymptotic dynamics as a tool for analyzing over-parameterized QNNs, we study the convergence of the QNN with one training sample $m = 1$. To set a separation from the regime of the sublinear convergence, consider the following setting: let $\mathbf{M}_0$ be a Pauli measurement, for any input state $\boldsymbol{\rho}$, instead of assigning $\hat{y} = \operatorname{tr}(\boldsymbol{\rho}\mathbf{U}(\boldsymbol{\theta})^\dagger \mathbf{M}_0 \mathbf{U}(\boldsymbol{\theta}))$, take $\gamma \operatorname{tr}(\boldsymbol{\rho}\mathbf{U}(\boldsymbol{\theta})^\dagger \mathbf{M}_0 \mathbf{U}(\boldsymbol{\theta}))$ as the prediction $\hat{y}$ at $\boldsymbol{\theta}$ for a scaling factor $\gamma > 1.0$. The γ-scaling of the measurement outcome can be viewed as a classical processing in the context of quantum information, or as an activation function (or a link function) in the context of machine learning, and is equivalent to a QNN with measurement $\gamma\mathbf{M}_0$. The following corollary implies the convergence of 1-sample QNN for $\gamma > 1.0$ under a mild initial condition:

Corollary 4.5. *Let $\boldsymbol{\rho}$ be a d-dimensional pure state, and let y be ± 1. Consider a QNN instance*

with a Pauli measurement $\mathbf{M}_0$*, an one-sample training set* $\mathcal{S} = \{(\boldsymbol{\rho}, y)\}$ *and an ansatz* $\mathbf{U}(\boldsymbol{\theta})$ *defined in Line 4.2. Assume the scaling factor* $\gamma > 1.0$ *and* $p \to \infty$ *with* $\eta = \frac{d^2-1}{p \operatorname{tr}(\mathbf{H}^2)}$*. Under the initial condition that the prediction at* $t = 0$*,* $\hat{y}(0)$ *is less than 1, the objective function converges linearly with*

$$L(t) \leq L(0) \exp(-C_1 t) \tag{4.11}$$

with the convergence rate $C_1 \geq \gamma^2 - 1$*.*

With a scaling factor γ and training set $\{(\boldsymbol{\rho}_j, y_j)\}_{j=1}^m$, the objective function, as a function of the parameterized measurement $\mathbf{M}(t)$, reads as: $L(\mathbf{M}(t)) = \frac{1}{2m} \sum_{j=1}^m (\gamma \operatorname{tr}(\boldsymbol{\rho}_j \mathbf{M}(t)) - y_j)^2$. As stated in Theorem 4.4, for sufficiently large number of parameters p, the convergence rate of the residual $\mathbf{r}(t)$ is determined by $\mathbf{K}_{\mathsf{asym}}(t)$, as the asymptotic dynamics of $\mathbf{r}(t)$ reads as $\frac{d}{dt}\mathbf{r} = -\mathbf{K}_{\mathsf{asym}}(\mathbf{M}(t))\mathbf{r}(t)$ with the chosen η. For $m = 1$, the asymptotic matrix $\mathbf{K}_{\mathsf{asym}}$ reduces to a scalar $k(t) = -\operatorname{tr}([\gamma \mathbf{M}(t), \boldsymbol{\rho}]^2) = 2(\gamma^2 - \hat{y}(t)^2)$. $\hat{y}(t)$ approaches the label y if $k(t)$ is strictly positive, which is guaranteed for $\hat{y}(t) < \gamma$. Therefore $|\hat{y}(0)| < 1$ implies that $|\hat{y}(t)| < 1$ and $k(t) \geq 2(\gamma^2 - 1)$ for all $t > 0$.

In Figure 4.2 (top), we plot the training curves of one-sample QNNs with $p = 320$ and varying $\gamma = 1.2, 1.4, 2.0, 4.0, 8.0$ with the same learning rate $\eta = 1e - 3/p$. As predicted in Corollary 4.5, the rate of convergence increases with the scaling factor γ. The proof of the corollary additionally implies that $k(t)$ depends on $\hat{y}(t)$: the convergence rate changes over time as the prediction $\hat{y}$ changes. Therefore, despite the linear convergence, the dynamics is different from that of kernel regression, where the kernel remains constant during training in the limit $p \to \infty$.

In Figure 4.2 (bottom), we plot the empirical rate of convergence $-\frac{d}{dt}\ln L(t)$ against the rate predicted by $\hat{y}$. Each data point is calculated for QNNs with different γ at different time steps by differentiating the logarithms of the training curves. The scatter plot displays an approximately linear dependency, indicating the proposed asymptotic dynamics is capable of predicting how the convergence rate changes during training, which is beyond the explanatory power of the kernel regression model. Note that the slope of the linear relation is not exactly one. This is because we choose a learning rate much smaller than η in the corollary statement to simulate the dynamics of gradient flow.

QNN convergence for $m > 1$. To characterize the convergence of QNNs with $m > 1$, we seek to empirically study the asymptotic dynamics in Line 4.10. According to Theorem 4.4, the (linear) rate of convergence is lower-bounded by the smallest eigenvalue of $\mathbf{K}_{\mathsf{asym}}(t)$, up to an constant scaling. In Figure 4.3, we simulate the asymptotic dynamics with various combinations of (γ, d, m), and evaluate the smallest eigenvalue of $\mathbf{K}_{\mathsf{asym}}(t)$ throughout the dynamics (Figure 4.3, details deferred to Section 4.10). For sufficiently large dimension d, the smallest eigenvalue of $\mathbf{K}_{\mathsf{asym}}$ depends on the ratio between the number of samples and the system dimension m/d and is proportional to the square of the scaling factor γ^2.

Empirically, we observe that the smallest convergence rates for training QNNs are obtained near the global minima (See Figure 4.6), suggesting the bottleneck of convergence occurs when L is small.

We now give theoretical evidence that, at most of the global minima, the eigenvalues of $\mathbf{K}_{\mathsf{asym}}$ are lower bounded by $2\gamma^2(1 - 1/\gamma^2 - O(m^2/d))$, suggesting a linear convergence in the neighborhood of these minima. To make this notion precise, we define the uniform measure over global minima as follows: consider a set of pure input states $\{\boldsymbol{\rho}_j = \mathbf{v}_j\mathbf{v}_j^\dagger\}_{j=1}^m$ that are

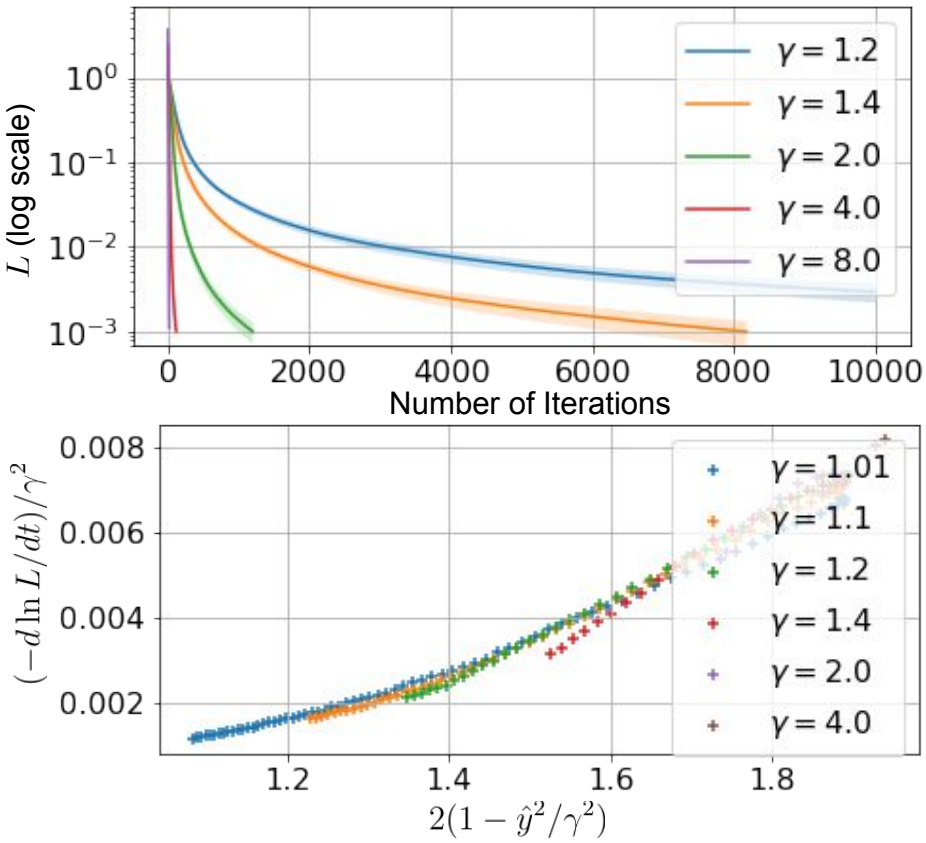


Figure 4.2: (Top) The training curves of one-sample QNNs with varying γ. The smallest convergence rate $-d \ln L/dt$ during training (i.e. the slope of the training curves under the log scale) increases with γ. (Bottom) The convergence rate $-d \ln L/dt|_{t=T}$ as a function of $2(\gamma^2 - \hat{y}^2(T))$ (jointly scaled by $1/\gamma^2$ for visualization) are evaluated at different time steps T for different γ. The approximately linear dependency shows that the proposed dynamics captures the QNN convergence beyond the explanatory power of the kernel regressions.

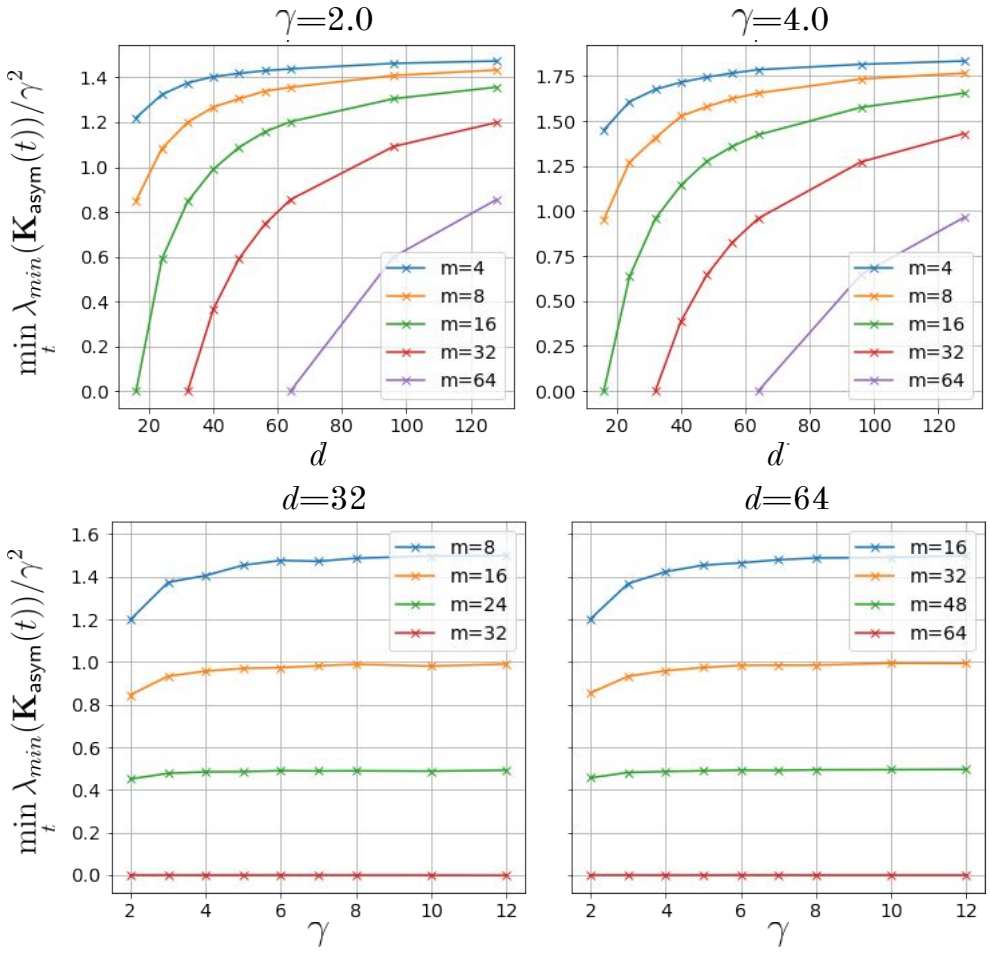


Figure 4.3: The smallest eigenvalue of $\mathbf{K}_{\text{asym}}$ for the asymptotic dynamics with varying system dimension d, scaling factor γ and number of training samples m. For sufficiently large d, the smallest eigenvalue depends on the ratio m/d and is proportional to the square of the scaling factor γ^2.

mutually orthogonal (i.e. $\mathbf{v}_i^\dagger \mathbf{v}_j = 0$ if $i \neq j$). For a large dimension d, the global minima of the asymptotic dynamics is achieved when the objective function is 0. Let $\mathbf{u}_j(t)$ (resp. $\mathbf{w}_j(t)$) denote the components of $\mathbf{v}_j$ projected to the positive (resp. negative) subspace of the measurement $\mathbf{M}(t)$ at the global minima. Recall that for a γ-scaled QNN with a Pauli measurement, the predictions $\hat{y}(t) = \gamma \operatorname{tr}(\rho_j \mathbf{M}(t)) = \gamma(\mathbf{u}_j^\dagger(t)\mathbf{u}_j(t) - \mathbf{w}_j^\dagger(t)\mathbf{w}_j(t))$. At the global minima, we have $\mathbf{u}_j(t) = \frac{1}{2}(1 \pm 1/\gamma)\hat{\mathbf{u}}_j(t)$ for some unit vector $\hat{\mathbf{u}}_j(t)$ for the j-th training sample with label ± 1. On the other hand, given a set of unit vectors $\{\hat{\mathbf{u}}_j\}_{j=1}^m$ in the positive subspace, there is a corresponding set of $\{\mathbf{u}_j(t)\}_{j=1}^m$ and $\{\mathbf{w}_j(t)\}_{j=1}^m$ such that $L = 0$ for sufficiently large d. By uniformly and independently sampling a set of unit vectors $\{\hat{\mathbf{u}}_j\}_{j=1}^m$ from the $d/2$-dimensional subspace associated with the positive eigenvalues of $\mathbf{M}(t)$, we induce a uniform distribution over all the global minima. The next theorem characterizes $\mathbf{K}_{\mathsf{asym}}$ under such an induced uniform distribution over all the global minima:

Theorem 4.6. *Let $\mathcal{S} = \{(\boldsymbol{\rho}_j, y_j)\}_{j=1}^m$ be a training set with orthogonal pure states $\{\boldsymbol{\rho}_j\}_{j=1}^m$ and equal number of positive and negative labels $y_j \in \{\pm 1\}$. Consider the smallest eigenvalue λ_g of $\mathbf{K}_{\mathsf{asym}}$ at the global minima of the asymptotic dynamics of an over-parameterized QNN with the training set $\mathcal{S}$, scaling factor γ and system dimension d. With probability $\geq 1 - \delta$ over the uniform measure over all the global minima*

$$\lambda_g \geq 2\gamma^2\left(1 - \frac{1}{\gamma^2} - C_2 \max\left\{\frac{m^2}{d}, \frac{m}{d}\log\frac{2}{\delta}\right\}\right), \tag{4.12}$$

which is strictly positive for large $\gamma > 1$ and $d = \Omega(\mathsf{poly}(m))$. Here C_2 is a positive constant.

We defer the proof of Theorem 4.6 to Section 4.9. A similar notion of a uniform measure over global minima was also used in [76]. Notice that the uniformness is dependent on the

parameterization of the global minima, and the uniform measure over all the global minima is not necessarily the measure induced by random initialization and gradient-based training. Therefore Theorem 4.6 is not a rigorous depiction of the distribution of convergence rate for a randomly-initialized over-parameterized QNN. Yet the prediction of the theorem aligns well with the empirical observations in Figure 4.3 and suggests that by scaling the QNN measurements, a faster convergence can be achieved: In Figure 4.4, we simulate p-parameter QNNs with dimension $d = 32$ and 64 with a scaling factor $\gamma = 4.0$ using the same setup as in Figure 4.1. The training early stops when the average $L(t)$ over the random seeds is less than 1×10^{-2}. In contrast to Figure 4.1, the convergence rate $-d \ln L/dt$ does not vanish as $L \to 0$, suggesting a simple (constant) scaling of the measurement outcome can lead to convergence within much fewer number of iterations.

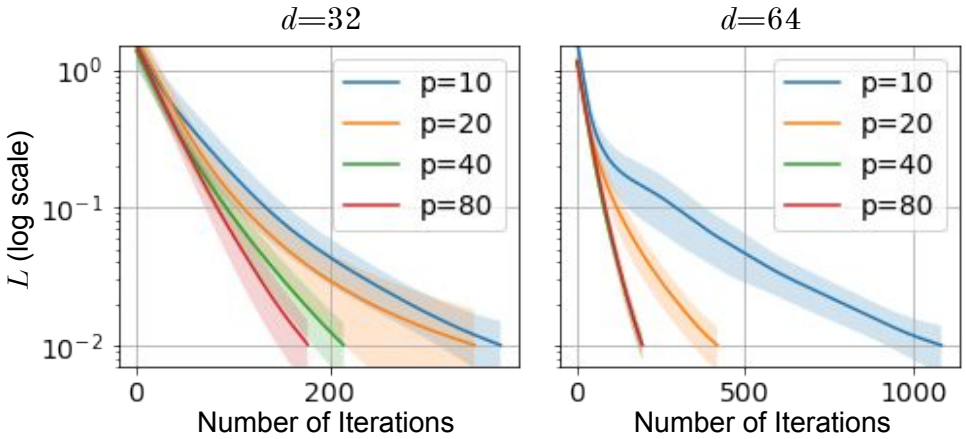


Figure 4.4: Training curves of QNNs with $\gamma = 4.0$ for learning a 4-sample dataset with labels ± 1. For $p = 10, 20, 40, 80$, the rate of convergence is greater than 0 as $L \to 0$, and it takes less than 1000 iterations for L in most of the instances to convergence below 1×10^{-2}. In contrast, in Figure 4.1, $L > 1 \times 10^{-1}$ after 10000 iterations despite the increasing number of parameters.

Another implication of Theorem 4.6 is the deviation of QNN dynamics from any kernel regressions. By straight-forward calculation, the normalized matrix $\mathbf{K}_{\text{asym}}(0)/\gamma^2$ at the random initialization is independent of the choices of γ. In contrast, the typical value of λ_g/γ^2 in

Theorem 4.6 is dependent on γ^2, suggesting non-negligible changes in the matrix $\mathbf{K}_{\mathsf{asym}}(t)$ governing the dynamics of $\mathbf{r}$ for finite scaling factors γ. Such phenomenon is empirically verified in Figure 4.5.

4.5 Conclusion

In this chapter, we characterize the dynamics of QNN training when the number of parameters $p \to \infty$ for different system dimension d, number of training samples m and the scaling factor γ. We show its deviation from the neural tangent kernel regression when either γ or d is finite. The key observation is that, although the QNN dynamics may coincide with that of kernel regression with the Gram matrix $K_{ij} = \mathrm{tr}(\boldsymbol{\rho}_i \boldsymbol{\rho}_j)$ in the large d limit, it deviates from the kernel regression dynamics by a non-vanishing amount as a result of the unitarity constraint.

In the setting of $m > 1$, the proof of the linear convergence of QNN training (Section 4.4) relies on the convergence of the asymptotic QNN dynamics as a premise. Given our empirical results, an interesting future direction might be to rigorously characterize the condition for the convergence of the asymptotic dynamics. Also we mainly consider (variants of) two-outcome measurements $\mathbf{M}_0$ with two eigensubspaces. It might be interesting to look into measurements with more complicated spectrums and see how the shapes of the spectrums affect the rates of convergence.

4.6 Proofs for Section 4.3

4.6.1 Proof of Lemma 4.1

Lemma 4.7 (Dynamics of the residual vector)**.** *Consider a QNN instance with an ansatz* $\mathbf{U}(\boldsymbol{\theta})$ *defined as in Line 4.1, a training dataset* $\mathcal{S} = \{(\boldsymbol{\rho}_j, y_j)\}_{j=1}^m$*, and a measurement* $\mathbf{M}_0$*. Under the gradient flow for the objective function* $L(\boldsymbol{\theta}) = \frac{1}{2m}\sum_{j=1}^m \left(\operatorname{tr}(\boldsymbol{\rho}_j \mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}_0\mathbf{U}(\boldsymbol{\theta})) - y_j\right)^2$ *with learning rate* η*, the residual vector* $\mathbf{r}$ *satisfies the differential equation*

$$\frac{d\mathbf{r}(\boldsymbol{\theta}(t))}{dt} = -\frac{\eta}{m}\mathbf{K}(\mathbf{M}(\boldsymbol{\theta}(t)))\mathbf{r}(\boldsymbol{\theta}(t)), \tag{4.3}$$

where $\mathbf{K}$ *is a positive semi-definite matrix-valued function of the parameterized measurement. The* (i,j)*-th element of* $\mathbf{K}$ *is defined as*

$$\sum_{l=1}^{p} \left(\operatorname{tr}\left(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_i]\mathbf{H}_l\right) \operatorname{tr}\left(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_j]\mathbf{H}_l\right)\right). \tag{4.4}$$

Here $\mathbf{H}_l := \mathbf{U}_0^\dagger \mathbf{U}_{1:l-1}^\dagger(\boldsymbol{\theta})\mathbf{H}^{(l)}\mathbf{U}_{1:l-1}(\boldsymbol{\theta})\mathbf{U}_0$*, is a function of* $\boldsymbol{\theta}$ *with* $\mathbf{U}_{1:r}(\boldsymbol{\theta})$ *being the shorthand for* $\mathbf{U}_r \exp(-i\theta_r \mathbf{H}^{(r)}) \cdots \mathbf{U}_1 \exp(-i\theta_1 \mathbf{H}^{(1)})$.

Proof. For succinctness, we drop the dependency on $\boldsymbol{\theta}(t)$ when there are no ambiguity. The unitary $\mathbf{U}_{r:p}(\boldsymbol{\theta})$ depends on $\boldsymbol{\theta}_l$ for $l \geq r$:

$$\frac{\partial \mathbf{U}_{r:p}}{\partial \theta_l} = \mathbf{U}_{l:p}(\boldsymbol{\theta})(-\mathrm{i}\mathbf{H}^{(l)})\mathbf{U}_{r:l-1}(\boldsymbol{\theta}) = -\mathrm{i}\mathbf{U}_{l:p}\mathbf{H}^{(l)}\mathbf{U}_{l:p}^\dagger \mathbf{U}_{r:p}. \tag{4.13}$$

Therefore for all $l \in [p]$

$$\frac{\partial \mathbf{M}(\boldsymbol{\theta}(t))}{\partial \theta_l} = \mathbf{U}_0^\dagger \left(\frac{\partial \mathbf{U}_{1:p}}{\partial \theta_l} \right)^\dagger \mathbf{M}_0 \mathbf{U}_{1:p} \mathbf{U}_0 + \mathbf{U}_0^\dagger \mathbf{U}_{1:p}^\dagger \mathbf{M}_0 \frac{\partial \mathbf{U}_{1:p}}{\partial \theta_l} \mathbf{U}_0, \tag{4.14}$$

$$= \mathrm{i}(\mathbf{U}_0^\dagger \mathbf{U}_{1:p}^\dagger \mathbf{U}_{l:p} \mathbf{H}^{(l)} \mathbf{U}_{l:p}^\dagger \mathbf{M}_0 \mathbf{U}_{1:p} \mathbf{U}_0) - \mathrm{i}(\mathbf{U}_0^\dagger \mathbf{U}_{1:p}^\dagger \mathbf{M}_0 \mathbf{U}_{l:p} \mathbf{H}^{(l)} \mathbf{U}_{l:p}^\dagger \mathbf{U}_{1:p} \mathbf{U}_0), \tag{4.15}$$

$$= \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta}(t))]. \tag{4.16}$$

By the chain rule with matrix parameters, we have

$$\frac{\partial L(\boldsymbol{\theta}(t))}{\partial \theta_l} = \mathrm{tr}\left(\nabla_{\mathbf{M}} L \frac{\partial \mathbf{M}}{\partial \theta_l} \right) = \mathrm{i}\,\mathrm{tr}(\nabla_{\mathbf{M}} L[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta}(t))]). \tag{4.17}$$

Furthermore, due to the gradient flow dynamics,

$$\frac{d\mathbf{M}(\boldsymbol{\theta}(t))}{dt} = \sum_{l=1}^{p} \frac{d\theta_l}{dt} \frac{\partial \mathbf{M}(\boldsymbol{\theta}(t))}{\partial \theta_l} = -\eta \sum_{l=1}^{p} \frac{\partial L(\boldsymbol{\theta}(t))}{\partial \theta_l} \frac{\partial \mathbf{M}(\boldsymbol{\theta}(t))}{\partial \theta_l}, \tag{4.18}$$

$$= \eta \sum_{l=1}^{p} \mathrm{tr}(\nabla_{\mathbf{M}} L[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta}(t))])[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta}(t))]. \tag{4.19}$$

By plugging in $\nabla_{\mathbf{M}} L = -\frac{1}{m} \sum_{j=1}^{m} r_j \boldsymbol{\rho}_j$, we show that the parameterized measurement $\mathbf{M}(\boldsymbol{\theta}) = \mathbf{U}^\dagger(\boldsymbol{\theta}) \mathbf{M}_0 \mathbf{U}(\boldsymbol{\theta})$ follows the dynamics

$$d\mathbf{M}(\boldsymbol{\theta}(t))/dt \frac{\eta}{m} \sum_{l=1}^{p} \mathrm{tr}\left(\sum_{j=1}^{m} r_j \boldsymbol{\rho}_j \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta}(t))] \right) \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta}(t))]. \tag{4.20}$$

By definition $r_i := y_i - \hat{y}_i$, and

$$\frac{dr_i}{dt} = -\frac{d\,\mathrm{tr}(\boldsymbol{\rho}_i \mathbf{M}(\boldsymbol{\theta}(t)))}{dt} = -\,\mathrm{tr}\left(\boldsymbol{\rho}_i \frac{d\mathbf{M}(\boldsymbol{\theta}(t))}{dt}\right) \tag{4.21}$$

$$= -\frac{\eta}{m}\sum_{l=1}^{p} \mathrm{tr}\left(\sum_{j=1}^{m} r_j \boldsymbol{\rho}_j \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta}(t))]\right) \mathrm{tr}\left(\boldsymbol{\rho}_i \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta}(t))]\right) \tag{4.22}$$

$$= -\frac{\eta}{m}\sum_{j=1}^{m} r_j \Big(\mathrm{tr}\left(\boldsymbol{\rho}_i \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta}(t))]\right) \mathrm{tr}\left(\boldsymbol{\rho}_j \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta}(t))]\right)\Big) \tag{4.23}$$

$$= -\frac{\eta}{m}\sum_{j=1}^{m} r_j \Big(\mathrm{tr}\left(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_i]\mathbf{H}_l\right) \mathrm{tr}\left(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_j]\mathbf{H}_l\right)\Big). \tag{4.24}$$

The last equality is due to the cyclicity of the trace operation. Making the identification $K_{ij}(\mathbf{M}(\boldsymbol{\theta}(t))) = \left(\mathrm{tr}\left(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_i]\mathbf{H}_l\right) \mathrm{tr}\left(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_j]\mathbf{H}_l\right)\right)$, we have $\frac{d\mathbf{r}(\boldsymbol{\theta}(t))}{dt} = -\mathbf{K}(\mathbf{M}(\boldsymbol{\theta}(t)))\mathbf{r}(\boldsymbol{\theta}(t))$. □

4.6.2 Proof of Theorem 4.2

Proof. The mean squared loss function $L(\boldsymbol{\theta}(t))$ can be expressed as $\frac{1}{2m}\mathbf{r}(\boldsymbol{\theta}(t))^T\mathbf{r}(\boldsymbol{\theta}(t))$. Using Lemma 4.1, the rate of convergence can be lower-bounded as

$$\frac{1}{L(\boldsymbol{\theta}(t))}\frac{dL(\boldsymbol{\theta}(t))}{dt} \tag{4.25}$$

$$= \frac{1}{\mathbf{r}(\boldsymbol{\theta}(t))^T\mathbf{r}(\boldsymbol{\theta}(t))}\frac{d}{dt}\mathbf{r}(\boldsymbol{\theta}(t))^T\mathbf{r}(\boldsymbol{\theta}(t)), \tag{4.26}$$

$$= -\frac{2\eta}{m}\cdot\frac{\mathbf{r}(\boldsymbol{\theta}(t))^T\mathbf{K}(\boldsymbol{\theta}(t))\mathbf{r}(\boldsymbol{\theta}(t))}{\mathbf{r}(\boldsymbol{\theta}(t))^T\mathbf{r}(\boldsymbol{\theta}(t))}, \tag{4.27}$$

$$\geq -\frac{2\eta}{m}\lambda_{\max}(\mathbf{K}(\boldsymbol{\theta}(t))). \tag{4.28}$$

The positive semi-definiteness of $\mathbf{K}(\boldsymbol{\theta}(t))$ suggests that $\lambda_{\max}(\mathbf{K}(\boldsymbol{\theta}(t))) \leq \mathrm{tr}(\mathbf{K}(\boldsymbol{\theta}(t)))$. We now proceed to bound $\mathrm{tr}(\mathbf{K}(\boldsymbol{\theta}(t)))$. Since the eigenvalues of $\mathbf{M}_0$ and $\mathbf{M}(\boldsymbol{\theta})$ all lie in $\{\pm 1\}$,

$\mathbf{M}(\boldsymbol{\theta}(t))$ decomposes into the difference of to projections, $\boldsymbol{\Pi}_+(\boldsymbol{\theta}(t))$ and $\boldsymbol{\Pi}_-(\boldsymbol{\theta}(t))$, projecting onto the subspaces associated with eigenvalues of $+1$ and -1 respectively. When $\hat{y}_j$ approaches y_j, the input state $\boldsymbol{\rho}_j$ lies almost completely in one of the eigen-subspaces, leading to a vanishing commutator $\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_j]$ such that $K_{jj}(\boldsymbol{\theta}(t))$ approaches zero:

Let $\mathbf{v}_j$ be the statevector representation of the pure state $\boldsymbol{\rho}_j$, such that $\boldsymbol{\rho}_j = \mathbf{v}_j \mathbf{v}_j^\dagger$. Vector $\mathbf{v}_j$ decomposes into the components within the positive and negative eigen-subspaces of $\mathbf{M}(\boldsymbol{\theta}(t))$: $\mathbf{v}_j = \mathbf{u}_j(\boldsymbol{\theta}(t)) + \mathbf{w}_j(\boldsymbol{\theta}(t))$, where $\mathbf{u}_j(\boldsymbol{\theta}(t)) = \boldsymbol{\Pi}_+(\boldsymbol{\theta}(t))\mathbf{v}_j$ and $\mathbf{w}_j(\boldsymbol{\theta}(t)) = \boldsymbol{\Pi}_-(\boldsymbol{\theta}(t))\mathbf{v}_j$. In the following we omit the arguments $\boldsymbol{\theta}(t)$ in $\mathbf{u}_j$ and $\mathbf{v}_j$ for succinctness, but the time dependence is to be implicitly understood. The commutator between the parameterized measurement and the input state can be written as $[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_j] = 2(\mathbf{u}_j \mathbf{w}_j^\dagger - \mathbf{w}_j \mathbf{u}_j^\dagger)$. Therefore

$$|\operatorname{tr}(\mathrm{i}[\mathbf{M}, \boldsymbol{\rho}_j]\mathbf{H}_l)| \leq 4 \|\mathbf{H}_l\|_{\mathsf{op}} \|\mathbf{u}_j\| \|\mathbf{w}_j\| . \tag{4.29}$$

Assume without loss of generality that the j-th label y_j is $+1$. Then $\|\mathbf{u}_j\|^2 + \|\mathbf{w}_j\|^2 = \|\mathbf{v}_j\|^2 1$ by definition, and $\|\mathbf{u}_j\|^2 - \|\mathbf{w}_j\|^2 = \operatorname{tr}(\mathbf{M}(\boldsymbol{\theta}(t))\boldsymbol{\rho}_j) = y_j - r_j = 1 - r_j$. Then $\|\mathbf{w}_j\|^2 = |r_j|/2$, and $\|\mathbf{u}_j\| \|\mathbf{w}_j\|$

Therefore we have,

$$\begin{aligned} K_{jj}(\boldsymbol{\theta}(t)) =& \sum_{l=1}^{p} \operatorname{tr}^2(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}(t)), \boldsymbol{\rho}_j]\mathbf{H}_l) && (4.30) \\ \leq& 16 \sum_{l=1}^{p} \|\mathbf{H}_l\|_{\mathsf{op}}^2 \frac{|r_j|}{2}(1 - \frac{|r_j|}{2}) && (4.31) \\ \leq& 16 \sum_{l=1}^{p} \|\mathbf{H}_l\|_{\mathsf{op}}^2 \frac{|r_j|}{2} && (4.32) \end{aligned}$$

As a result

$$\frac{1}{L(\boldsymbol{\theta}(t))}\frac{dL(\boldsymbol{\theta}(t))}{dt} \tag{4.33}$$

$$\geq -\frac{2\eta}{m}\operatorname{tr}(\mathbf{K}(\boldsymbol{\theta}(t))) \geq -\frac{2\eta}{m}\sum_{i=1}^{m} K_{ii} \tag{4.34}$$

$$\geq -\frac{16\eta}{m}\sum_{l=1}^{p}\|\mathbf{H}_l\|_{\mathsf{op}}^2\sum_{i=1}^{m}|r_j| \tag{4.35}$$

$$\geq -16\sqrt{2}\eta\sum_{l=1}^{p}\|\mathbf{H}_l\|_{\mathsf{op}}^2\sqrt{L(\boldsymbol{\theta}(t))} \tag{4.36}$$

$$= -16\sqrt{2}\eta\sum_{l=1}^{p}\left\|\mathbf{H}^{(l)}\right\|_{\mathsf{op}}^2\sqrt{L(\boldsymbol{\theta}(t))}. \tag{4.37}$$

Here we use the fact that $\sum_{j=1}^{m}|r_j| \leq \sqrt{m}\sqrt{\mathbf{r}(\boldsymbol{\theta}(t))^T\mathbf{r}(\boldsymbol{\theta}(t))} = \sqrt{2m^2L(\boldsymbol{\theta}(t))}$.

The theorem statement follows directly by integrating the inequality above:

$$L(\boldsymbol{\theta}(t))^{-\frac{3}{2}}dL(\boldsymbol{\theta}(t)) \geq -24\eta\sum_{l=1}^{p}\left\|\mathbf{H}^{(l)}\right\|_{\mathsf{op}}^2 dt \tag{4.38}$$

$$\implies -2d(L(\boldsymbol{\theta}(t))^{-1/2}) \geq -24\eta\sum_{l=1}^{p}\left\|\mathbf{H}^{(l)}\right\|_{\mathsf{op}}^2 dt \tag{4.39}$$

$$\implies L(\boldsymbol{\theta}(T))^{-\frac{1}{2}} - L(\boldsymbol{\theta}(0))^{-\frac{1}{2}} \leq 12\eta\sum_{l=1}^{p}\left\|\mathbf{H}^{(l)}\right\|_{\mathsf{op}}^2 T \tag{4.40}$$

$$\implies L(\boldsymbol{\theta}(T))^{-\frac{1}{2}} - c_0 \leq c_1 T \tag{4.41}$$

□

4.7 Proof of Lemma 4.3

Lemma 4.8 (Decomposition of the residual dynamics)**.** *Let $\mathcal{S}$ be a training set with m samples $\{(\boldsymbol{\rho}_j, y_j)\}_{j=1}^m$, and let $\mathbf{U}(\boldsymbol{\theta})$ be a p-parameter ansatz generated by a non-zero $\mathbf{H}$ as in Line 4.2. Consider a QNN instance with a training set $\mathcal{S}$, ansatz $\mathbf{U}(\boldsymbol{\theta})$ and a measurement $\mathbf{M}_0$. Under the gradient flow with $\eta = \frac{m}{p}\frac{d^2-1}{\mathrm{tr}(\mathbf{H}^2)}$, the residual vector $\mathbf{r}(t)$ as a function of time t through $\boldsymbol{\theta}(t)$ evolves as*

$$\frac{d\mathbf{r}(t)}{dt} = -(\mathbf{K}_{\mathsf{asym}}(t) + \mathbf{K}_{\mathsf{pert}}(t))\mathbf{r}(t) \tag{4.6}$$

where both $\mathbf{K}_{\mathsf{asym}}$ and $\mathbf{K}_{\mathsf{pert}}$ are functions of time through the parameterized measurement $\mathbf{M}(\boldsymbol{\theta}(t))$, such that

$$(\mathbf{K}_{\mathsf{asym}}(t))_{ij} := \mathrm{tr}\left(\mathrm{i}[\mathbf{M}(t), \boldsymbol{\rho}_i]\ \mathrm{i}[\mathbf{M}(t), \boldsymbol{\rho}_j]\right), \tag{4.7}$$

$$(\mathbf{K}_{\mathsf{pert}}(t))_{ij} := \mathrm{tr}\left(\mathrm{i}[\mathbf{M}(t), \boldsymbol{\rho}_i] \otimes \mathrm{i}[\mathbf{M}(t), \boldsymbol{\rho}_j]\Delta(t)\right). \tag{4.8}$$

Here $\Delta(t)$ is a $d^2 \times d^2$ Hermitian as a function of t through $\boldsymbol{\theta}(t)$.

Throughout the proof, we make use of the following notations. Let $\mathcal{H}$ be a d-dimensional Hilbert space, and let $\{\mathbf{e}_a\}_{a\in[d]}$ be a basis of $\mathcal{H}$. We use $\mathbf{I}_{d\times d}$ denote the identity matrix $\sum_{a\in[d]} \mathbf{e}_a\mathbf{e}_a^\dagger$. We use $\otimes$ for kronecker products on vectors, matrices and Hilbert spaces. For the $d^2 \times d^2$-dimensional product space $\mathcal{H} \otimes \mathcal{H}$, let $\mathbf{W}_{d^2\times d^2}$ denote the swap matrix $\sum_{a,b\in[d]} \mathbf{e}_a\mathbf{e}_b^\dagger \otimes \mathbf{e}_b\mathbf{e}_a^\dagger$.

We will also make use of the well-known integration formula with respect to the haar measure over d-dimensional unitaries (see e.g. [59] for more details).

Proof. As proven in Lemma 4.1, we track the dynamics of the parameterized measurement $\mathbf{M}(\boldsymbol{\theta})$:

$$\frac{d\mathbf{M}(\boldsymbol{\theta})}{dt} = \sum_{l=1}^{p} \frac{d\theta_l}{dt} \cdot \frac{\partial \mathbf{M}(\boldsymbol{\theta})}{\partial \theta_l} \tag{4.42}$$

$$= \sum_{l=1}^{p} (-\eta) \operatorname{tr}\left(\mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta})] \nabla_{\mathbf{M}} L\right) \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta})] \tag{4.43}$$

$$= \sum_{l=1}^{p} \eta \operatorname{tr}\left(\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \mathbf{H}_l\right) \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta})] \tag{4.44}$$

$$= \sum_{l=1}^{p} \eta \mathrm{i}[\operatorname{tr}\left(\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \mathbf{H}_l\right) \mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta})] \tag{4.45}$$

$$= \sum_{l=1}^{p} \eta \mathrm{i}[\operatorname{tr}_1\left((\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \otimes \mathbf{I})(\mathbf{H}_l \otimes \mathbf{H}_l)\right), \mathbf{M}(\boldsymbol{\theta})] \tag{4.46}$$

Let $Z(\mathbf{H}, d)$ denote the ratio $\frac{\operatorname{tr}(\mathbf{H}^2)}{d^2-1}$, the learning rate η can be expressed as $\frac{m}{pZ(\mathbf{H},d)}$. Let $\mathbf{Y}(\boldsymbol{\theta}(t))$ denote the normalized $d^2 \times d^2$-complex matrix $\frac{1}{pZ(\mathbf{H},d)} \sum_{l=1}^{p} \mathbf{H}_l \otimes \mathbf{H}_l$ for $\mathbf{H}_l$ defined in Lemma 4.1 and let $\mathbf{Y}^\star$ denote $\mathbf{W}_{d^2 \times d^2} - \frac{1}{d}\mathbf{I}_{d^2 \times d^2}$, the asymptotic version of $\mathbf{Y}$. We can accordingly decompose the dynamics into the asymptotic dynamics and the deviation (perturbation) from the asymptotic

dynamics:

$$\frac{d\mathbf{M}(\boldsymbol{\theta})}{dt} =(\eta p Z(\mathbf{H}, d))\mathrm{i}[\mathrm{tr}_1\left((\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \otimes \mathbf{I})\mathbf{Y}\right), \mathbf{M}(\boldsymbol{\theta})] \tag{4.47}$$

$$=(\eta p Z(\mathbf{H}, d))\mathrm{i}[\mathrm{tr}_1\left((\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \otimes \mathbf{I})\mathbf{Y}^\star\right), \mathbf{M}(\boldsymbol{\theta})] \tag{4.48}$$

$$+ (\eta p Z(\mathbf{H}, d))\mathrm{i}[\mathrm{tr}_1\left((\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \otimes \mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star)\right), \mathbf{M}(\boldsymbol{\theta})] \tag{4.49}$$

$$=(\eta p Z(\mathbf{H}, d))\mathrm{i}[(\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})], \mathbf{M}(\boldsymbol{\theta})] \tag{4.50}$$

$$+ (\eta p Z(\mathbf{H}, d))\mathrm{i}[\mathrm{tr}_1\left((\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \otimes \mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star)\right), \mathbf{M}(\boldsymbol{\theta})] \tag{4.51}$$

$$= - (\eta p Z(\mathbf{H}, d))[\mathbf{M}(\boldsymbol{\theta}), [\mathbf{M}(\boldsymbol{\theta}), \nabla_{\mathbf{M}} L]] \tag{4.52}$$

$$- (\eta p Z(\mathbf{H}, d))[\mathbf{M}(\boldsymbol{\theta}), \mathrm{tr}_1\left(([\mathbf{M}(\boldsymbol{\theta}), \nabla_{\mathbf{M}} L] \otimes \mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star)\right)] \tag{4.53}$$

Plugging in that $\nabla_{\mathbf{M}} L(\mathbf{M}(\boldsymbol{\theta})) = -\frac{1}{m}\sum_{i=1}^m r_i \boldsymbol{\rho}_i$ with the residual $r_i := y_i - \hat{y}_i = \mathrm{tr}(\mathbf{M}(\boldsymbol{\theta})\rho_i) - y_i$:

$$\frac{d\mathbf{M}(\boldsymbol{\theta})}{dt} = \sum_{j=1}^m r_j[\mathbf{M}(\boldsymbol{\theta}), [\mathbf{M}(\boldsymbol{\theta}), \boldsymbol{\rho}_j]] \tag{4.54}$$

$$+ \sum_{j=1}^m r_j[\mathbf{M}(\boldsymbol{\theta}), \mathrm{tr}_1\left(([\mathbf{M}(\boldsymbol{\theta}), \boldsymbol{\rho}_j] \otimes \mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star)\right)] \tag{4.55}$$

Trace after multiplying $\boldsymbol{\rho}_i$ on both sides:

$$\frac{dr_i}{dt} = -\mathrm{tr}(\boldsymbol{\rho}_i \frac{d\mathbf{M}(\boldsymbol{\theta})}{dt}) = -\sum_{j=1}^m r_j \,\mathrm{tr}\left(\boldsymbol{\rho}_i[\mathbf{M}(\boldsymbol{\theta}), [\mathbf{M}(\boldsymbol{\theta}), \boldsymbol{\rho}_j]]\right) \tag{4.56}$$

$$- \sum_{j=1}^m r_j \,\mathrm{tr}\left(\boldsymbol{\rho}_i[\mathbf{M}(\boldsymbol{\theta}), \mathrm{tr}_1\left(([\mathbf{M}(\boldsymbol{\theta}), \boldsymbol{\rho}_j] \otimes \mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star)\right)]\right)$$

$$\tag{4.57}$$

The lemma follows directly from rearranging: for the first term,

$$
\begin{aligned}
&-\sum_{j=1}^{m} r_j \operatorname{tr}(\boldsymbol{\rho}_i[\mathbf{M}(\boldsymbol{\theta}),[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_j]]) && (4.58)\\
=&-\sum_{j=1}^{m} r_j \operatorname{tr}([\boldsymbol{\rho}_i,\mathbf{M}(\boldsymbol{\theta})][\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_j]) && (4.59)\\
=&-\sum_{j=1}^{m} r_j \operatorname{tr}(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_i]\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_j]). && (4.60)
\end{aligned}
$$

For the second term,

$$
\begin{aligned}
&-\sum_{j=1}^{m} r_j \operatorname{tr}\left(\boldsymbol{\rho}_i[\mathbf{M}(\boldsymbol{\theta}),\operatorname{tr}_1\left(([\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_j]\otimes\mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t))-\mathbf{Y}^\star)\right)]\right) && (4.61)\\
=&-\sum_{j=1}^{m} r_j \operatorname{tr}\left(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_i]\operatorname{tr}_1\left((\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_j]\otimes\mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t))-\mathbf{Y}^\star)\right)\right) && (4.62)\\
=&-\sum_{j=1}^{m} r_j \operatorname{tr}\left((\mathbf{I}\otimes\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_i])(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_j]\otimes\mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t))-\mathbf{Y}^\star)\right) && (4.63)\\
=&-\sum_{j=1}^{m} r_j \operatorname{tr}\left((\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_j]\otimes\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_i])(\mathbf{Y}(\boldsymbol{\theta}(t))-\mathbf{Y}^\star)\right) && (4.64)\\
=&-\sum_{j=1}^{m} r_j \operatorname{tr}\left((\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_i]\otimes\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\boldsymbol{\rho}_j])(\mathbf{Y}(\boldsymbol{\theta}(t))-\mathbf{Y}^\star)\right) && (4.65)
\end{aligned}
$$

The last equality follows from the fact that $\mathbf{Y}$ and $\mathbf{Y}^\star$ are invariant under the swapping of spaces.

The lemma follows by identifying the matrix $\Delta(t)$ with $\mathbf{Y}(\boldsymbol{\theta}(t))-\mathbf{Y}^\star$. □

4.8 Proof of Theorem 4.4

Theorem 4.4 (Linear convergence of QNN with mean-square loss)**.** *Let $\mathcal{S}$ be a training set with m samples $\{(\boldsymbol{\rho}_j,y_j)\}_{j=1}^m$, and let $\mathbf{U}(\boldsymbol{\theta})$ be a p-parameter ansatz generated by a non-zero $\mathbf{H}$ as in*

Line 4.2. Consider a QNN instance with the training set $\mathcal{S}$, ansatz $\mathbf{U}(\boldsymbol{\theta})$ and a measurement $\mathbf{M}_0$, trained by gradient flow with $\eta = \frac{m}{p}\frac{d^2-1}{\mathrm{tr}(\mathbf{H}^2)}$. Then for sufficiently large number of parameters p, if the smallest eigenvalue of $\mathbf{K}_{\mathsf{asym}}(t)$ is greater than a constant C_0, then with high probability over the random initialization of the periodic ansatz, the loss function converges to zero at a linear rate

$$L(t) \leq L(0)\exp(-\frac{C_0 t}{2}). \tag{4.9}$$

Proof. In Lemma 4.3, we decompose the QNN dynamics into the asymptotic term and the perturbation term depending on $\Delta(t) = \mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star$. We now show that the use of the terms "asymptotic" and "perturbation" are exact, by showing that $\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star$ vanishes as $p \to \infty$. We make use of the characterization of a similarly-defined quantity in [87], restated as Lemma 4.9 and 4.11, such that for sufficiently large p, $\|\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star\|_{\mathsf{op}}$ vanishes for all t with high probability over the randomness in $\{\mathbf{U}_l\}_{l=0}^p$. Recall that the perturbation term $\mathbf{K}_{\mathsf{pert}}$ is defined as

$$(\mathbf{K}_{\mathsf{pert}}(t))_{ij} := \mathrm{tr}\left((\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}), \boldsymbol{\rho}_i] \otimes \mathrm{i}[\mathbf{M}(\boldsymbol{\theta}), \boldsymbol{\rho}_j])(\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star)\right). \tag{4.66}$$

By choosing sufficiently large p, we have $\|\mathbf{K}_{\mathsf{pert}}(t)\|_{\mathsf{op}} \leq C_0/10$ and therefore the loss function converging to zero at a rate $\geq C_0/2$. □

Lemma 4.9 (Concentration at initialization, adapted from Lemma 3.4 in [87])**.** *Over the randomness of ansatz initialization (i.e. for $\{\mathbf{U}_l\}_{l=1}^p$ sampled $i.i.d.$ with respect to the Haar measure), for any*

initial $\boldsymbol{\theta}(0)$*, with probability* $1-\delta$*:*

$$\|\mathbf{Y}(\boldsymbol{\theta}(0))-\mathbf{Y}^\star\|_{\mathsf{op}} \le \frac{1}{\sqrt{p}}\cdot\frac{2\|\mathbf{H}\|_{\mathsf{op}}^2}{Z}\sqrt{2\log\frac{d^2}{\delta}}. \tag{4.67}$$

See Section 3.11 for the proof.

As we pointed out in the main body, a vanishing perturbation term at initialization is not sufficient to guarantee the term remain perturbative throughout the training. We now show in Lemma 4.11 that $\mathbf{Y}(\boldsymbol{\theta}(t))-\mathbf{Y}^\star$ remain small during training by showing $\mathbf{Y}(\boldsymbol{\theta}(t))-\mathbf{Y}((0))$ vanishes in the limit $p\to\infty$. But before that, we show that, while the QNN predictions changes much during training, the change in the parameters measured in ℓ_2- or ℓ_∞-norm ($\|\boldsymbol{\theta}(t)-\boldsymbol{\theta}(0)\|_2$ or $\|\boldsymbol{\theta}(t)-\boldsymbol{\theta}(0)\|_\infty$) vanishes as $p\to\infty$ during the training of QNN:

Lemma 4.10 (Slow-varying θ in QNNs)**.** *Suppose that under learning rate* $\eta=\frac{m}{pZ(\mathbf{H},d)}$*, for all* $0\le t\le T$*, the loss function* $L(\boldsymbol{\theta}(t))\le L(\boldsymbol{\theta}(0))\exp(-at)$ *for some constant* a*, then for all* $0\le t_1,t_2\le T$*:*

$$\|\boldsymbol{\theta}(t_2)-\boldsymbol{\theta}(t_1)\|_\infty \le \frac{1}{p}\frac{\sqrt{2}m\|\mathbf{H}\|_F\|\mathbf{M}\|_F\sqrt{L(\boldsymbol{\theta}(0))}}{Z}|t_1-t_2|, \tag{4.68}$$

$$\|\boldsymbol{\theta}(t_2)-\boldsymbol{\theta}(t_1)\|_2 \le \frac{1}{\sqrt{p}}\frac{\sqrt{2}m\|\mathbf{H}\|_F\|\mathbf{M}\|_F\sqrt{L(\boldsymbol{\theta}(0))}}{Z}|t_1-t_2|. \tag{4.69}$$

Proof. We first bound the absolute value of the derivative $\frac{d\theta_l}{dt}$:

$$|\frac{d\theta_l}{dt}| = \eta|\frac{\partial L}{\partial\theta_l}| = \frac{\eta}{2m}|\sum_{i=1}^m r_i\,\mathrm{tr}(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}),\mathbf{H}_l]\boldsymbol{\rho}_i)|. \tag{4.70}$$

Plugging in $\eta = \frac{m}{pZ}$, we have

$$|\frac{d\theta_l}{dt}| = \frac{1}{2pZ}|\sum_{i=1}^{m} r_i \operatorname{tr}(\mathrm{i}[\mathbf{M}(\boldsymbol{\theta}), \mathbf{H}_l]\boldsymbol{\rho}_i)| = \frac{1}{2pZ}|\langle \mathbf{r}, \mathbf{a}\rangle|, \tag{4.71}$$

where the vector $\mathbf{a}$ is defined such that $a_j = \operatorname{tr}(i[\mathbf{M}(\boldsymbol{\theta}), \mathbf{H}_l]\boldsymbol{\rho}_j)$ for $j \in [m]$. The ℓ_2-norm of $\mathbf{a}$

$$\begin{aligned}
\|\mathbf{a}\|_2^2 &= \sum_{j=1}^{m} \operatorname{tr}^2(\mathrm{i}[\mathbf{M}, \mathbf{H}_l]\boldsymbol{\rho}_j) && (4.72)\\
&= \operatorname{tr}\left((i[\mathbf{M}, \mathbf{H}_l])^{\otimes 2} \sum_{j=1}^{m} \boldsymbol{\rho_j}^{\otimes 2}\right) && (4.73)\\
&\leq \left\|(i[\mathbf{M}, \mathbf{H}_l])^{\otimes 2}\right\|_F \left\|\sum_{j=1}^{m} \boldsymbol{\rho_j}^{\otimes 2}\right\|_F && (4.74)\\
&\leq \|i[\mathbf{M}, \mathbf{H}_l]\|_F^2 \left\|\sum_{j=1}^{m} \boldsymbol{\rho_j}^{\otimes 2}\right\|_F && (4.75)\\
&\leq (2\|\mathbf{M}\|_F \|\mathbf{H}_l\|_F)^2 \sum_{j=1}^{m} \left\|\boldsymbol{\rho_j}^{\otimes 2}\right\|_F && (4.76)\\
&\leq (2\|\mathbf{M}\|_F \|\mathbf{H}\|_F \sqrt{m})^2. && (4.77)
\end{aligned}$$

Therefore we can bound $|\frac{d\theta_l}{dt}|$ as

$$\begin{aligned}
|\frac{d\theta_l}{dt}| &\leq \frac{1}{2pZ}\|\mathbf{r}\|_2\|\mathbf{a}\|_2 && (4.78)\\
&\leq \frac{1}{2pZ}\sqrt{2mL(\boldsymbol{\theta}(t))} \cdot 2\|\mathbf{M}\|_F \|\mathbf{H}\|_F \sqrt{m} && (4.79)\\
&= \frac{1}{p}\frac{\sqrt{2}m\|\mathbf{M}\|_F \|\mathbf{H}\|_F}{Z}\sqrt{L(\boldsymbol{\theta}(t))} && (4.80)\\
&\leq \frac{1}{p}\frac{\sqrt{2}m\|\mathbf{M}\|_F \|\mathbf{H}\|_F}{Z}\sqrt{L(\boldsymbol{\theta}(0))}\exp(-at/2) && (4.81)
\end{aligned}$$

Hence for all $l \in [p]$:

$$|\theta_l(t_2) - \theta_l(t_1)| = |\int_{t_1}^{t_2} dt d\theta_l(t)/dt| \leq \int_{t_1}^{t_2} dt |d\theta_l(t)/dt| \tag{4.82}$$

$$\leq \int_{t_1}^{t_2} dt \frac{1}{p} \frac{\sqrt{2}m \|\mathbf{M}\|_F \|\mathbf{H}\|_F}{Z} \sqrt{L(\boldsymbol{\theta}(0))} \exp(-at/2) \tag{4.83}$$

$$\leq \frac{2}{a} \cdot \frac{1}{p} \frac{\sqrt{2}m \|\mathbf{M}\|_F \|\mathbf{H}\|_F}{Z} \sqrt{L(\boldsymbol{\theta}(0))} |\exp(-at_1/2) - \exp(-at_2/2)| \tag{4.84}$$

$$\leq \frac{1}{p} \frac{\sqrt{2}m \|\mathbf{M}\|_F \|\mathbf{H}\|_F}{Z} \sqrt{L(\boldsymbol{\theta}(0))} |t_1 - t_2| \tag{4.85}$$

$$\tag{4.86}$$

The bounds on the ℓ_2- and ℓ_∞-norm follows from direct computation. □

We are now ready to show $\mathbf{Y}(t_2) - \mathbf{Y}(t_1)$ vanishes as $p \to \infty$:

Lemma 4.11 (Concentration during training, adapted from Lemma 3.5 in [87])**.** *Suppose that under learning rate* $\eta = \frac{m}{pZ(\mathbf{H},d)}$*, for all* $0 \leq t \leq T$*, the loss function* $L(\boldsymbol{\theta}(t))$ *decreases as* $L(\boldsymbol{\theta}(0)) \exp(-at)$ *then with probability* $\geq 1 - \delta$*, for all* $0 \leq t \leq T$*:* $\|\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}(\boldsymbol{\theta}(0))\|_{\mathsf{op}} \leq C_3 \cdot \frac{T}{\sqrt{p}}$*, where* C_3 *is a constant of* T *and* p*.*

See Section 3.11 for the proof.

4.9 Proof for Theorem 4.6

In this section, we present the proof for Theorem 4.6 for characterizing the rate of convergence at global minima:

Theorem 4.6. *Let* $\mathcal{S} = \{(\boldsymbol{\rho}_j, y_j)\}_{j=1}^m$ *be a training set with orthogonal pure states* $\{\boldsymbol{\rho}_j\}_{j=1}^m$ *and equal number of positive and negative labels* $y_j \in \{\pm 1\}$*. Consider the smallest eigenvalue* λ_g

of $\mathbf{K}_{\mathsf{asym}}$ *at the global minima of the asymptotic dynamics of an over-parameterized QNN with the training set* $\mathcal{S}$*, scaling factor* γ *and system dimension* d*. With probability* $\geq 1-\delta$ *over the uniform measure over all the global minima*

$$\lambda_g \geq 2\gamma^2(1-\frac{1}{\gamma^2}-C_2\max\{\frac{m^2}{d},\frac{m}{d}\log\frac{2}{\delta}\}), \tag{4.12}$$

which is strictly positive for large $\gamma > 1$ *and* $d = \Omega(\mathsf{poly}(m))$*. Here* C_2 *is a positive constant.*

We start by presenting a few helper lemma:

4.9.1 Helper lemma for $\mathbf{K}_{\mathsf{asym}}$

Lemma 4.12. *Let* $\mathbf{A},\mathbf{B}$ *be* $d\times d$ *Hermitians. Let* $\|\cdot\|_{\mathsf{op}}$ *denote the operator norm of a given Hermitian and let* $\circ$ *denote the Hadamard product (i.e. the elementwise multiplication) of two matrices, we have*

$$\|\mathbf{A}\circ\mathbf{B}\|_{\mathsf{op}} \leq \|\mathbf{A}\|_{\mathsf{op}}\|\mathbf{B}\|_{\mathsf{op}}. \tag{4.87}$$

Proof. For any $d\times d$ Hermitian matrix, let $\lambda_i(\cdot)$ denote its i-th smallest eigenvalue. The Hadamard product $\mathbf{A}\circ\mathbf{B}$ is a $d\times d$ principal submatrix of the Kronecker product $\mathbf{A}\otimes\mathbf{B}$, and by the Poincaré separation theorem (see e.g. Corollary 4.3.37 in [89]):

$$\lambda_1(\mathbf{A}\otimes\mathbf{B}) \leq \lambda_i(\mathbf{A}\circ\mathbf{B}) \leq \lambda_{d^2}(\mathbf{A}\otimes\mathbf{B}). \tag{4.88}$$

The statement follows from the fact that the eigenvalues of $\mathbf{A}\otimes\mathbf{B}$ take the form of $\lambda_i(\mathbf{A})\lambda_j(\mathbf{B})$

for $i, j \in [d]$. $\square$

Lemma 4.13 ($\mathbf{K}_{\mathsf{asym}}$ for asymptotic dynamics)**.** *Let $\mathcal{S}$ be a m-sample training set composed of pure states $\{\boldsymbol{\rho}_j = \mathbf{v}_j\mathbf{v}_j^\dagger\}_{j=1}^m$. Let $\mathbf{M}_0$ be a Pauli-like measurement with eigenvalues ± 1 and trace-0. Consider training a QNN with $\mathcal{S}$, measurement $\mathbf{M}_0$ and a scaling factor of γ. The positive semidefinite matrix $\mathbf{K}_{\mathsf{asym}}$ can be expressed entry-wise as*

$$(\mathbf{K}_{\mathsf{asym}})_{ij}(\mathbf{M}(t)) = 8\gamma^2 Re(\mathbf{u}_j^\dagger(t)\mathbf{u}_i(t)\mathbf{w}_i^\dagger(t)\mathbf{w}_j(t)), \tag{4.89}$$

where $\mathbf{u}_i(t) := \mathbf{\Pi}_+(t)\mathbf{v}_i$ (resp. $\mathbf{w}_i(t) := \mathbf{\Pi}_-(t)\mathbf{v}_i$) is the projection of $\mathbf{v}_i$ into the postive (resp. negative) subspace of $\mathbf{M}(t) = \gamma(\mathbf{\Pi}_+(t) - \mathbf{\Pi}_-(t))$. Let $\mathbf{P}(t) := (\mathbf{u}_i^\dagger(t)\mathbf{u}_j(t))_{i.j\in[m]}$ and $\mathbf{N}(t) := (\mathbf{w}_i^\dagger(t)\mathbf{w}_j(t))_{i,j\in[m]}$ be the Gram matrices of $\{\mathbf{u}_i(t)\}_{i=1}^m$ and $\{\mathbf{w}_i(t)\}_{i=1}^m$, we have:

$$\lambda_{\min}(\mathbf{K}_{\mathsf{asym}}(t)) \geq 8\gamma^2\lambda_{\min}(\mathbf{P}(t)) \min_{i\in[m]}(\mathbf{N}_{ii}(t)) \geq 8\gamma^2\lambda_{\min}(\mathbf{P}(t))\lambda_{\min}(\mathbf{N}(t)). \tag{4.90}$$

Proof. For succinctness, we drop the time dependency t when there are no ambiguities. Calculate the expression of $(\mathbf{K}_{\mathsf{asym}})_{ij}$ for pure states $\boldsymbol{\rho}_i = \mathbf{v}_i\mathbf{v}_i^\dagger$:

$$\begin{aligned}
(\mathbf{K}_{\mathsf{asym}}(\mathbf{M}(t)))_{ij} &= \operatorname{tr}\left(\mathrm{i}[\mathbf{M}, \boldsymbol{\rho}_i]\ \mathrm{i}[\mathbf{M}, \boldsymbol{\rho}_j]\right) && (4.91)\\
&= \operatorname{tr}\left(\mathbf{M}^2\boldsymbol{\rho}_i\boldsymbol{\rho}_j\right) + \operatorname{tr}\left(\mathbf{M}^2\boldsymbol{\rho}_j\boldsymbol{\rho}_i\right) - 2\operatorname{tr}\left(\mathbf{M}\boldsymbol{\rho}_i\mathbf{M}\boldsymbol{\rho}_j\right) && (4.92)\\
&= 2\gamma^2\big(\operatorname{tr}(\boldsymbol{\rho}_i\boldsymbol{\rho}_j) - \operatorname{tr}((\mathbf{\Pi}_+ - \mathbf{\Pi}_-)\boldsymbol{\rho}_i(\mathbf{\Pi}_+ - \mathbf{\Pi}_-)\boldsymbol{\rho}_j)\big) && (4.93)
\end{aligned}$$

Plugging in $\boldsymbol{\rho}_i = \mathbf{v}_i\mathbf{v}_i^\dagger$, we have:

$$\frac{1}{2\gamma^2}(\mathbf{K}_{\mathsf{asym}}(\mathbf{M}(t)))_{ij} = |\mathbf{u}_i^\dagger\mathbf{u}_j + \mathbf{w}_i^\dagger\mathbf{w}_j|^2 - |(\mathbf{u}_i + \mathbf{w}_i)^\dagger(\boldsymbol{\Pi}_+ - \boldsymbol{\Pi}_-)(\mathbf{u}_j + \mathbf{w}_j)|^2 \tag{4.94}$$

$$= |\mathbf{u}_i^\dagger\mathbf{u}_j + \mathbf{w}_i^\dagger\mathbf{w}_j|^2 - |(\mathbf{u}_i + \mathbf{w}_i)^\dagger(\mathbf{u}_j - \mathbf{w}_j)|^2 \tag{4.95}$$

$$= |\mathbf{u}_i^\dagger\mathbf{u}_j + \mathbf{w}_i^\dagger\mathbf{w}_j|^2 - |\mathbf{u}_i^\dagger\mathbf{u}_j - \mathbf{w}_i^\dagger\mathbf{w}_j|^2 \tag{4.96}$$

$$= 2\mathbf{u}_i^\dagger\mathbf{u}_j \cdot \mathbf{w}_j^\dagger\mathbf{w}_i + 2\mathbf{u}_j^\dagger\mathbf{u}_i \cdot \mathbf{w}_i^\dagger\mathbf{w}_j \tag{4.97}$$

$$= 4Re(\mathbf{u}_j^\dagger\mathbf{u}_i\mathbf{w}_i^\dagger\mathbf{w}_j), \tag{4.98}$$

or $(\mathbf{K}_{\mathsf{asym}}(\mathbf{M}(t)))_{ij} = 8\gamma^2 Re(\mathbf{u}_j^\dagger\mathbf{u}_i\mathbf{w}_i^\dagger\mathbf{w}_j)$.

Let $\mathbf{P}(t)$ and $\mathbf{N}(t)$ be the Gram matrices for $\{\mathbf{u}_i(t)\}_{i=1}^m$ and $\{\mathbf{w}_i(t)\}_{i=1}^m$:

$$(\mathbf{P}(t))_{ij} = \mathbf{u}(t)_i^\dagger\mathbf{u}(t)_j,\ (\mathbf{N}(t))_{ij} = \mathbf{w}(t)_i^\dagger\mathbf{w}(t)_j, \tag{4.99}$$

the matrix $\mathbf{K}_{\mathsf{asym}}$ can be expressed as $\mathbf{K}_{\mathsf{asym}} = 4\gamma^2\mathbf{P} \circ \mathbf{N}^T + 4\gamma^2\mathbf{P}^T \circ \mathbf{N}$, where $\circ$ denotes the Hadamard product, with $\mathbf{P}$ and $\mathbf{N}$ being positive semidefinite matrices. Following a result of Schur's (e.g. see Lemma 6.5 in [85]), we estimate the smallest eigenvalue of $\mathbf{K}_{\mathsf{asym}}$ as

$$\lambda_{\min}(\mathbf{K}_{\mathsf{asym}}(\mathbf{M}(\boldsymbol{\theta}))) \geq 8\gamma^2 \max\big(\min_{i\in[m]}(\mathbf{N}_{ii})\lambda_{\min}(\mathbf{P}), \min_{i\in[m]}(\mathbf{P}_{ii})\lambda_{\min}(\mathbf{N})\big). \tag{4.100}$$

□

The second statement in the limit suggests that the $\mathbf{K}_{\mathsf{asym}}$ is positive definite unless the subspaces spanned by $\mathbf{u}_j$ or $\mathbf{w}_j$ are not full rank, though we do not make use of this fact in the proof of Theorem 4.6.

4.9.2 Proof of Theorem 4.6

Proof. For each input state $\boldsymbol{\rho}_j = \mathbf{v}_j\mathbf{v}_j^\dagger$, let $\mathbf{u}_j$ and $\mathbf{w}_j$ denote the projection of $\mathbf{v}_j$ onto the positive and negative subspaces of the measurement. Since the measurment is updated throughout the training, $\mathbf{u}_j$ and $\mathbf{w}_j$ are functions of time. For a QNN with the scaling factor γ, the QNN prediction for the input state $\boldsymbol{\rho}_j$ at time t is $\hat{y}_j = \gamma(\mathbf{u}_j^\dagger(t)\mathbf{u}_j(t) - \mathbf{w}_j^\dagger(t)\mathbf{w}_j(t))$. Additionally by the normalization of quantum states and the orthogonality of the training sample, we have $\mathbf{u}_j^\dagger(t)\mathbf{u}_j(t) + \mathbf{w}_j^\dagger(t)\mathbf{w}_j(t) = \delta_{ij}$, where δ_{ij} is the Kronecker delta function. Combining these two conditions, we can solve that $\mathbf{u}_j^\dagger\mathbf{u}_j = \frac{1}{2}(1 \pm 1/\gamma)$ and $\mathbf{v}_j^\dagger\mathbf{v}_j = \frac{1}{2}(1 \mp 1/\gamma)$ for $y_j = \pm 1$.

By Lemma 4.13, the diagonal entries $(\mathbf{K}_{\mathsf{asym}})_{jj} = 8\gamma^2 Re(\mathbf{u}_j^\dagger\mathbf{u}_j\mathbf{w}_j^\dagger\mathbf{w}_j) = 8\gamma^2 \cdot \frac{1}{2}(1 \pm 1/\gamma) \cdot \frac{1}{2}(1 \mp 1/\gamma) = 2\gamma^2(1 - 1/\gamma^2)$.

Without loss of generality, assume $y_1 = y_2 = \cdots = y_{m/2} = 1$ and $y_{m/2+1} = y_{m/2+2} = \cdots y_m = -1$. Then $\mathbf{u}_j = \sqrt{\frac{1+1/\gamma}{2}}\hat{\mathbf{u}}_j$ for $1 \le j \le m/2$ and $\mathbf{u}_j = \sqrt{\frac{1-1/\gamma}{2}}\hat{\mathbf{u}}_j$ for $m/2 + 1 \le j \le m$. Here $\hat{\mathbf{u}}_j$ are unit vectors defined as $\mathbf{u}_j/\sqrt{\mathbf{u}_j^\dagger\mathbf{u}_j}$. For the off-diagonal entries, $(\mathbf{K}_{\mathsf{asym}})_{ij} = 8\gamma^2 Re(\mathbf{u}_i^\dagger\mathbf{u}_j\mathbf{w}_j^\dagger\mathbf{w}_i) = 8\gamma^2 Re(\mathbf{u}_i^\dagger\mathbf{u}_j \cdot (-\mathbf{u}_j^\dagger\mathbf{u}_i)) = -8\gamma^2|\mathbf{u}_i^\dagger\mathbf{u}_j|^2$. For the first equality we use the orthogonality among $\{\mathbf{v}_j\}_{j=1}^m$.

Define $m \times m$ Hermitian $\mathbf{G}$ such that $G_{ij} = \hat{\mathbf{u}}_i^\dagger\hat{\mathbf{u}}_j$ and $\mathbf{R}$ such that $R_{ij} = \frac{1}{2}(1 + 1/\gamma)$ for $1 \le i, j \le m/2$, $R_{ij} = \frac{1}{2}(1 - 1/\gamma)$ for $m/2 + 1 \le i, j \le m$, and $R_{ij} = \frac{1}{2}\sqrt{1 - 1/\gamma^2}$ for $1 \le i \le m/2, m/2 + 1 \le j \le m$ or $m/2 + 1 \le i \le m, 1 \le j \le m/2$. The off-diagonal entries can be expressed $-8\gamma^2 R_{ij}G_{ij}G_{ji}$.

Using the notations of $\mathbf{R}$ and $\mathbf{G}$, the matrix $\mathbf{K}_{\mathsf{asym}}$ at the global minima can be expressed

as

$$\mathbf{K}_{\mathsf{asym}} = 2\gamma^2(1-1/\gamma^2)\mathbf{I} - 8\gamma^2\mathbf{R}\circ(\mathbf{G}-\mathbf{I})\circ(\mathbf{G}^T-\mathbf{I}), \tag{4.101}$$

where $\mathbf{I}$ is the $m \times m$ identity matrix.

Eigenvalues of $\mathbf{R}$. Let $\mathbf{e}_1$ and $\mathbf{e}_2$ denote the unit vectors

$$\mathbf{e}_1 = \sqrt{\frac{2}{m}}(1,1,\cdots,1,0,0,\cdots,0)^T \tag{4.102}$$

$$\mathbf{e}_2 = \sqrt{\frac{2}{m}}(0,0\cdots,0,1,1,\cdots,1)^T \tag{4.103}$$

that are zero in the first (last) $m/2$ entries. The matrix $\mathbf{R}$ can be written as

$$\frac{m}{2}(\frac{1}{2}(1+1/\gamma)\mathbf{e}_1\mathbf{e}_1^\dagger + \frac{1}{2}(1-1/\gamma)\mathbf{e}_2\mathbf{e}_2^\dagger + \frac{1}{2}\sqrt{1-1/\gamma^2}\mathbf{e}_1\mathbf{e}_2^\dagger + \frac{1}{2}\sqrt{1-1/\gamma^2}\mathbf{e}_2\mathbf{e}_1^\dagger) \tag{4.104}$$

and can be shown to have eigenvalues $(\frac{m}{2},0,\cdots,0)$ by straight-forward calculation.

Eigenvalues of $\mathbf{G}$. Over the uniform measure over all the global minima, the direction vectors $\hat{\mathbf{u}}_i$ are sampled independently and uniformly from a $d/2$-dimensional (complex) sphere. By the approximate isometric properties (see e.g. Theorem 5.58 in [90]), the gram matrix $\mathbf{G}$ of $\{\hat{\mathbf{u}}_j\}_{j=1}^m$ is approximately an isometry: with probability $\geq 1-2\exp(-c_p t^2)$

$$\|\mathbf{G}-\mathbf{I}\|_{\mathsf{op}} \leq c_m \frac{\max\{\sqrt{m},t\}}{\sqrt{d}} \tag{4.105}$$

for constants c_p and c_m.

Applying Lemma 4.12 to $\mathbf{R}$, $\mathbf{G}-\mathbf{I}$ and $\mathbf{G}^T-\mathbf{I}$, we have that with probability $\geq 1-\delta$, the

smallest eigenvalues of $\mathbf{K}_{\mathsf{asym}}$ at global minima is greater than or equal to

$$2\gamma^2(1 - 1/\gamma^2 - C_2 \max\{\frac{m^2}{d}, \frac{m\log(2/\delta)}{d}\}) \tag{4.106}$$

for some constant $C_2 > 0$. □

4.10 More on Experiments

4.10.1 Experiment details

Our numerical experiments involve simulating both quantum neural networks and the asymptotic dynamics.

QNN simulation. We simulate the QNN experiments using the framework of Pytorch [91] with the periodic ansatze defined in Definition 4.1. The generating Hamiltonian $\mathbf{H}$ are chosen to be a d-dimensional diagonal matrix with $d/2$ $\sqrt{d - d^{-1}}$ and $d/2$ $-\sqrt{d - d^{-1}}$ on the diagonal (normalized such that $\mathrm{tr}(\mathbf{H^2})/(d^2 - 1) = 1$). Each instance of the experiments is specified by the number of samples m, system dimension d, number of parameters p and the scaling factor γ. A m-sample dataset is generated by randomly sampled m orthogonal pure states $\{\mathbf{v}_i\}_{i=1}^m \in \mathbb{C}^d$ and randomly assigned half of the samples with label $+1$ and the other half label -1 (i.e. $\{y_i\}_{i=1}^m \subset \{\pm 1\}^m$).

The optimizer we use is the standard gradient descent optimizer. To simulate the dynamics of gradient flow, we choose the learning rate to be $0.001/p$ and the maximum number of epochs is set to be 10000. We run the experiments on Amazon EC2 C5 Instances.

Asymptotic dynamics simulation. Theorem 4.4 allows us to examine the behavior of QNN

dynamics when $p \to \infty$ by studying the asymptotic dynamics:

$$\frac{d\mathbf{M}(t)}{dt} = -\eta \sum_{j=1}^{m} r_j [\mathbf{M}(t), [\mathbf{M}(t), \boldsymbol{\rho}_j]], \quad \text{where } \forall j \in [m], r_j := \operatorname{tr}(\mathbf{M}(t)\boldsymbol{\rho}_j) - y_j. \tag{4.107}$$

For a QNN asymptotic dynamics with number of samples m, system dimension d and scaling factor γ, we initialize $\mathbf{M}(0)$ as

$$\gamma \mathbf{U} \begin{bmatrix} +1 & 0 & \cdots & 0 & 0 \\ 0 & +1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & -1 & 0 \\ 0 & 0 & \cdots & 0 & -1 \end{bmatrix} \mathbf{U}^\dagger \tag{4.108}$$

with $\mathbf{U}$ being a $d \times d$ haar random unitary. Similar to the QNN simulation, the training set is chosen to be m orthogonal pure states with labels randomly sampled from $\{\pm 1\}$. The simulation of the asymptotic dynamics is run on Intel Core i7-7700HQ Processor (2.80Ghz) with 16G memory.

4.10.2 $\mathbf{K}_{\mathsf{asym}}$ as a function of t

In Corollary 4.5, we see that the convergence rates for one-sample QNNs change significantly during training. Theorem 4.4 allows us further verify this observation for training sets with $m > 1$ by simulating the asymptotic dynamics.

In Figure 4.5, we plot the relative change of the $\mathbf{K}_{asym}(t)$ defined as

$$(\mathbf{K}_{asym}(t))_{ij} := \mathrm{tr}\left(i[\mathbf{M}(t), \boldsymbol{\rho}_i] i[\mathbf{M}(t), \boldsymbol{\rho}_j]\right). \tag{4.109}$$

Each of the data point is averaged over 100 random initialization of $\mathbf{M}(0)$. It is observed that $\mathbf{K}_{asym}(t)$ changes significantly ($\geq 5\%$) for each of the hyperparameters d, m and γ. Therefore we conclude that the deviation from the neural tangent kernel regression is ubiquitous in general for practical settings. Particularly it rules out the existing belief that the $d \to \infty$ alone can lead to a neural tangent kernel-like behavior in QNNs. Same is observed for over-parameterized QNNs (Figure 4.6)

4.11 Over-parameterizations for General Variational Algorithms

In this section, we discussion extensions of our characterization of the asymptotic dynamics to general variational algorithms. Broadly, we consider variational algorithms with loss function $L(\boldsymbol{\theta})$ that depends on $\boldsymbol{\theta}$ through the parameterized measurement $\mathbf{M}(\boldsymbol{\theta})$ (i.e. $L(\boldsymbol{\theta}) = L(\mathbf{M}(\boldsymbol{\theta}))$). We provide a characterization of the asymptotic dynamics if the loss function L satisfies a property we will refer to as robust and fast-convergence:

Definition 4.2 (Robust and fast convergence)**.** A loss function $L(\boldsymbol{\theta}) = L(\mathbf{M}(\boldsymbol{\theta}))$ is said to be have the property of robust and fast convergence, if the projected gradient decay exponentially as

$$\left\| [\mathbf{M}(\boldsymbol{\theta}(t)), \nabla_{\mathbf{M}} L(\boldsymbol{\theta}(t))] \right\|_F \leq A \exp(-at) \tag{4.110}$$

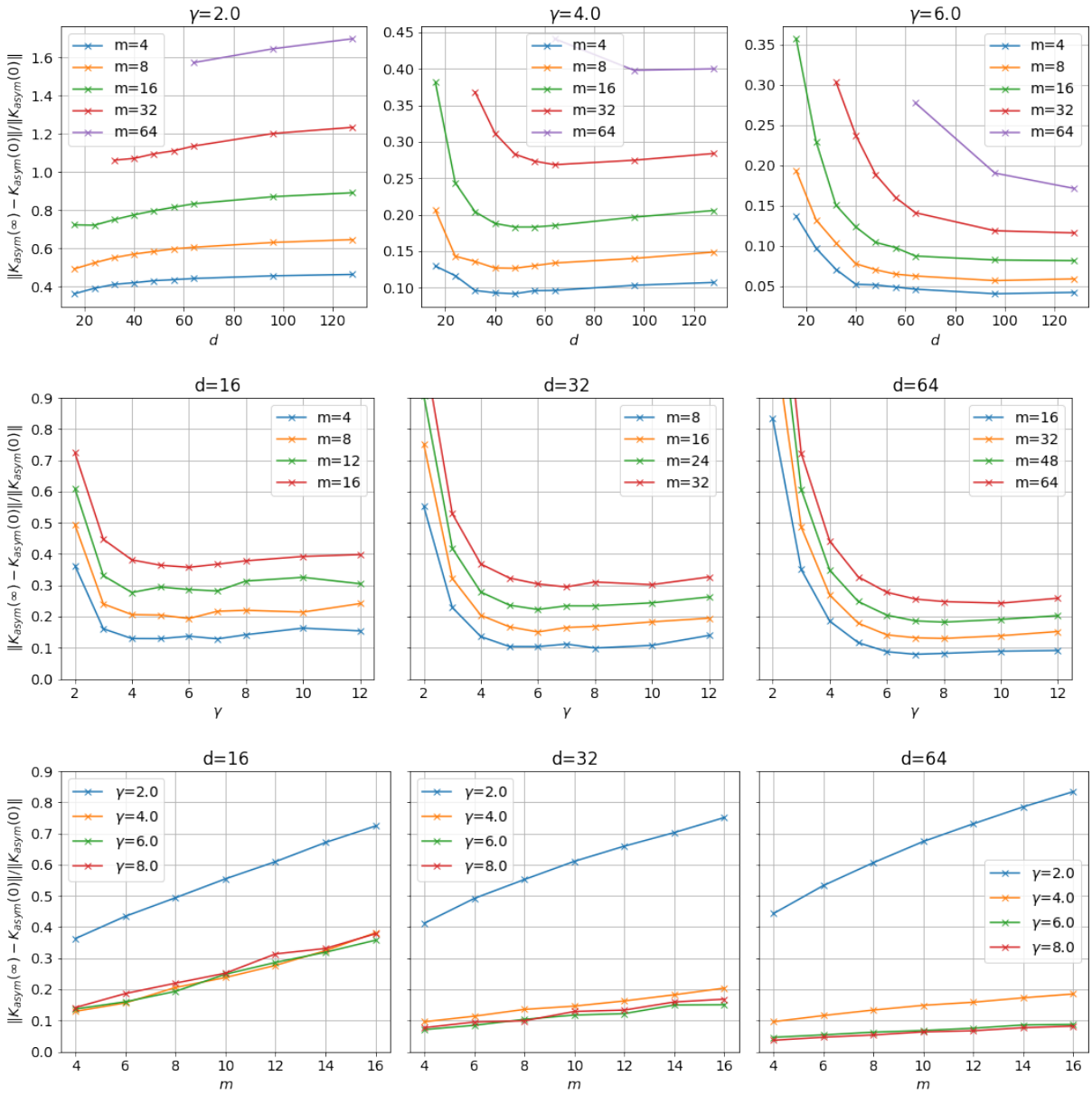


Figure 4.5: Relative change of $\mathbf{K}_{\text{asym}}(t)$ in the QNN asymptotic dynamics for varying system dimension d, scaling factor γ and number of training samples m. $\mathbf{K}_{asym}(t)$ changes significantly ($\geq 5\%$) throughout training.

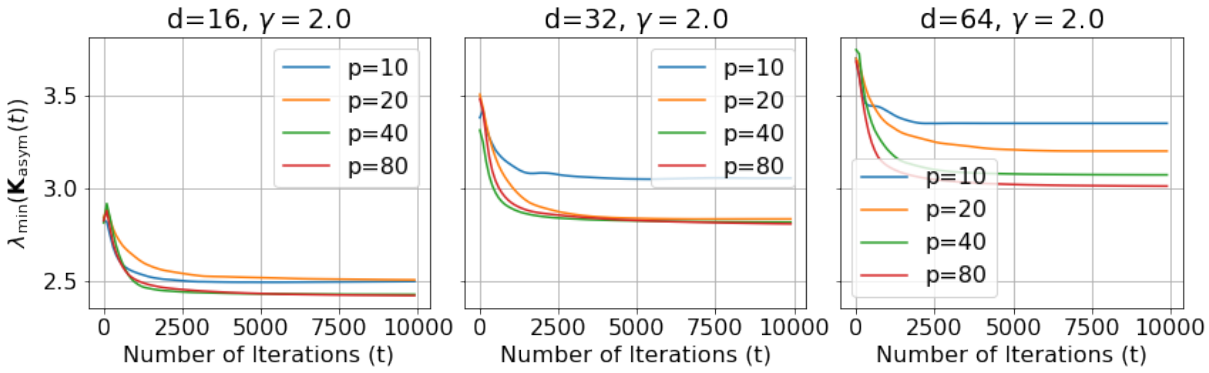


Figure 4.6: Change of the $\lambda_{\min}(\mathbf{K}_{\text{asym}}(t))$ during the QNN training in QNNs with $m = 4, \gamma = 2.0$ and varying d.

for some constants A and a, under the perturbed dynamics

$$\frac{d\mathbf{M}(\boldsymbol{\theta})}{dt} = -[\mathbf{M}(\boldsymbol{\theta}), [\mathbf{M}(\boldsymbol{\theta}), \nabla_{\mathbf{M}} L]] - [\mathbf{M}(\boldsymbol{\theta}), \mathrm{tr}_1\left(([\mathbf{M}(\boldsymbol{\theta}), \nabla_{\mathbf{M}} L] \otimes \mathbf{I})(\mathcal{E}(t))\right)] \tag{4.111}$$

where $\{\mathcal{E}(t)\}_{t\geq 0}$ are $d^2 \times d^2$ Hermitians bounded in operator norms by ϵ, where ϵ is a sufficiently small positive number.

For any L satisfy the property of robust and fast convergence, its asymptotic dynamics in the limit $p \to \infty$ is

$$\frac{d\mathbf{M}(\boldsymbol{\theta})}{dt} = -[\mathbf{M}(\boldsymbol{\theta}), [\mathbf{M}(\boldsymbol{\theta}), \nabla_{\mathbf{M}} L]]. \tag{4.112}$$

This follows directly by plugging the following Lemma 4.14 into the proof of Theorem 4.4 and note that the deviation of $\boldsymbol{\theta}$ is bounded due to the exponential decay of gradients.

Lemma 4.14 (Decomposition of the general dynamics)**.** *Consider a loss function L depending on the partially trainable ansatz through the parameterized measurement $\mathbf{M}(\boldsymbol{\theta}) = \mathbf{U}(\boldsymbol{\theta})^\dagger \mathbf{M}_0 \mathbf{U}(\boldsymbol{\theta})$. Then under the gradient flow wit learning rate $\eta = \frac{1}{pZ(\mathbf{H},d)}$, the dynamics of the parameterized*

measurement $\mathbf{M}(\boldsymbol{\theta}(t))$ *can be expressed as*

$$\frac{d\mathbf{M}(\boldsymbol{\theta})}{dt} = -[\mathbf{M}(\boldsymbol{\theta}), [\mathbf{M}(\boldsymbol{\theta}), \nabla_{\mathbf{M}} L]] - [\mathbf{M}(\boldsymbol{\theta}), \operatorname{tr}_1\left(([\mathbf{M}(\boldsymbol{\theta}), \nabla_{\mathbf{M}} L] \otimes \mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star)\right)]. \tag{4.113}$$

A similar lemma can be proved for the parameterized output state $\boldsymbol{\rho}(\boldsymbol{\theta}) = \mathbf{U}(\boldsymbol{\theta})\boldsymbol{\rho}_0\mathbf{U}^\dagger(\boldsymbol{\theta})$ (i.e. switching to the Schroedinger picture from the Heisenberg picture) for the loss function $L(\boldsymbol{\theta}) = L(\boldsymbol{\rho}(\boldsymbol{\theta}))$.

Proof. Similar to Lemma 4.3, we consider the general form of $L(\boldsymbol{\theta}) = L(\mathbf{M}(\boldsymbol{\theta}))$ and track the dynamics of the parameterized measurement $\mathbf{M}(\boldsymbol{\theta})$:

$$\begin{aligned}
\frac{d\mathbf{M}(\boldsymbol{\theta})}{dt} &= \sum_{l=1}^{p} \frac{d\theta_l}{dt} \cdot \frac{\partial \mathbf{M}(\boldsymbol{\theta})}{\partial \theta_l} && (4.114)\\
&= \sum_{l=1}^{p} (-\eta) \operatorname{tr}\left(\mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta})] \nabla_{\mathbf{M}} L\right) \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta})] && (4.115)\\
&= \sum_{l=1}^{p} \eta \operatorname{tr}\left(\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \mathbf{H}_l\right) \mathrm{i}[\mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta})] && (4.116)\\
&= \sum_{l=1}^{p} \eta \mathrm{i}[\operatorname{tr}\left(\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \mathbf{H}_l\right) \mathbf{H}_l, \mathbf{M}(\boldsymbol{\theta})] && (4.117)\\
&= \sum_{l=1}^{p} \eta \mathrm{i}[\operatorname{tr}_1\left((\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \otimes \mathbf{I})(\mathbf{H}_l \otimes \mathbf{H}_l)\right), \mathbf{M}(\boldsymbol{\theta})] && (4.118)
\end{aligned}$$

Plug in $\eta = \frac{1}{pZ(\mathbf{H},d)}$. Let $\mathbf{Y}(\boldsymbol{\theta}(t))$ denote the normalized $d^2 \times d^2$-complex matrix $\frac{1}{pZ\mathbf{H},d)} \sum_{l=1}^{p} \mathbf{H}_l \otimes \mathbf{H}_l$ and let $\mathbf{Y}^\star$ denote $\mathbf{W}_{d^2 \times d^2} - \frac{1}{d}\mathbf{I}_{d^2 \times d^2}$, the asymptotic version of $\mathbf{Y}$. We can accordingly decompose the dynamics into the asymptotic dynamics and the deviation (perturbation) from the

asymptotic dynamics:

$$\frac{d\mathbf{M}(\boldsymbol{\theta})}{dt} = \mathrm{i}[\mathrm{tr}_1\left((\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \otimes \mathbf{I})\mathbf{Y}\right), \mathbf{M}(\boldsymbol{\theta})] \tag{4.119}$$

$$= \mathrm{i}[\mathrm{tr}_1\left((\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \otimes \mathbf{I})\mathbf{Y}^\star\right), \mathbf{M}(\boldsymbol{\theta})] \tag{4.120}$$

$$+ \mathrm{i}[\mathrm{tr}_1\left((\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \otimes \mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star)\right), \mathbf{M}(\boldsymbol{\theta})] \tag{4.121}$$

$$= \mathrm{i}[(\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})], \mathbf{M}(\boldsymbol{\theta})] \tag{4.122}$$

$$+ \mathrm{i}[\mathrm{tr}_1\left((\mathrm{i}[\nabla_{\mathbf{M}} L, \mathbf{M}(\boldsymbol{\theta})] \otimes \mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star)\right), \mathbf{M}(\boldsymbol{\theta})] \tag{4.123}$$

$$= - [\mathbf{M}(\boldsymbol{\theta}), [\mathbf{M}(\boldsymbol{\theta}), \nabla_{\mathbf{M}} L]] \tag{4.124}$$

$$- [\mathbf{M}(\boldsymbol{\theta}), \mathrm{tr}_1\left(([\mathbf{M}(\boldsymbol{\theta}), \nabla_{\mathbf{M}} L] \otimes \mathbf{I})(\mathbf{Y}(\boldsymbol{\theta}(t)) - \mathbf{Y}^\star)\right)]. \tag{4.125}$$

□

Chapter 5: Principled Designs of Variational Quantum Eigensolvers

5.1 Sampling for VQE Performance Evaluation

5.1.1 Estimating d_{eff} and κ_{eff}

Given a VQE problem $(\mathbf{M}, |\Phi\rangle, \mathbf{U})$ with a compatible ansatz design $\mathcal{A}$, we can estimate the column space of $\mathbf{Q}$ of the invariant subspace by estimating the support of the matrix

$$\hat{\mathbf{\Pi}} = \frac{1}{R}\sum_{r=1}^{R} \mathbf{U}_r |\Phi\rangle\langle\Phi| \mathbf{U}_r^{\dagger} \tag{5.1}$$

with $\mathbf{U}_r$ sampled $i.i.d.$ from the Haar measure over $G_{\mathcal{A}}$. Empirically we approximate the Haar measure over $G_{\mathcal{A}}$ by calculating

$$\mathbf{U}(\boldsymbol{\phi}) = \prod_{l'=1}^{L_{\mathsf{sample}}} \prod_{k=1}^{K} \exp(-i\phi_{l',k}\mathbf{H}^{(k)}) \tag{5.2}$$

for large L_{sample} and randomly initialized $\{\phi_{l',k}\}_{k\in[K],l'\in[L_{\mathsf{sample}}]}$ (throughout this work $\phi_{l,k}$ are sampled uniformly and $i.i.d.$ from the whole real space). Any orthonormal basis of the support of $\mathbf{\Pi}$ can be used as $\mathbf{Q}$ to estimate d_{eff} and κ_{eff} using Definition 3.6. The computational cost for the procedure depends on the quantities R and L_{sample}, and can be $\mathsf{poly}(d_{\mathsf{eff}})$ in the worst case, therefore we do not claim a fundamental superiority in terms of computational complexity for

large d and d_{eff} when compared with the standard practice of directly training VQE over multiple random initializations and sweeping different number of parameters. However, we do observe in our experiments that the estimation of d_{eff} and κ_{eff} for a family of problem Hamiltonians is tremendously faster than training over multiple random initializations and varying number of parameters for a single problem Hamiltonian. For example, it takes < 0.2 hours to evaluate transverse field ising model with up to 10-qubit for transverse field ranging from 0.1 to 1.5 on an Amazon C5 EC2 instance, while it takes 5 hours to characterize a 4-qubit instance with transverse field $g = 0.3$ by performing training using the same machine.

Example: Kitaev model. For a concrete example, consider the HVA for the Kitaev model on square-octagon lattice with external field introduced in [1]. We will see that the proper ansatz design leads to an effective dimension much smaller than the system dimension ($d_{\mathsf{eff}} = 76$ v.s. $d = 256$) and that the effective ratio κ_{eff} can be orders of magnitudes smaller than $\kappa = \frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}$ (Figure 5.3).

The problem Hamiltonian for Kitaev models with external field is defined as

$$\mathbf{M}_{\mathsf{Kitaev}}(J_{xy}, h) = \sum_{(u,v)\in S_Z} Z_u Z_v + \frac{J_{xy}}{\sqrt{2}}\Big(\sum_{(u,v)\in S_X} X_u X_v + \sum_{(u,v)\in S_Y} Y_u Y_v\Big) + h \sum_{i=0}^{7} \big(X_i + Y_i + Z_i\big) \tag{5.3}$$

with X_i denoting the Pauli-X matrix acting on the i-th qubit. This system has coupling in the X, Y, Z directions on edge sets S_X, S_Y and S_Z respectively. The parameter J_{xy} controls the coupling in the X/Y-direction and h controls the strength of the external field. For 8-qubit Kitaev models on square-octagon lattice, by labeling each qubit with indexes 0 through 7, the edge sets are defined as $S_X = \{(0,1),(2,3)\}$, $S_Y = \{(1,2),(0,3)\}$, and $S_Z = \{(4,0),(1,5),(3,7),(2,6)\}$

(See Figure 5.1 or Figure 1(c) in [1]).

We use the ansatz proposed in [1] $\mathcal{A} = \{\mathbf{H}^{(1)}, \cdots, \mathbf{H}^{(6)}\}$ with

$$\begin{aligned}
\mathbf{H}^{(1)} &\propto \sum_{(u,v)\in S_X} X_u X_v, \mathbf{H}^{(4)} \propto \sum_{i=0}^{7} X_i, \\
\mathbf{H}^{(2)} &\propto \sum_{(u,v)\in S_Y} Y_u Y_v, \mathbf{H}^{(5)} \propto \sum_{i=0}^{7} Y_i, \\
\mathbf{H}^{(3)} &\propto \sum_{(u,v)\in S_Z} Z_u Z_v, \mathbf{H}^{(6)} \propto \sum_{i=0}^{7} Z_i.
\end{aligned} \tag{5.4}$$

In Figure 5.2, we plot the eigenvalues of $\hat{\mathbf{\Pi}}$ for the Kitaev models for input state $|\Phi\rangle = |0\rangle^{\otimes 8}$ and the ansatz specified in Equation (5.4) with L_{sample} chosen to be 20. The x-axis correponds to the indices of eigenvalues for the 256×256 problem Hamiltonian, and the y-axis corresponds to the sorted eigenvalues. The spectrums are color-coded for different R ranging from 1 to 100, with blue corresponding to small R and red corresponding to large R. For small R, $\hat{\mathbf{\Pi}}$ is restricted to a small subspace. As the number of samples R increases, the rank of $\hat{\mathbf{\Pi}}$ increases, and converges to a matrix with uniform eigenvalues. Figure 5.2 indicates that the $|\Phi\rangle$ lies within the 76-dimensional invariant subspace V embedded in a 256-dimensional state space $\mathcal{H}$. It is also verified that the ground state of $\mathbf{M}_{\mathsf{Kitaev}}$ lies within the subspace V as well. We also compare the effective ratio κ_{eff} with $\kappa = \frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}$ (i.e. the effective ratio for generic ansatz designs) for a wide range of parameters (J_{xy}, h) in Figure 5.3. We observe that the HVA proposed in [1] reduces κ_{eff} by orders of magnitudes.

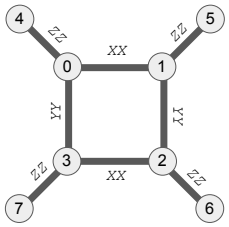


Figure 5.1: Configuration of the 8-qubit Kitaev model on square-octagon lattice defined in [1]. Qubits are labeled by $0, 1, \cdots, 7$, and each edge corresponds to an interation term. The types of interactions XX, YY and ZZ are as specified in texts.

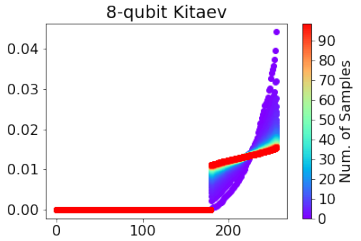
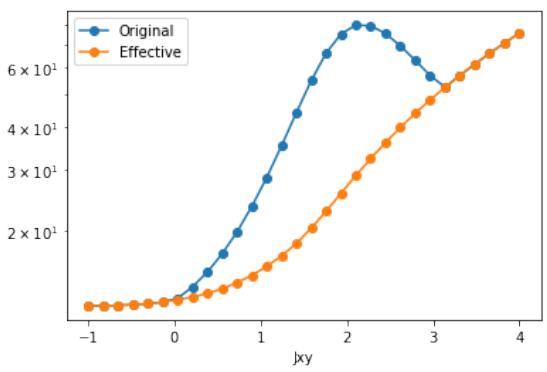
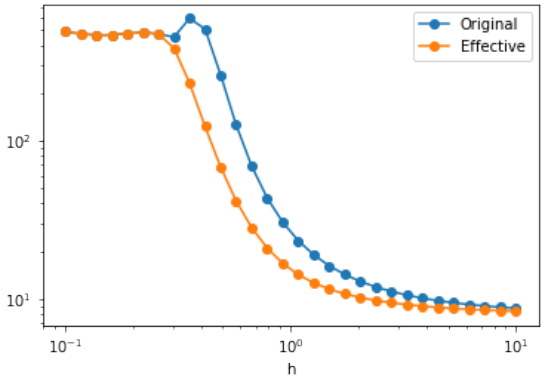


Figure 5.2: Spectrum of $\hat{\mathbf{\Pi}}$ for 8-qubit Kitaev model with 8 qubits for number of samples $R = 1, 2, \cdots, 100$. As the number of samples increases (the color changing from blue to red), $\hat{\mathbf{\Pi}}$ converges to a Hermitian with uniform spectrum, and can thus be good approximation of the normalized projection to the invariant subspace V.



(a) Kitaev Model: N=8, varying J_{xy}



(b) Kitaev Model: N=8, varying h

Figure 5.3: The spectral ratio κ_{eff} for 8-qubit Kitaev models by varying J_{xy} while fixing the external field $h = 1$ and varying h while fixing $J_{xy} = 1$. The effective ratio is significantly smaller than the actual ratio for a wide range of (J_{xy}, h).

5.1.2 Anastz Performance Evaluation

The effective quantities d_{eff} and κ_{eff} serve as a guideline for designing and comparing ansatz designs. We demonstrate Corollary 3.7 by explaining the performances of different ansatz for Ising models and Heisenberg models. Specifically, (1) we calculate d_{eff} and κ_{eff} using the procedure described in Subsection 5.1.1 and (2) directly estimate the over-parameterization thresholds by repetitive training over random initializations with different number of parameters. The results are summarized as follows:

- For transverse field Ising (TFI) model, we compare ansatz TFI_{2alt} and TFI_{3alt} (defined below). TFI_{2alt} and TFI_{3alt} have identical κ_{eff}, but TFI_{2alt} has smaller d_{eff}. Empirically, we observe TFI_{2alt} reaches over-parameterization with fewer number of parameters.

- For the Heisenberg XXZ model, we compare ansatz XXZ_{4alt} and XXZ_{6alt} (defined below). XXZ_{4alt} and XXZ_{6alt} have d_{eff} of same order of magnitude, but the κ_{eff} of XXZ_{6alt} diverges at the critical point while κ_{eff} of XXZ_{4alt} remain bounded. Empirically, we observe that as the system approaches the level-crossing point, XXZ_{6alt} requires significantly more number of parameters to obtain a good approximation to the ground state.

- For both TFI and XXZ models and all HVA considered, d_{eff} is much smaller than the system dimension d. Also for TFI_{2alt}, TFI_{3alt}, XXZ_{4alt}, the effective ratio κ_{eff} remain bounded near level-crossings where $\kappa = \frac{\lambda_d - \lambda_1}{\lambda_2 - \lambda_1}$ approaches infinity. This explains why problem-specifc HVA can be used to solve VQEs that can not be efficiently solved by general-purposed ansatz like HEA ([62]) (Recall that for typical HEA design, d_{eff} is the system dimension d and κ_{eff} is simply κ).

These observations demonstrate the predicting power of the quantities d_{eff} and κ_{eff} and highlight that problem-specific ansatz designs are crucial to the efficient training of VQE in practice.

Transverse Field Ising (TFI) Models. We start by considering the one-dimensional transverse field Ising models (TFI1d). The N-qubit problem Hamiltonian is defined as

$$\mathbf{M}_{\mathsf{TFI1d}}(g) = \sum_{i=0}^{N-1} X_i X_{i+1} + g \sum_{i=0}^{N-1} Z_i \tag{5.5}$$

with periodic boundary conditions (i.e the N-th qubit is identified with the 0-th qubit). The parameter g is the strength of the transverse field. We choose the input state $\frac{1}{\sqrt{2^N}}(1, 1, \cdots, 1)^T$ and a compact HVA for TFI1d model proposed in [62] with $K = 2$ and

$$\mathbf{H}^{(1)} \propto \sum_{i=0}^{N-1} X_i X_{i+1}, \quad \mathbf{H}^{(2)} \propto \sum_{i=0}^{N-1} Z_i. \tag{5.6}$$

In addition to the HVA with 2-alternating Hermitian mentioned in Equation 5.6 (which we will now refer to as TFI_{2alt}), we consider the ansatz design TFI_{3alt} that contains 3 Hermitians $\mathcal{A} = \{\mathbf{H}^{(1)}, \mathbf{H}^{(2)}, \mathbf{H}^{(3)}\}$ with

$$\mathbf{H}^{(1)} \propto \sum_{\mathsf{even}\, i} X_i X_{i+1}, \quad \mathbf{H}^{(2)} \propto \sum_{\mathsf{odd}\, i} X_i X_{i+1}, \quad \mathbf{H}^{(3)} \propto \sum_{i=0}^{N-1} Z_i. \tag{5.7}$$

For all the experiments, $\{\mathbf{H}^{(k)}\}_{k=1}^{K}$ are normalized such that $Z(\mathbf{H}^{(k)}, d) = \mathrm{tr}({\mathbf{H}^{(k)}}^2)/(d^2 - 1) = 1$.

Compared with TFI_{2alt}, TFI_{3alt} decouples the odd and even coupling in the X direcion. The effective dimension d_{eff} of TFI_{2alt} and TFI_{3alt} for $N = 4, 6, 8, 10$ are summarized in Table 5.1: both ansatz designs achieve small effective dimension compared with the system

dimension d, and the effective dimension d_{eff} for TFI_{2alt} is consistently smaller than that of TFI_{3alt} for different N's.

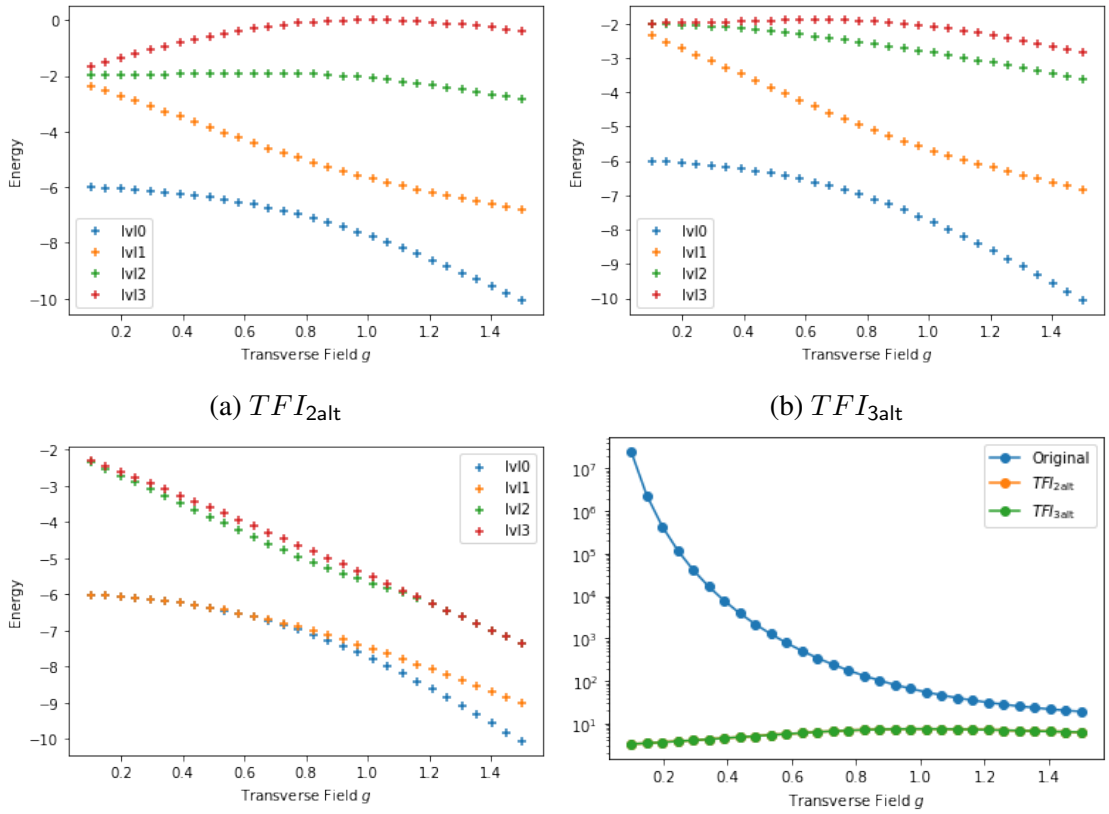
N	4	6	8	10
d	16	64	256	1024
TFI_{2alt}	4	8	16	32
TFI_{3alt}	5	10	25	50

Table 5.1: System dimensions d for N-qubit TFI1d models with $N = 4, 6, 8, 10$ and corresponding effective dimensions d_{eff} for ansatz TFI_{2alt} and TFI_{3alt}.

Despite the difference in d_{eff}, TFI_{2alt} and TFI_{3alt} has similar κ_{eff}: in Figure 5.4, we visualize the eigenvalues and κ_{eff} of TFI_{2alt}, TFI_{3alt} and of the original problem Hamiltonian $\mathbf{M}_{\text{TFI1d}}(g)$ with varying transverse field g for 6-qubit TFI1d models. In Figure 5.4 (a) and (b), we plot the 4 smallest eigenvalues of the effective Hamiltonian $\mathbf{M}'$ associated with TFI_{2alt} and TFI_{3alt}: while TFI_{2alt} and TFI_{3alt} have different effective dimensions, they have similar eigenvalues.

This allows us to demonstrate the dependency of the threshold on the d_{eff} with controlled κ_{eff}. In Figure 5.5 we plot the success rate against the number of parameters p for both ansatz with number of qubits $N = 4, 6, 8, 10$: it is observed that TFI_{2alt} (▼) consistently achieve lower over-parameterization threshold p than TFI_{3alt} (■) due to smaller d_{eff}.

Ground states of TFI1d models are degenerated for $|g| \leq 1$ in the thermodynamic limit $N \to \infty$. Although there are no degeneracy for finite N, the first excitation energy (i.e. the smallest eigen-gaps) decrease quickly as g drops below 1.0. In Figure 5.4(c), we visualize the smallest 4 eigenvalues for $N = 6$. The vanishing eigen-gap for small g leads to drastic increase of κ_{eff} as plotted in blue in Figure 5.4. On the contrary, the effective ratio κ_{eff} for both TFI_{2alt} and TFI_{3alt} remain small as g approaches 0. As a result, the over-parameterization threshold



(a) TFI_{2alt}

(b) TFI_{3alt}

(c) Original

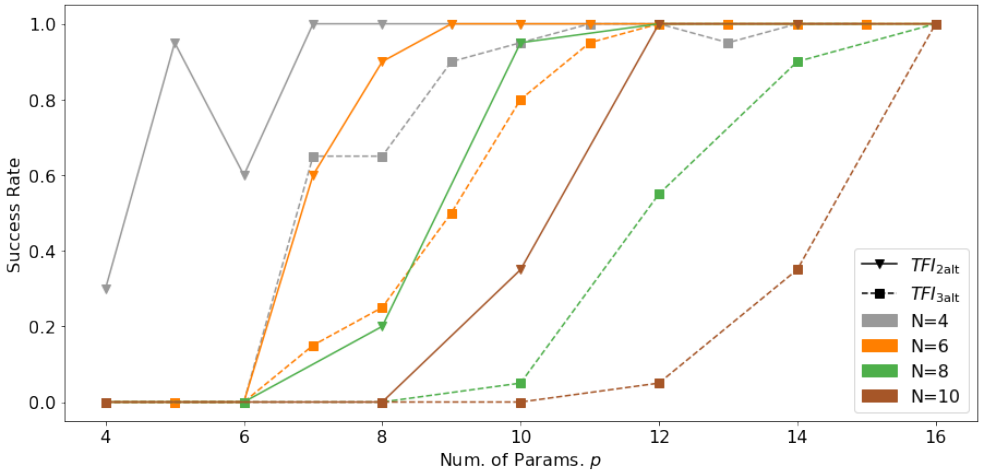
(d) Comparison of κ_{eff}

Figure 5.4: Energy of the ground state and the first 3 excitation states. The smallest 4 eigenvalues for the effective Hamiltonian with TFI_{2alt} (a), TFI_{3alt} (b) and for the original Hamiltonian $H_{\text{TFI1d}}(g)$ (c) for $N = 6$ with transverse field g varying from 0.1 to 1.5. As plotted in (d) κ_{eff} for the original Hamiltonian increases quickly for g close to 0 while κ_{eff} for both TFI_{2alt} and TFI_{3alt} remain small. Note that κ_{eff} for TFI_{2alt} and TFI_{3alt} are overlapping.

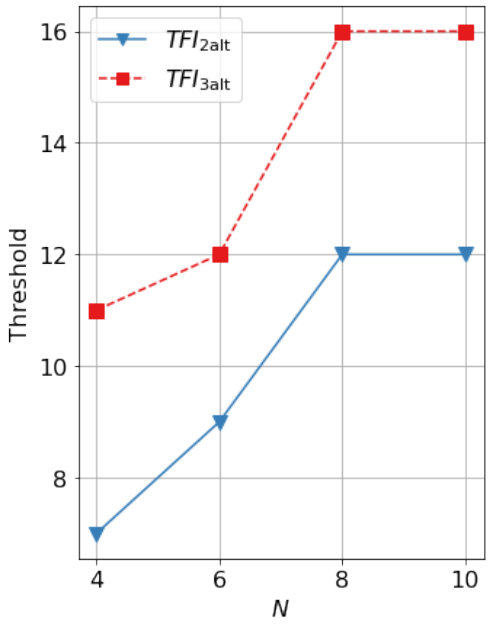
remains almost the same for TFI_{2alt} as the transverse field g decreases from 0.5 to 0.1 (as shown in Figure 5.6). This shows that the usage of HVA instead of general purpose ansatz design allows solving VQE problems efficiently near critical points.

Heisenberg XXZ Models. The one-dimensional XXZ (XXZ1d) model is a special case of Heisenberg model with problem Hamiltonian defined as

$$\mathrm{M}_{\text{XXZ1d}}(J_{zz}) = \sum_{i=0}^{N-1} X_i X_{i+1} + Y_i Y_{i+1} + J_{zz} \sum_{i=0}^{N-1} Z_i Z_{i+1}. \tag{5.8}$$

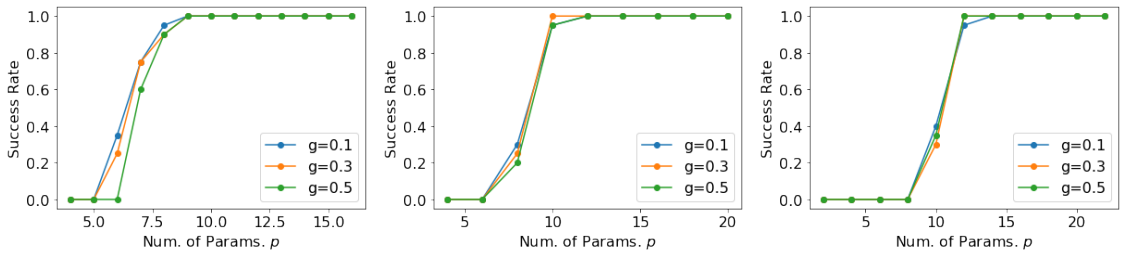


(a)



(b)

Figure 5.5: Comparison of the over-parameterization threshold for TFI_{2alt} and TFI_{3alt} ansatz for $N = 4, 6, 8, 10$. (a) The success rates for finding a solution with error less than 0.01 versus the number of parameters for instances with different ansatz and different sizes. The number of qubits is encoded by different colors and the ansatz design is encoded by ▼ for TFI_{2alt} and ■ for TFI_{3alt}. For each data point, the success rate is estimated over 20 random initializations. (b) Plot of the over-parameterization threshold versus number of qubits for different ansatz. The threshold is defined as the smallest number of parameters to achieve success rate over 98%. For each N, the threshold for TFI_{2alt} is lower than that of TFI_{3alt}.



(a) N=6 (b) N=8 (c) N=10

Figure 5.6: Comparison of the over-parameterization threshold for TFI_{2alt} with transverse field $g = 0.1, 0.3, 0.5$ for (a) $N = 6$ (b)$N = 8$ (c) $N = 10$. The x-axis is the number of trainable parameters p, and the y-axis is the success rate for finding a solution with error less than 0.01. For $N = 6, 8, 10$, despite the vanishing eigen-gap of $H_{TFI1d}(g)$ for small g, the ground state can be found with reasonable p with ansatz TFI_{2alt}. For each data point, the success rate is estimated over 20 random initializations.

The parameter J_{zz} controls the coupling in the Z-direction. XXZ1d model is essentially different from the TFI1d model in that an actual level-crossing happens for finite N at $J_{zz} = -1$.

We examine the ansatz design proposed in [62] (denoted as XXZ_{4alt}):

$$\mathbf{H}^{(1)} \propto \sum_{\text{even}i} X_i X_{i+1} + \sum_{\text{even}i} Y_i Y_{i+1}, \tag{5.9}$$

$$\mathbf{H}^{(2)} \propto \sum_{\text{odd}i} X_i X_{i+1} + \sum_{\text{odd}i} Y_i Y_{i+1}, \tag{5.10}$$

$$\mathbf{H}^{(3)} \propto \sum_{\text{even}i} Z_i Z_{i+1}, \quad \mathbf{H}^{(4)} \propto \sum_{\text{odd}i} Z_i Z_{i+1} \tag{5.11}$$

as well as a similar design (denoted as XXZ_{6alt})

$$\mathbf{H}^{(1)} \propto \sum_{\text{even}i} X_i X_{i+1}, \quad \mathbf{H}^{(2)} \propto \sum_{\text{odd}i} X_i X_{i+1}, \quad \mathbf{H}^{(3)} \propto \sum_{\text{even}i} Y_i Y_{i+1}, \tag{5.12}$$

$$\mathbf{H}^{(4)} \propto \sum_{\text{odd}i} Y_i Y_{i+1}, \quad \mathbf{H}^{(5)} \propto \sum_{\text{even}i} Z_i Z_{i+1}, \quad \mathbf{H}^{(6)} \propto \sum_{\text{odd}i} Z_i Z_{i+1}. \tag{5.13}$$

The effective dimensions for XXZ_{4alt} and XXZ_{6alt} are summarized in Table 5.2. While both

XXZ_{4alt} and XXZ_{6alt} significantly reduce the effective dimension d_{eff}, XXZ_{4alt} further removes the level-crossing: in Figure 5.7, we see that both XXZ_{4alt} and XXZ_{6alt} reduces the ratio κ_{eff} by orders of magnitude, and the ratio κ_{eff} for XXZ_{4alt} (in orange) remains small as $J_{zz} \to -1$ while the ratio for both XXZ_{6alt} (in green) and the original Hamiltonian (in blue) increases to infinity. In Figure 5.8, we present side-by-side the success rates of XXZ_{4alt} and XXZ_{6alt} for $N = 4$ with $J_{zz} = -0.9, -0.5, -0.3, 0.1$. It is observed that the over-parameterization threshold XXZ_{4alt} remain similar across different values of J_{zz} and the over-parameterization thresholds for XXZ_{6alt} increase significantly as J_{zz} decreases to -0.9 due to the vanishing eigen-gaps.

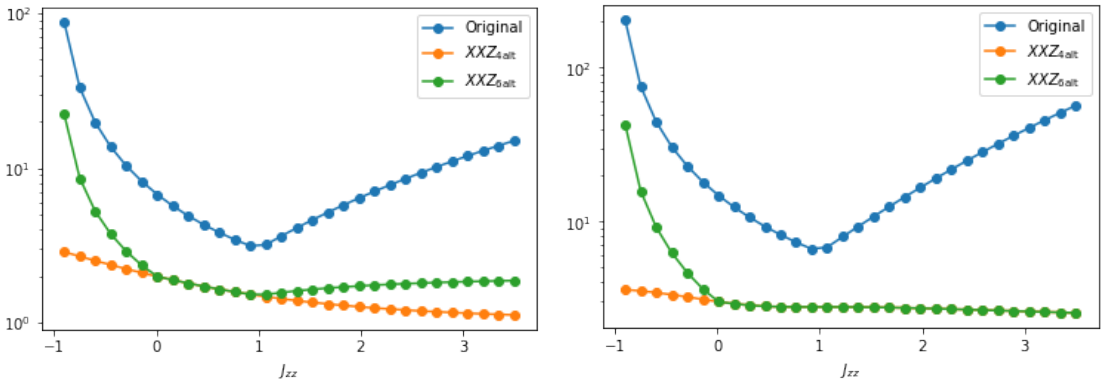
N	4	6	8	10
d	16	64	256	1024
XXZ_{4alt}	3	4	12	21
XXZ_{6alt}	4	5	19	34

Table 5.2: System dimensions d and effective dimensions d_{eff} for XXZ1d model with $N = 4, 6, 8, 10$ for XXZ_{4alt} and XXZ_{6alt}.

5.2 VQE compression

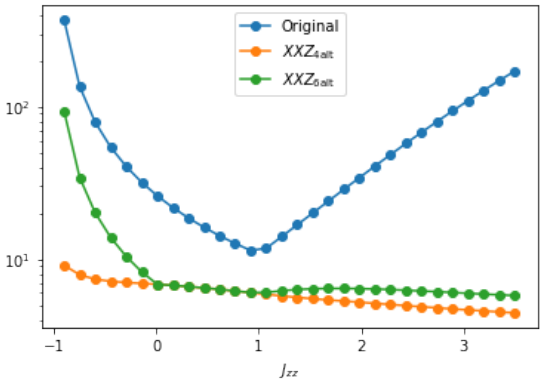
A practical implication of the characterizations of the *trainability* threshold $p^\star$ and *expressivity* threshold $p_\star$ is the compression of VQE. A major application of VQE is to probe the properties of the ground state of a given Hamiltonian by repeatedly preparing the ground state via a variational circuit and making measurements. A shallower variational ansatz for the ground state preparation means lower cost on the quantum resources.

As implicated by our main theorem, problem Hamiltonian $\mathbf{M}$ with larger spectral ratio κ means more number of parameters in an ansatz for efficient training, suggesting that the ground states are harder and more expensive to prepare near critical points. Here we present a procedure

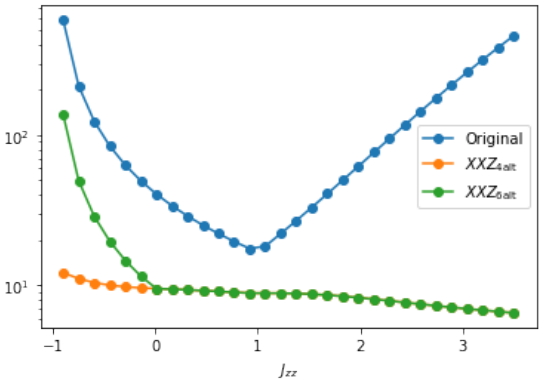


(a) XXZ1d Model: N=4

(b) XXZ1d Model: N=6

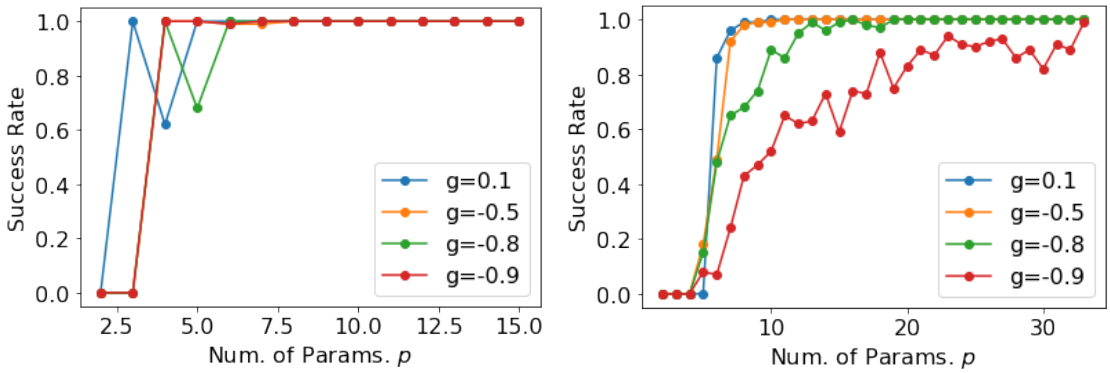


(c) XXZ1d Model: N=8



(d) XXZ1d Model: N=10

Figure 5.7: Spectral ratios κ and κ_{eff} for XXZ_{4alt} and XXZ_{6alt} for $N = 4, 6, 8, 10$. We plot κ for XXZ1d model and κ_{eff} for XXZ_{4alt} and XXZ_{6alt} for different values of J_{zz}. For both XXZ_{6alt} and the original problem Hamiltonian, level crossing happens at $J_{zz} = -1$, making it impossible to solve for the ground state when J_{zz} is close to -1. Note that the level crossing breaks down under XXZ_{4alt}.



(a) XXZ_{4alt} (b) XXZ_{6alt}

Figure 5.8: Comparison of the over-parameterization threshold for (a) XXZ_{4alt} and (b) XXZ_{6alt} with Z-coupling $J_{zz} = 0.1, -0.3, -0.5, -0.9$. The x-axis is the number of trainable parameters p, and the y-axis is the success rate for finding a solution with error less than 0.01. For XXZ_{4alt} the over-parameterization threshold remain similar for various J_{zz}, while for XXZ_{6alt} the threshold drastically increases as J_{zz} approaches -1 as a result of the level-crossing. For each data point, the success rate is estimated over 100 random initializations.

for compressing the state preparation circuit such that the test-time cost is independent of κ.

Two-stage Compression. Consider a $d \times d$ problem Hamiltonian $\mathbf{M}$ with spectral ratio κ and an ansatz design $\mathcal{A}$. Without of loss of generality, we assume $\mathcal{A}$ mixes to the whole d-dimensional special unitary group. Let $\mathbf{U}^o(\boldsymbol{\theta}^o)$ be the variational ansatz found by solving $\min_{\boldsymbol{\theta}} \langle\Phi|(\mathbf{U}^o(\boldsymbol{\theta}))^\dagger \mathbf{M} \mathbf{U}^o(\boldsymbol{\theta})|\Phi\rangle$ with gradient descent. Theorem 3.3 suggests $U^o(\cdot)$ has $O(\mathsf{poly}(d, \kappa))$ parameters to ensure $|\Psi^o\rangle = \mathbf{U}^o(\theta^o)|\Phi\rangle$ approximates the ground state with infidelity $\leq \epsilon$. The *original* ansatz $\mathbf{U}^o(\cdot)$ can be further compressed by solving a second VQE problem $\min_{\boldsymbol{\theta}} -\langle\Phi|\mathbf{U}^c(\boldsymbol{\theta})^\dagger|\Psi^o\rangle\langle\Psi^o|\mathbf{U}^c(\boldsymbol{\theta})|\Phi\rangle$. The *compressed* ansatz $\mathbf{U}^c(\cdot)$ with $O(\mathsf{poly}(d))$ parameters is sufficient to ensure a solution $|\Psi^c\rangle := \mathbf{U}^c(\theta^c)|\Psi\rangle$ be a ϵ-approximation to $|\Psi^o\rangle$ and therefore a 2ϵ-approximation to $|\Psi^\star\rangle$.

The compression stage can be implemented as

$$\min_{\boldsymbol{\theta}} -\langle\Phi|(\mathbf{U}^c(\boldsymbol{\theta}))^\dagger \mathbf{U}^o(\boldsymbol{\theta}^o)|\Phi\rangle\langle\Phi|(\mathbf{U}^o(\boldsymbol{\theta}^o))^\dagger \mathbf{U}^c(\boldsymbol{\theta})|\Phi\rangle \tag{5.14}$$

using swap tests. While the compression procedure does not reduce the resources for training, it compresses the variational ansatz for preparing the ground state during the test time. This procedure can significantly reduce the resource cost for $\mathbf{M}$ with vanishing first excitation energy.

5.3 VQE Preconditioning

Inspired by the κ-dependency on *trainability* threshold, it is natural to consider the idea of the *preconditioning* technique from the scientific computing literature to reduce κ. Preconditioning utilizes a transformed problem Hamiltonian $\overline{\mathbf{M}}$ that shares the ground state with the original $\bar{\mathbf{M}}$ but with reduced spectral ratio κ. Here as a proof of concept, we consider the polynomial preconditioning and showcase it on the Kitaev model on a mixed lattice defined in Section 5.1.1.

The polynomial preconditioning is defined as follows:

Definition 5.1 (Polynomial Preconditioning)**.** Let k be a positive integer, and a be a real number in $[0, \infty)$. The (k, a)-polynomial preconditioning of a problem Hamiltonian $\mathbf{M}$ results in

$$\overline{\mathbf{M}}_{k,a} := -(\lambda \mathbf{I} - \mathbf{M})^k \tag{5.15}$$

where λ is an upper estimate of the largest eigenvalue of $\mathbf{M}$. The real number a measures the accuracy of the estimation λ, such that $a := \frac{\lambda - \lambda_d}{\lambda_d - \lambda_1}$. Here $\lambda_1 < \lambda_2 \leq \cdots \leq \lambda_d$ are eigenvalues of $\mathbf{M}$ in an ascending order.

For any integer k and $a \geq 0$, the polynomial preconditioning preserves the order of eigenvectors:

Lemma 5.1 (Soundness of Polynomial Preconditioning)**.** *For a Hermitian* $\mathbf{M}$ *with the eigen-*

decomposition $\mathbf{M} = \sum_{j=1}^{d} \lambda_j \mathbf{v}_j \mathbf{v}_j^\dagger$ *such that* $\lambda_1 \leq \cdots \leq \lambda_j \leq \cdots \leq \lambda_d$*, the* (k, a)*-polynomial-preconditioned Hermitian in Definition 5.1 has an eigen-decomposition* $\overline{\mathbf{M}}_{k,a} = \sum_{j=1}^{d} \lambda'_j \mathbf{v}_j \mathbf{v}_j^\dagger$ *with* $\lambda'_1 \leq \cdots \leq \lambda'_j \leq \cdots \leq \lambda'_d$.

The proof of the soundness lemma follows directly from the fact that the polynomial function $f_k := -(\lambda - x)^k$ is monotonically increasing for $x < \lambda$. Additionally, the polynomial function amplifies the smaller eigen-gaps, leading to a much smaller spectral ratio κ:

$$\overline{\kappa}_{\lambda,k} = \frac{(\lambda - \lambda_1)^k - (\lambda - \lambda_d)^k}{(\lambda - \lambda_1)^k - (\lambda - \lambda_2)^k} \tag{5.16}$$

$$= \frac{1 - (1 - \frac{1}{1+a})^k}{1 - (1 - \frac{1}{\kappa(1+a)})^k}. \tag{5.17}$$

Specifically, for sufficiently accurate estimation of λ_d (i.e. $a \to 0$) and sufficiently large original κ, the preconditioning procedure results in $\kappa(\overline{\mathbf{M}}_{k,a}) \approx \kappa(\mathbf{M})/k$. See Figure 5.9 for an illustration. The statements above hold also for the effective quantities when the ansatze are restricted and

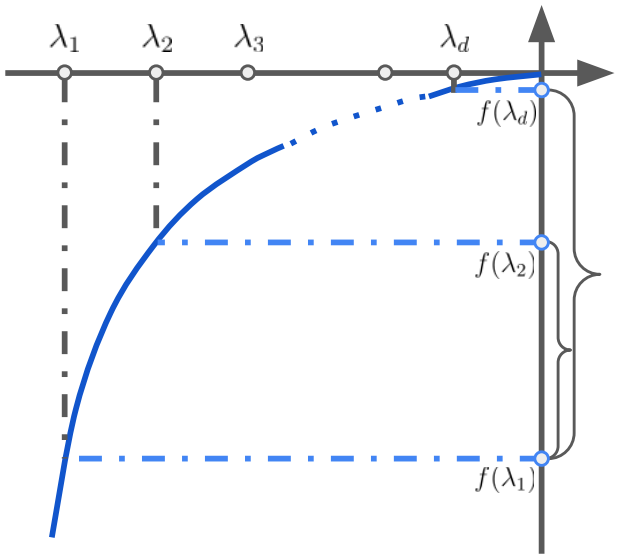


Figure 5.9: A schematic diagram of the polynomial transformation $f_k := -(\lambda - x)^k$. Note that (1) f_k preserves the order of eigen-values and (2) amplifies the first excitation energy.

traverses a proper subgroup of the $SU(d)$.

To showcase the power of polynomial preconditioning, we simulate the VQE training of a 8-qubit Kitaev model defined in Section 5.1.1 with $J_{xy} = 0.7$ and $h = 0.03$. In Figure 5.10, the barplot is the success rate for training a 6-parameter (i.e. one-layer) Hamiltonian variational ansatz as defined in Section 5.1.1 preconditioned with varying polynomial degrees k (along the x-axis) and estimation accuracies $a = 0\%$ and 10% (color-coded). As a reference, we plot the success rates for training ansatze with number of parameters $p = 6, 12, 18, 64$ (i.e. number of layers $L = 1, 2, 3, 4$) for an unpreconditioned problem Hamiltonian as horizontal dotted lines. Observe that the preconditioning allows the training of a 6-parameter ansatz to achieve a success rate larger than that of a 24-parameter ansatz. Also we see that the preconditioning technique is insensitive to the estimation accuracy of λ.

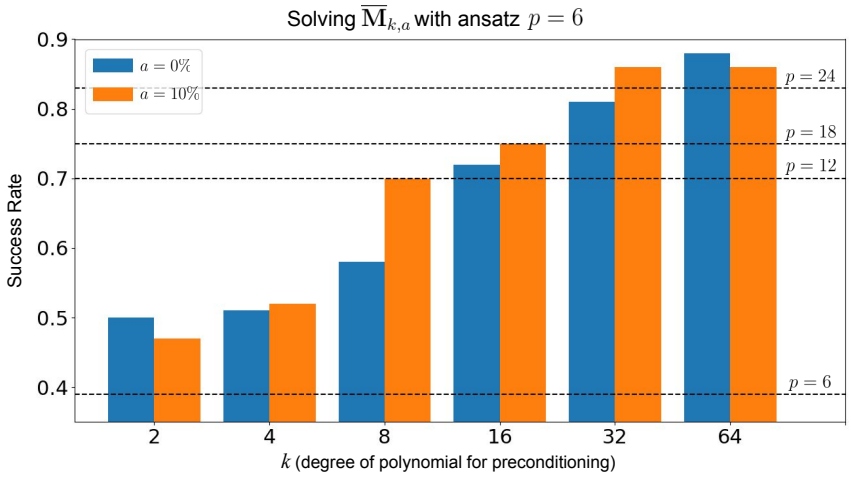


Figure 5.10: Effect of preconditioning in a 8-qubit Kitaev model on the mixed lattice with XY-coupling $J_{xy} = 0.7$ and external magnetic field $h = 0.03$. **Barplots**: the success rates for optimizing a 6-parameter ansatz with varying pairs of (k, a); **Horizontal Lines**: the success rates for optimizing ansatze with $6, 12, 18, 24$ parameters without preconditioning. (1) Comparing the barplots with the horizontal lines, a 6-parameter ansatz with preconditioned problem Hamiltonian outperforms a 24-parameter ansatz without preconditioning in terms of the success rate; (2) Comparing the bars in orange and blue, the effect of preconditioning is insensitive to the choice of a.

Chapter 6: Quantum Exploration Algorithms for Multi-Armed Bandits

In this chapter, we move beyond the scope of variational quantum algorithms and consideer the optimization problems in general quantum algorithms.

Identifying the best arm of a multi-armed bandit is a central problem in bandit optimization. We study a quantum computational version of this problem with coherent oracle access to states encoding the reward probabilities of each arm as quantum amplitudes. Specifically, we show that we can find the best arm with fixed confidence using $\tilde{O}\big(\sqrt{\sum_{i=2}^{n}\Delta_i^{-2}}\big)$ quantum queries, where Δ_i represents the difference between the mean reward of the best arm and the i^{th}-best arm. This algorithm, based on variable-time amplitude amplification and estimation, gives a quadratic speedup compared to the best possible classical result. We also prove a matching quantum lower bound (up to poly-logarithmic factors).

6.1 Introduction

The multi-armed bandit (MAB) model is one of the most fundamental settings in reinforcement learning. This simple scenario captures crucial issues such as the tradeoff between exploration and exploitation. Furthermore, it has wide applications to areas including operations research, mechanism design, and statistics.

A basic challenge about multi-armed bandits is the problem of *best-arm identification*,

where the goal is to efficiently identify the arm with the largest expected reward. This problem captures a common difficulty in practical scenarios, where at unit cost, only partial information about the system of interest can be obtained. A real-world example is a recommendation system, where the goal is to find appealing items for users. For each recommendation, only feedback on the recommended item is obtained. In the context of machine learning, best-arm identification can be viewed as a high-level abstraction and core component of active learning, where the goal is to minimize the uncertainty of an underlying concept, and each step only reveals the label of the data point being queried.

Quantum computing is a promising technology with potential applications to diverse areas including cryptanalysis, optimization, and simulation of quantum physics. Quantum computing devices have recently been demonstrated to experimentally outperform classical computers on a specific sampling task [5]. While noise limits the current practical usefulness of quantum computers, they can in principle be made fault tolerant and thus capable of executing a wide variety of algorithms. It is therefore of significant interest to understand quantum algorithms from a theoretical perspective to anticipate future applications. In particular, there has been increasing interest in *quantum machine learning* (see for example the surveys by [92, 93, 94, 95]). In this paper, we study best-arm identification in multi-armed bandits, establishing quantum speedup.

Problem Setup. We work in a standard multi-armed bandit setting [96] in which the MAB has n arms, where arm $i \in [n] \coloneqq \{1, \ldots, n\}$ is a Bernoulli random variable taking value 1 with probability p_i and value 0 with probability $1 - p_i$. Each arm can therefore be regarded as a coin with *bias* p_i. As our algorithms and lower bounds are symmetric with respect to the arms, we assume without loss of generality that $p_1 \geq \cdots \geq p_n$, and denote $\Delta_i \coloneqq p_1 - p_i$ for all $i \in \{2, \ldots, n\}$. We further assume that $p_1 > p_2$, i.e., the best arm is unique. Given a parameter

$\delta \in (0,1)$, our goal is to use as few queries as possible to determine the best arm with probability $\geq 1-\delta$. This is known as the *fixed-confidence setting*. We primarily characterize complexity in terms of the parameter

$$H := \sum_{i=2}^{n} \frac{1}{\Delta_i^2} \tag{6.1}$$

which arises in the analysis of classical MAB algorithms (as discussed below).

We consider a quantum version of best-arm identification in which we can access the arms *coherently*. This means we have access to a quantum oracle $\mathcal{O}$ that acts as

$$\begin{aligned} \mathcal{O}\colon |i\rangle_I |0\rangle_B |0\rangle_J \\ \mapsto |i\rangle_I (\sqrt{p_i}|1\rangle_B |v_i\rangle_J + \sqrt{1-p_i}|0\rangle_B |u_i\rangle_J), \end{aligned} \tag{6.2}$$

where $|v_i\rangle$ and $|u_i\rangle$ are arbitrary states, for all $i \in [n]$. We have used standard Dirac notation which we review in the Preliminaries section. Register I is the "index" register with n states that correspond to the n arms. Register B is the single-qubit "bandit" register with two states, $|1\rangle$ corresponding to a reward and $|0\rangle$ corresponding to no reward. Register J is a multi-qubit "junk" register. For convenience, we omit register labels when this causes no confusion. Compared to pulling an arm classically—which can be implemented by measuring the bandit register—the quantum oracle allows access to different arms in superposition, a necessary feature for quantum speedup. In real-world applications, we usually have junk when instantiating our oracle (see below). When deriving our results however, we will assume there is no junk (i.e., we set $|v_i\rangle = |u_i\rangle = 1$ for all $i \in [n]$ in (6.2)). This is without loss of generality as the algorithm we construct is insensitive to junk.

Previous work on quantum algorithms for clustering [97, 98] and reinforcement learning [95,

99] has discussed how to instantiate $\mathcal{O}$. In clustering, $\mathcal{O}$ is created using the SWAP test where for each i, p_i encodes the distance between some fixed vector and the i^{th} vector in some collection. Our algorithm can be used to speed up the algorithms of [97] and [98]. In reinforcement learning, $\mathcal{O}$ naturally appears in stochastic agent environments; for instance, $\mathcal{O}$ can be viewed as a special case of the oracle in [99] for a Markov decision problem (MDP) of epoch length 1 and state set $\{0, 1\}$, where the goal of the agent is to reach the state 1.

As a concrete example, consider a classical Monte Carlo strategy[1]: at a given position, evaluate the quality of a next move i by uniformly randomly playing out games $x \in X(i)$, where $X(i)$ is the set of valid games from move i onwards, and querying a computer program f that computes a bit $f(i, x) \in \{0, 1\}$ indicating if game x is won (1) or lost (0). In the classical case, we obtain one sample of win or loss using one query to f. In the quantum case, we can also instantiate one query to the quantum oracle in Equation (6.2) using just one query to f. To do this, we apply the circuit for f, made reversible in the usual way [100, Sec. 1.4.1], on the quantum state corresponding to uniformly random play as follows:

$$\begin{aligned} &|i\rangle|0\rangle \frac{1}{\sqrt{|X(i)|}} \sum_{x \in X(i)} |x\rangle \\ \overset{f}{\mapsto} &|i\rangle \sum_{x \in X(i)} \frac{1}{\sqrt{|X(i)|}} |f(i,x)\rangle|x\rangle \\ = &|i\rangle(\sqrt{p_i}|1\rangle|u_i\rangle + \sqrt{1-p_i}|0\rangle|v_i\rangle), \end{aligned} \tag{6.3}$$

where $|u_i\rangle$ and $|v_i\rangle$ are some states, and p_i is the empirical probability that move i leads to a win. Our quantum algorithm then uses quadratically fewer calls to f compared with classical Monte Carlo search to find the best next move.

[1] This is Monte Carlo tree search without tree expansion.

We stress that we do not need to know the p_is to instantiate the quantum oracle above. We also remark that our algorithm does not apply to every MAB situation. For example, in clinical trials to identify the best drug, we cannot instantiate the quantum oracle because human participants, unlike computer programs, cannot be queried in superposition.

Our algorithm can also be adapted to work when the reward distributions are promised to have bounded variance (for example, if they are sub-Gaussian). The adaptation essentially follows by replacing amplitude estimation (introduced in the Preliminaries section) with quantum mean estimation [101], which works on any distribution with bounded variance. We remark that the situation is different for the other main type of bandits: adversarial bandits. Studies on adversarial bandits are mainly focused on regret minimization and a quantum analogue first requires a proper notion of regret which we are unsure how to even define.

Contributions. In this paper, we give a comprehensive study of best-arm identification using quantum algorithms. Specifically, we obtain the following main result:

Theorem 6.1. *Given a multi-armed bandit oracle $\mathcal{O}$ and confidence parameter $\delta \in (0,1)$, there exists a quantum algorithm that, with probability $\geq 1-\delta$, outputs the best arm using $\tilde{O}\big(\sqrt{H}\big)$ queries to $\mathcal{O}$. Moreover, this query complexity is optimal up to poly-logarithmic factors in n, δ, and Δ_2.*

This represents a quadratic quantum speedup over what is possible classically. The speedup essentially derives from Grover's search algorithm [3], where a marker oracle is used to approximately "rotate" a uniform initial state to the marked state. One way to understand the quadratic speedup is to observe that each rotation step, making one query to the oracle, increases the amplitude of the marked state by $\Omega(1/\sqrt{n})$. This is possible since quantum computation linearly manipulates

amplitudes, which are square roots of probabilities.

However, to establish Theorem 6.1 we use more sophisticated machinery that extends Grover's algorithm, namely variable-time amplitude amplification (VTAA) [102, 103] and estimation (VTAE) [104]. We apply VTAA and VTAE on a variable-time quantum algorithm $\mathcal{A}$ that we construct. $\mathcal{A}$ outputs a state with labeled "good" and "bad" parts. Using that label, VTAA removes the bad part so that only the good part remains, and VTAE estimates the proportion of the good part. In our application, the good part is eventually the best-arm state.

We emphasize that our quantum algorithm, like classical ones [96, 105, 106, 107, 108], does not require any prior knowledge about the p_is.

Given knowledge of p_1 and p_2, our quantum algorithm is conceptually related to the classical successive elimination (SE) algorithm [96]. Namely, we use that knowledge to help eliminate sub-optimal arms i by checking whether $p_i < (p_1 + p_2)/2$, say. The quantum quadratic speedup arises because we can check this "in superposition" across the different arms. For intuition only, checking in superposition can be thought of as a form of checking in parallel. We stress however that while it does not make sense to compare the parallel (classical) sample complexity of best-arm identification with its usual (classical) sample complexity, it does makes sense to compare the latter with the quantum query complexity. We also stress that the similarity of our quantum algorithm to SE, given knowledge of p_1 and p_2, ends at the conceptual level. Technically, our algorithm makes the SE concept work by first marking all sub-optimal arms and then rotating towards the unmarked best arm in quantum state space via a careful application of VTAA. This has no classical analogue.

It is classically easy to remove any assumed knowledge of p_1 and p_2 because classical samples from a multi-armed bandit contain information about their values. Quantumly however,

we cannot simply ask our quantum multi-armed bandit to supply *classical* samples as that would prevent interference, eliminating any quantum speedup. Therefore, we need to do something conceptually different in the quantum case. We construct another quantum algorithm whose goal is to estimate both p_1 and p_2 to precision $\Theta(\Delta_2)$ using $\tilde{O}(\sqrt{H})$ quantum queries. For a given test point l, VTAE (roughly) gives us the ability to *count* the number of arms i with $p_i > l$, and thus allows us to perform binary search to find p_1 and p_2.

Related Work. Classically, a naive algorithm for best-arm identification is to simply sample each arm the same number of times and output the arm with the best empirical bias [96]. This algorithm has complexity $O(\frac{n}{\Delta_2^2}\log(\frac{n}{\delta}))$ but is sub-optimal for most multi-armed bandit instances. Therefore, classical research on best-arm identification [96, 105, 106, 107, 108] has primarily focused on proving bounds of the form $\tilde{O}(H)$ (recall that $H := \sum_{i=2}^{n}\frac{1}{\Delta_i^2}$), which can be shown to be almost tight for every instance. The first work to provide an algorithm with such complexity is [96], giving $O(H\log(\frac{n}{\delta}) + \sum_{i=2}^{n}\Delta_i^{-2}\log(\Delta_i^{-1}))$. This was further improved to $O\big(H\log(\frac{1}{\delta}) + \sum_{i=2}^{n}\Delta_i^{-2}\log\log(\Delta_i^{-1})\big)$ by [105, 106, 107], which is almost optimal except for the additive term of $\sum_{i=2}^{n}\Delta_i^{-2}\log\log(\Delta_i^{-1})$ [108]. More recent work [109, 110] has focused on bringing down even this additive term by tightening both the upper and lower bounds, leaving behind a gap only of the order $\sum_{i=2}^{n}\Delta_i^{-2}\log\log(\min\{n, \Delta_i^{-1}\})$.

Prior work on quantum machine learning has focused primarily on supervised [111, 112, 113, 114] and unsupervised learning [97, 98, 112, 115]. [116, 117, 118] gave quantum algorithms for general reinforcement learning with provable guarantees, but do not consider the best-arm identification problem. The only directly comparable previous work on quantum algorithms for best-arm identification that we are aware of are [119] and [98].[2] By applying Grover's

[2] [98] is not framed as solving best-arm identification, but is partly concerned with this problem.

algorithm, [119] shows that quantum computers can find the best arm with confidence $p_1/\sum_{i=1}^n p_i$ quadratically faster than classical ones. However, [119] does not show how to find the best arm with a given *fixed* confidence, which is the standard requirement. In fact, there is a relatively simple quantum algorithm, analogous to the naive classical algorithm, that can achieve arbitrary confidence with quadratic speedup in terms of n/Δ_2^2. This algorithm, which appears in Fig. 3 of [98], works by using the quantum minimum finding of [120] on top of quantum amplitude estimation [121]. As in the classical case, we show that this simple quantum algorithm is suboptimal for most multi-armed bandit instances. Specifically, we show that a quantum algorithm can achieve quadratic speedup in terms of the parameter H.

6.2 Preliminaries

Definitions and Notations. Quantum computing is naturally formulated in terms of linear algebra. An n-dimensional *quantum state* is a unit vector in the complex Hilbert space $\mathcal{C}^n$, i.e., $\vec{x} = (x_1, \ldots, x_n)^\top$ such that $\sum_{i=1}^n |x_i|^2 = 1$. Such a column vector $\vec{x}$ is written in *Dirac notation* as $|x\rangle$ and called a "ket". The complex conjugate transpose of $|x\rangle$ is written $\langle x|$ and called a "bra", i.e., $\langle x| \coloneqq \vec{x}^\dagger$. The reason for the names is because the combination of a bra and a ket is a inner product bracket: $\langle x|y\rangle \coloneqq \langle x||y\rangle = \vec{x}^\dagger \vec{y} = \langle x, y\rangle \in \mathbb{C}$.

The *computational basis* of $\mathcal{C}^n$ is the set of vectors $\{\vec{e}_1, \ldots, \vec{e}_n\}$, where $\vec{e}_i = (0, \ldots, 1, \ldots, 0)^\top$ is a one-hot column vector with 1 in the i^{th} coordinate. In Dirac notation, it is common to reserve symbols $|i\rangle \coloneqq \vec{e}_i$ and $\langle i| \coloneqq \vec{e}_i^\dagger = \vec{e}_i^\top$. Then, for example, $|x\rangle = \sum_{i=1}^n x_i |i\rangle$ and $\langle x| = \sum_{i=1}^n x_i^* \langle i|$.

The *tensor product* of quantum states is their Kronecker product: if $|x\rangle \in \mathcal{C}^{n_1}$ and $|y\rangle \in$

$\mathcal{C}^{n_2}$, then

$$|x\rangle|y\rangle \coloneqq |x\rangle \otimes |y\rangle \tag{6.4}$$

$$\coloneqq (x_1y_1, x_1y_2, \ldots, x_{n_1}y_{n_2})^\top \in \mathcal{C}^{n_1} \otimes \mathcal{C}^{n_2}. \tag{6.5}$$

A quantum algorithm is a sequence of unitary matrices, i.e., a linear transformation U such that $U^\dagger = U^{-1}$.

For any $p \in [0, 1]$, we define the *coin state* in $\mathcal{C}^2$ as

$$|\mathsf{coin}\ p\rangle \coloneqq \sqrt{p}|1\rangle + \sqrt{1-p}|0\rangle = (\sqrt{1-p}, \sqrt{p})^\top. \tag{6.6}$$

Measuring $|\mathsf{coin}\ p\rangle$ in the computational basis gives 1 with probability p, hence the name.

Quantum Multi-arm Bandit Oracle. Recall the quantum multi-armed bandit oracle defined in (6.2). The arms are accessed in *superposition* by applying the unitary oracle $\mathcal{O}$ on a state $|x\rangle_I|0\rangle_B$ in the joint register of I and B. This results in the output quantum state

$$\mathcal{O}|x\rangle_I|0\rangle_B = \sum_{i=1}^{n} x_i|i\rangle_I|\mathsf{coin}\ p_i\rangle_B \tag{6.7}$$

(recall that we assume there is no junk). A classical pull of the i-th arm can be simulated by choosing $|x\rangle_I = |i\rangle_I$ with $|i\rangle_I|\mathsf{coin}\ p_i\rangle_B$ as the output, and then measuring register B to observe 1 with probability p_i.

In this paper, we mainly focus on *quantum query complexity*, which is defined as the total number of oracle queries. If we have an efficient quantum algorithm for an explicit computational

problem in the query complexity setting, then if we are given an explicit circuit realizing the black-box transformation, we will have an efficient quantum algorithm for the problem.

Amplitude Amplification and Estimation. Our quantum speed-up can be traced back to *amplitude amplification and estimation* [121]. For a classical randomized algorithm for a search problem that returns a correct solution y with probability p_{succ}, the success probability can be amplified to a constant by $O(1/p_{\text{succ}})$ repetitions. Let $\mathcal{A}$ be a quantum procedure that outputs a quantum state $\sqrt{p_{\text{succ}}}|1\rangle|y\rangle + \sqrt{1-p_{\text{succ}}}|0\rangle|y'\rangle$ for some arbitrary quantum state $|y'\rangle$. Measuring the output state yields the solution y with probability p_{succ} just like a classical randomized algorithm. [121] provided an amplitude amplification procedure that amplifies the amplitude of $|1\rangle|y\rangle$ to a constant with $O(1/\sqrt{p_{\text{succ}}})$ queries to the quantum procedure $\mathcal{A}$. This effectively provides a randomized algorithm with constant success probability with query complexity $O(t/\sqrt{p_{\text{succ}}})$ if $\mathcal{A}$ makes t queries to the oracle. The same speed-up can be achieved for the closely related task of estimating p_{succ} with *amplitude estimation*.

Amplitude amplification and estimation originates from *Grover's search algorithm.* [3]. The formal statements of Grover's algorithm and amplitude amplification and estimation are postponed to the start of the appendix. We refer the interested reader to the book [100] on quantum computing for a detailed introduction to basic definitions (Section 3), Grover's algorithm and amplitude amplification (Section 6), and related topics.

Variable-time Amplitude Amplification and Estimation. *Variable-time amplitude amplification* (VTAA) and *estimation* (VTAE) are procedures that apply on top of so-called variable-time quantum algorithms that may stop at different (variable) time steps with certain probabilities. More precisely, for $t = (t_1, t_2, \cdots, t_m) \in \mathbb{R}^m$ and $w = (w_1, w_2, \cdots, w_m) \in \mathbb{R}^m$, a (t, w)-variable-time algorithm $\mathcal{A}$ is one that can be divided into m steps (i.e., $\mathcal{A} = \mathcal{A}_m \cdots \mathcal{A}_1$) where t_j

is the query complexity of $\mathcal{A}_j \cdots \mathcal{A}_1$ and w_j is the probability of stopping at step j. We have:

Theorem 6.2 (Informal: Variable-time amplitude amplification and estimation–[102, 103, 104])**.** *Given a (t, w)-variable-time quantum algorithm $\mathcal{A} = \mathcal{A}_m \cdots \mathcal{A}_1$ with success probability p_{succ}, there exists a quantum algorithm $\mathcal{A}'$ that uses $O(Q)$ queries to output the solution with probability $\geq \frac{1}{2}$, where*

$$Q \coloneqq t_m \log(t_m) + \frac{t_{\text{avg}}}{\sqrt{p_{\text{succ}}}} \log(t_m). \tag{6.8}$$

with $t_{\text{avg}} \coloneqq \sqrt{\sum_{j=1}^{m} w_j t_j^2}$ being the root-mean-square average query complexity of $\mathcal{A}$.

There also exists a quantum algorithm that uses $O(\frac{Q}{\epsilon} \log^2(t_m) \log\log(\frac{t_m}{\delta}))$ queries to estimate p_{succ} with multiplicative error ϵ with probability $\geq 1 - \delta$.

For comparison, recall that applying amplitude amplification and estimation procedures on general quantum algorithms requires $O(t_m/\sqrt{p_{\text{succ}}})$ queries. See the first section of the appendix for a rigorous definition of variable-time algorithms and formal statements of the query complexities of variable-time amplitude amplification and estimation.

6.3 Fast Quantum Algorithm For Best-arm Identification

In this section, we construct a quantum algorithm for best-arm identification and analyze its performance. Specifically:

Theorem 6.3. *Given a multi-armed bandit oracle $\mathcal{O}$ and confidence parameter $\delta \in (0, 1)$, there exists a quantum algorithm that outputs the best arm with probability $\geq 1 - \delta$ using $\tilde{O}(\sqrt{H})$ queries to $\mathcal{O}$.*

Throughout this section, the oracle $\mathcal{O}$ is fixed, so we may omit explicit reference to it. All

logs have base 2.

There are essentially two steps in our construction. In the first step, we construct two subroutines Amplify and Estimate using VTAA and VTAE, respectively, on a variable-time quantum algorithm $\mathcal{A}$. Roughly speaking, given $l \in [0,1]$, Amplify outputs an arm index i randomly chosen from those i with $p_i > l$ while Estimate counts the number of such is. This means that if we knew the values of p_1 and p_2, we could take l to be $(p_1 + p_2)/2$, then Amplify would output the best arm. But we can use Estimate in a binary search procedure to estimate p_1 and p_2. This is exactly what we do in the second step and so we are done.

We now discuss the construction more precisely. Amplify and Estimate actually use two thresholds l_2, $l_1 \in [0,1]$ with $l_2 < l_1$ instead of a single threshold l. In the first step, we construct a variable-time quantum algorithm denoted $\mathcal{A}$ (Algorithm 1) that is initialized in a uniform superposition state $|u\rangle := \frac{1}{\sqrt{n}} \sum_{i \in [n]} |i\rangle$ (since initially we have no information about which arm is the best). Given an input interval $I = [l_2, l_1]$, $\mathcal{A}$ "flags" arm indices in $S'_{\text{right}} := \{i \in [n] : p_i \geq l_1\}$ with a bit $f = 1$ and those in $S'_{\text{left}} := \{i \in [n] : p_i \leq l_2\}$ with a bit $f = 0$. The flag bit f is written to a separate flag register F, so that the state (approximately) becomes $\frac{1}{\sqrt{n}} \big(\sum_{i \in S'_{\text{right}}} |i\rangle |1\rangle_F + \sum_{i \in S'_{\text{left}}} |i\rangle |0\rangle_F + \sum_{i \in S'_{\text{middle}}} |i\rangle |\psi_i\rangle_F \big)$ for some states $|\psi_i\rangle \in \mathcal{C}^2$, where $S'_{\text{middle}} := [n] - (S'_{\text{left}} \cup S'_{\text{right}}) = \{i \in [n] : l_2 < p_i < l_1\}$. The flag bit f stored in the F register indicates whether VTAA (resp. VTAE), when applied on $\mathcal{A}$, should ($f = 1$) or should not ($f = 0$) amplify (resp. estimate) that part of the state. We then apply VTAA and VTAE on $\mathcal{A}$ to construct Amplify and Estimate, respectively. Amplify produces a uniform superposition of all those is with F register in $|1\rangle$, i.e., it amplifies such is relative to the others. Estimate counts the number of such is. More precisely, Estimate (approximately) counts the number of indices in S'_{right}, as their F register is in $|1\rangle$, plus some (unknown) fraction of indices in S'_{middle} as dictated

by the fraction of $|1\rangle$ in the (unknown) states $|\psi_i\rangle$.

In the second step, we use Estimate as a subroutine in Locate (Algorithm 2) to find a interval $[l_2, l_1]$ such that $p_2 < l_2 < l_1 < p_1$ and that $|l_1 - l_2| \geq \Delta_2/4$. Then, running Amplify with these l_2, l_1 in BestArm (Algorithm 4) gives the state $|1\rangle$ containing the best-arm index because only p_1 is to the right of l_2. Locate is a type of binary search that counts the number of indices in S'_{right} using Estimate. There is a technical difficulty here because Estimate actually counts the number of indices in S'_{right} plus some fraction of indices in S'_{middle}. Trying to fix this by simply setting $l_2 = l_1$, so that $S'_{\text{middle}} = \emptyset$, does not work as it would increase the cost of Estimate. We overcome this difficulty via the Shrink subroutine (Algorithm 3) of Locate, which employs a technique from recent work on quantum ground state preparation [122]. See Figure 6.1 for an illustration of the overall structure of the algorithm.

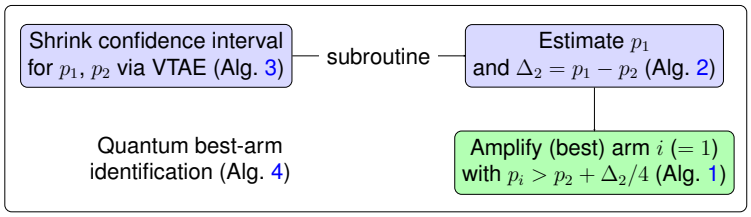


Figure 6.1: Overview of our best-arm identification algorithm.

6.3.1 Amplify and Estimate

We first construct a variable-time quantum algorithm (Algorithm 1) that we call $\mathcal{A}$ throughout. $\mathcal{A}$ uses the following registers: input register I; bandit register B; clock register $C = (C_1, \ldots, C_{m+1})$, where each C_i is a qubit; ancillary amplitude estimation register $P = (P_1, \ldots, P_m)$, where each P_i has $O(m)$ qubits; and flag register F. We set $m := \lceil \log(1/(l_1 - l_2)) \rceil + 2$ as assigned in Algorithm 1.

$\mathcal{A}$ is indeed a variable-time quantum algorithm according to Definition 6.1. This is because we can write $\mathcal{A} = \mathcal{A}_{m+1}\mathcal{A}_m \cdots \mathcal{A}_1\mathcal{A}_0$ as a product of $m+2$ sub-algorithms, where $\mathcal{A}_0$ is the initialization step (Line 4), $\mathcal{A}_j$ consists of the operations in iteration j of the for loop (Lines 6–9) for $j \in [m]$, and $\mathcal{A}_{m+1}$ is the termination step (Lines 10–11). The state spaces $\mathcal{H}_C$ and $\mathcal{H}_A$ in Definition 6.1 correspond to the state spaces of the C register and the remaining registers of $\mathcal{A}$, respectively. $\mathcal{A}_{m+1}$ ensures that Condition 4 of Definition 6.1 is satisfied.

Algorithm 1: $\mathcal{A}(\mathcal{O}, l_2, l_1, \alpha)$

Input: Oracle $\mathcal{O}$ as in (6.2); $0 < l_2 < l_1 < 1$; approximation parameter $0 < \alpha < 1$.

1 $\Delta \leftarrow l_1 - l_2$
2 $m \leftarrow \left\lceil \log \frac{1}{\Delta} \right\rceil + 2$
3 $a \leftarrow \frac{\alpha}{2mn^{3/2}}$
4 Initialize state to $\frac{1}{\sqrt{n}} \sum_{i=1}^{n} |i\rangle_I |\mathsf{coin}\ p_i\rangle_B |0\rangle_C |0\rangle_P |1\rangle_F$
5 **for** $j = 1, \ldots, m$ **do**
6 $\epsilon_j \leftarrow 2^{-j}$
7 **if** *register I is in state $|i\rangle$ and registers $C_1, \ldots, C_{j-1}$ are in state $|0\rangle$* **then**
8 Apply $\mathsf{GAE}(\epsilon_j, a; l_1)$ with $\mathcal{O}_{p_i}$ on registers B, C_j, and P_j
9 Apply controlled-NOT gate with control on register C_j and target on register F
10 **if** *registers $C_1, \ldots, C_m$ are in state $|0\rangle$* **then**
11 Flip the bit stored in register C_{m+1}

With $\Delta := l_1 - l_2$ being the length of $[l_2, l_1]$, we define the following three sets that partition $[n]$:

$$S_{\text{left}} := \{i \in [n] : p_i < l_1 - \Delta/2\}, \tag{6.9}$$

$$S_{\text{middle}} := \{i \in [n] : l_1 - \Delta/2 \le p_i < l_1 - \Delta/8\}, \tag{6.10}$$

$$S_{\text{right}} := \{i \in [n] : p_i \ge l_1 - \Delta/8\}. \tag{6.11}$$

These sets play the roles of aforementioned S'_{left}, S'_{middle}, and S'_{right}. They can be regarded as

functions of (the input to) $\mathcal{A}$. For later convenience, we also define $S_{\mathrm{lm}} \coloneqq S_{\mathrm{left}} \cup S_{\mathrm{middle}}$ and $S_{\mathrm{mr}} \coloneqq S_{\mathrm{middle}} \cup S_{\mathrm{right}}$.

Lemma 6.4 (Correctness of $\mathcal{A}$)**.** *Let p_{succ} denote the success probability $\mathcal{A}$. Then $|p_{\mathrm{succ}} - p'_{\mathrm{succ}}| \leq \frac{2\alpha}{n}$ where $p'_{\mathrm{succ}} = \frac{1}{n}\big(|S_{\mathrm{right}}| + \sum_{i \in S_{\mathrm{middle}}} |\beta_{i,1}|^2\big)$ for some $|\beta_{i,1}|^2 \in [0,1]$.*

At a high level, at iteration j, Line 8 approximately identifies those $i \in S_{\mathrm{left}}$ with $p_i \in [l_1 - 2\epsilon_j, l_1 - \epsilon_j)$ and stops computation on these is by setting their associated C registers to $|1\rangle$. Line 9 then flags these is by setting their associated F registers to $|0\rangle$, indicating failure. We defer the detailed proof to the supplementary material which is mainly concerned with bounding the error in the aforementioned approximation, as well as the lemma as follows.

Lemma 6.5 (Complexity of $\mathcal{A}$)**.** *With $\Delta = l_1 - l_2$ being the length of the interval, we have:*

1. *The j^{th} stopping time t_j of $\mathcal{A}_j\mathcal{A}_{j-1}\cdots\mathcal{A}_0$ is of order $\sum_{k=1}^{j} \frac{1}{\epsilon_k}\log\frac{1}{a} \leq 2^{j+1}\log\frac{1}{a}$. In particular, $t_{m+1} = O(\frac{1}{\Delta}\log\frac{1}{a})$.*

2. *The average stopping time squared, t^2_{avg}, is of order*

$$\frac{1}{n}\left(\frac{|S_{\mathrm{right}}|}{\Delta^2} + \sum_{i \in S_{\mathrm{lm}}} \frac{1}{(l_1 - p_i)^2}\right)\log^2\left(\frac{1}{a}\right). \tag{6.12}$$

Now we fix algorithm $\mathcal{A}$ and its input parameters. We always assume that $|S_{\mathrm{right}}| \geq 1$, which we need for some of the following results to hold. This is without loss of generality as we can always add an artificial arm 0 with bias $p_0 = 1$ to the bandit oracle $\mathcal{O}$, as we do in Line 3 of Algorithm 3.

We apply VTAA and VTAE (Theorem 6.2)[3] on our variable-time quantum algorithm $\mathcal{A}$ to

[3]The state spaces $\mathcal{H}_C$, $\mathcal{H}_F$, and $\mathcal{H}_W$ correspond to the state spaces of the C, F, and remaining registers of $\mathcal{A}$, respectively.

prepare the state $|\psi_{\text{succ}}\rangle$ and to estimate the probability p_{succ}, respectively. This gives two new algorithms Amplify and Estimate with the following performance guarantees.

Lemma 6.6 (Correctness and complexity of $\mathsf{Amplify}(\mathcal{A}, \delta)$, $\mathsf{Estimate}(\mathcal{A}, \epsilon, \delta)$)**.** *Let* $\mathcal{A} = \mathcal{A}(\mathcal{O}, l_2, l_1, 0.01\delta)$. *Then* $\mathsf{Amplify}(\mathcal{A}, \delta)$ *uses* $O(Q)$ *queries to output an index* $i \in S_{\text{mr}}$ *with probability* $\geq 1 - \delta$*, and* $\mathsf{Estimate}(\mathcal{A}, \epsilon, \delta)$ *uses* $O(Q/\epsilon)$ *queries to output an estimate* r *of* p'_{succ} *(defined in Lemma 6.4) such that*

$$(1-\epsilon)\Big(p'_{\text{succ}} - \frac{0.1}{n}\Big) < r < (1+\epsilon)\Big(p'_{\text{succ}} + \frac{0.1}{n}\Big) \tag{6.13}$$

with probability $\geq 1 - \delta$*, where* Q *is*

$$\left(\frac{1}{\Delta^2} + \frac{1}{|S_{\text{right}}|}\sum_{S_{\text{lm}}} \frac{1}{(l_1 - p_i)^2}\right) \mathsf{poly}\Big(\log\Big(\frac{n}{\delta\Delta}\Big)\Big), \tag{6.14}$$

where $\Delta = l_1 - l_2$.

This lemma follows by applying Lemma 6.4 and Lemma 6.5 to Theorem 6.2. The proof detail is given in the appendices.

6.3.2 Quantum algorithm for best-arm identification

In this subsection, we use Amplify and Estimate to construct three algorithms (Algorithms 2–4) that work together to identify the best arm following the outline that we described at the beginning of this section.

We state the correctness and complexities of Amplify and Estimate as follows:

Lemma 6.7 (Correctness and complexity of Algorithm 2)**.** *Fix a confidence parameter* $0 < \delta < 1$. *Then the event* $E = \{p_1 \in I_1 \text{ and } p_2 \in I_2 \text{ in all iterations of the while loop}\}$ *holds with*

Algorithm 2: $\mathsf{Locate}(\mathcal{O}, \delta)$

Input: Oracle $\mathcal{O}$ as in (6.2); confidence parameter $0 < \delta < 1$.

$I_1, I_2 \leftarrow [0, 1]$
$\delta \leftarrow \delta/8$
while $\min I_1 - \max I_2 < 2\,|I_1|$ **do**
    $I_1 \leftarrow \mathsf{Shrink}(\mathcal{O}, 1, I_1, \delta)$
    $I_2 \leftarrow \mathsf{Shrink}(\mathcal{O}, 2, I_2, \delta)$
    $\delta \leftarrow \delta/2$
return I_1, I_2

Algorithm 3: $\mathsf{Shrink}(\mathcal{O}, k, I, \delta)$

Input: Oracle $\mathcal{O}$ as in (6.2); $k \in \{1, 2\}$; interval $I = [a, b]$; confidence parameter $0 < \delta < 1$.

$\epsilon \leftarrow (b - a)/5$
$\delta \leftarrow \delta/2$
Append arm $i = 0$ with bias $p_0 = 1$ to $\mathcal{O}$; call the resulting oracle $\mathcal{O}'$
Construct variable-time quantum algorithms $\mathcal{A}_1, \mathcal{A}_2$:
    $\mathcal{A}_1 \leftarrow \mathcal{A}(\mathcal{O}', l_2 = a + \epsilon, l_1 = a + 3\epsilon, 0.01\delta)$
    $\mathcal{A}_2 \leftarrow \mathcal{A}(\mathcal{O}', l_2 = a + 2\epsilon, l_1 = a + 4\epsilon, 0.01\delta)$
$r_1 \leftarrow \mathsf{Estimate}(\mathcal{A}_1, \epsilon = 0.1, \delta)$
$r_2 \leftarrow \mathsf{Estimate}(\mathcal{A}_2, \epsilon = 0.1, \delta)$
$B_1 \leftarrow \mathbb{1}(r_1 > \frac{k+0.5}{n+1})$; $B_2 \leftarrow \mathbb{1}(r_2 > \frac{k+0.5}{n+1})$
switch (B_1, B_2) **do**
    case $(0, 0) : I \leftarrow [a, a + 3\epsilon]$
    case $(0, 1) : I \leftarrow [a + \epsilon, a + 4\epsilon]$
    case $(1, 0) : I \leftarrow [a + \epsilon, a + 4\epsilon]$
    case $(1, 1) : I \leftarrow [a + 2\epsilon, a + 5\epsilon = b]$
return I

probability $\geq 1 - \delta$. *When* E *holds, Algorithm 2 also satisfies the following for both* $k \in \{1, 2\}$*:*

1. *its while loop (Line 3) breaks at or before the end of iteration* $\left\lceil \log_{5/3}(\frac{1}{\Delta_2}) \right\rceil + 3$ *and then returns* I_k *with* $p_k \in I_k$ *and* $\min I_1 - \max I_2 \geq 2\,|I_1|$*; during the while loop, we always have* $|I_1| = |I_2| \geq \Delta_2/8$*; and*

2. *it uses* $O\big(\sqrt{H}\,\mathsf{poly}()(\log(\frac{n}{\delta\,\Delta_2}))\big)$ *queries.*

Lemma 6.8 (Correctness and complexity of Algorithm 3)**.** *Fix* $k \in \{1, 2\}$*, an interval* $I = [a, b]$*,*

Algorithm 4: $\mathsf{BestArm}(\mathcal{O}, \delta)$

Input: Oracle $\mathcal{O}$ as in (6.2); confidence parameter $0 < \delta < 1$.

1 $\delta \leftarrow \delta/2$
2 $I_1, I_2 \leftarrow \mathsf{Locate}(\mathcal{O}, \delta)$
3 $l_1 \leftarrow \min I_1$ (left endpoint of I_1)
4 $l_2 \leftarrow \max I_2$ (right endpoint of I_2)
5 Construct variable-time quantum algorithm $\mathcal{A}$:
6 $\quad \mathcal{A} \leftarrow \mathcal{A}(\mathcal{O}, l_2, l_1, 0.01\delta)$
7 $i \leftarrow \mathsf{Amplify}(\mathcal{A}, \delta)$
8 **return** i

and a confidence parameter $0 < \delta < 1$. *Suppose that* $p_k \in I$ *and* $|I| \geq \Delta_2/8$. *Then Algorithm 3*

1. *outputs an interval* J *with* $|J| = \frac{3}{5}|I|$ *such that* $p_k \in J$ *with probability* $\geq 1 - \delta$, *and*

2. *uses* $O\big(\sqrt{H}\,\mathsf{poly}\big(\log\big(\frac{n}{\delta\Delta_2}\big)\big)\big)$ *queries.*

The proofs of Lemma 6.7 and Lemma 6.8 appear in the supplementary material.

The following theorem is equivalent to Theorem 6.3.

Theorem 6.9 (Correctness and complexity of Algorithm 4)**.** *Fix a confidence parameter* $0 < \delta < 1$. *Then, with probability* $\geq 1 - \delta$, *Algorithm 4*

1. *outputs the best arm, and*

2. *uses* $O\big(\sqrt{H}\,\mathsf{poly}\big(\log\big(\frac{n}{\delta\Delta_2}\big)\big)\big)$ *queries.*

Proof. Note that δ is halved at the beginning, on Line 1. For the first claim, we know from the first claim of Lemma 6.7 that, with probability $\geq 1 - \delta/2$, the two intervals I_k assigned in Line 2 have $\min I_1 - \max I_2 \geq 2|I_1| \geq \Delta_2/4$ and $p_k \in I_k$. Assuming this holds, we have $p_2 < l_2 < l_2 + \Delta_2/4 \leq l_1 < p_1$ for the endpoints l_k assigned in Lines 3 and 4. This means that the variable-time quantum algorithm $\mathcal{A}$ defined in Line 6 has $S_{\text{right}} \cup S_{\text{middle}} = \{1\}$, so

$\mathsf{Amplify}(\mathcal{A}, \delta/2)$ returns index 1 with probability $\geq 1 - \delta/2$. Therefore, the overall probability of Algorithm 4 returning the best arm is at least $1 - \delta$.

The second claim follows immediately from adding the complexity of $\mathsf{Locate}(\mathcal{O}, \delta/2)$ (Lemma 6.7) and $\mathsf{Amplify}(\mathcal{A}, \delta/2)$ (Lemma 6.6, using $l_1 - l_2 \geq \Delta_2/4$). □

By establishing Theorem 6.9, we have established Theorem 6.3, our main claim. As discussed previously, the main complexity measure of interest in the classical case is H, and we see that we get a quadratic speedup in terms of this parameter.

We can see that the poly-logarithmic factor has degree about 6 from (6.38), (6.40), and (6.42). It would be interesting to reduce this degree. A more fundamental challenge is to remove the variable n that appears in our log factors. In the classical case, n was already removed from log factors in early work [96] by a procedure called "median elimination". However, quantizing the median elimination framework is nontrivial, as the query complexity for outputting the $n/2$ smallest items among n elements is $\Theta(n)$ [123, Theorem 1], exceeding our budget of $O(\sqrt{n})$.

As corollaries of our main results in the fixed-confidence setting, we provide results on best-arm identification in the PAC (Probably Approximately Correct) and fixed-budget settings. In the (ϵ, δ)-PAC setting, the goal is to identify an arm i with $p_i \geq p_1 - \epsilon$ with probability $\geq 1 - \delta$. Our best-arm identification algorithm can be modified to work in this setting as well. More precisely, we can modify Locate (Algorithm 2) by adding a breaking condition to the while loop when $|I_1|$ (or equivalently $|I_2|$) is smaller than ϵ. This gives the following result:

Corollary 6.10. *There is a quantum algorithm that finds an ϵ-optimal arm with query complexity $O\big(\sqrt{\min\{\frac{n}{\epsilon^2}, H\}} \cdot \mathsf{poly}()(\log(\frac{n}{\delta\,\Delta_2}))\big)$.*

Note that our modification means that the Amplify step in Algorithm 4 takes an input

interval I with $|I| = l_1 - l_2 \in [\epsilon/2, \epsilon]$. The correctness and complexity follow directly from Lemma 6.4 and Lemma 6.6. For comparison, [96] gave a classical PAC algorithm with complexity $O\big(\frac{n}{\epsilon^2}\log\big(\frac{n}{\delta}\big)\big)$, which was later improved to $O\big(\sum_{i=1}^{n}\min\{\epsilon^{-2}, \Delta_i^{-2}\}\cdot\log\big(\frac{n}{\delta\Delta_2}\big)\big)$ by [105].

In the supplementary material, we also show how to identify the best arm with high probability for a fixed number of total queries (the fixed-budget setting) given knowledge of H.

6.4 Quantum Lower Bound

In this section, we describe a lower bound for the quantum best-arm identification problem. Our lower bound shows that the algorithm of Theorem 6.3 is optimal up to poly-logarithmic factors.

Theorem 6.11. *Let $p \in (0, 1/2)$. For any biases $p_i \in [p, 1-p]$, any quantum algorithm that identifies the best arm requires $\Omega(\sqrt{H})$ queries to the multi-armed bandit oracle $\mathcal{O}$.*

To prove this lower bound, we use the quantum adversary method to show quantum hardness of distinguishing n oracles $\mathcal{O}_x$, $x \in [n]$, corresponding to the following n bandits. In the 1^{st} bandit, we assign bias p_i to arm i for all i. In the x^{th} bandit for $x \in \{2, \ldots, n\}$, we assign bias $p_1 + \eta$ to arm x and p_i to arm i for all $i \neq x$, where η is an appropriately chosen parameter. This hard set of bandits is inspired by the proof of a corresponding classical lower bound [108, Theorem 5].

More precisely, for a positive integer T, consider an arbitrary T-query quantum algorithm that distinguishes the oracles $\mathcal{O}_x$. The main idea of the adversary method is to keep track of certain quantities $s_k \in \mathbb{R}$ where $k \in \{0, 1, \ldots, T\}$. For each k, s_k quantifies how close the states of the quantum algorithm are when it operates using k queries to the different $\mathcal{O}_x$. At the start,

when $k = 0$, s_0 must be large because when no queries have been made, the states must be close. At the end, when $k = T$, s_T must be small because the states are distinguishable by assumption.

The key point is that we can also bound how much s_k can change in one query, that is we can bound the quantities $|s_{k+1} - s_k|$ for each k. Of course, this bound immediately gives a lower bound on T, the number of queries it takes to go from s_0 (large) to s_T (small). To bound $|s_{k+1} - s_k|$, the key point is to bound the distance between oracles, i.e. matrices, $\mathcal{O}_x$ and $\mathcal{O}_y$ for different $x, y \in [n]$.

We defer the full proof and full description of the quantum adversary method to the supplementary material.

6.5 Conclusions

In this paper, we propose a quantum algorithm for identifying the best arm of a multi-armed bandit, which gives a quadratic speedup compared to the best possible classical result. We also prove a matching quantum lower bound (up to poly-logarithmic factors).

This work leaves several natural open questions:

- Can we give fast quantum algorithms for the exploitation of multi-armed bandits? In particular, can we give online algorithms with favorable regret? The quantum hedging algorithm [124] and the quantum boosting algorithm [125] might be relevant to this challenge.

- Can we give fast quantum algorithms for other types of multi-armed bandits, such as contextual bandits or adversarial bandits (e.g., [126, 127, 128])?

- Can we give fast quantum algorithms for finding a near-optimal policy of a Markov decision process (MDP)? MDPs are a natural generalization of MABs, where the goal is to maximize

the expected reward over sequences of decisions. [96] gave a reduction from this problem to best-arm identification by viewing the Q-function of each state as a multi-armed bandit.

6.6 Appendix: Preliminaries on Quantum Algorithms

6.6.1 Grover's search and amplitude amplification and estimation

Our quantum speedup conceptually originates from *Grover's search algorithm* [3]. Consider a function $f_w\colon [n] \to \{-1, 1\}$ such that $f_w(i) = 1$ if and only if $i \neq w$, so that w can be viewed as a (unique) marked item. To search for w, classically we need $\Omega(n)$ queries to f_w. Quantumly, we can use one call of f_w to create an oracle U_w such that $U_w|i\rangle = |i\rangle$ for all $i \neq w$ and $U_w|w\rangle = -|w\rangle$.Now consider the uniform superposition $|u\rangle \coloneqq \frac{1}{\sqrt{n}} \sum_{i\in[n]} |i\rangle$ as well as the state $|r\rangle \coloneqq \frac{1}{\sqrt{n-1}} \sum_{i\in[n]/\{w\}} |i\rangle$. The angle between $U_w|u\rangle$ and $|u\rangle$ is $\theta \coloneqq \arccos(1/n) = \Theta(1/\sqrt{n})$. Note that the unitary U_w reflects about $|r\rangle$, and the unitary $U_u = 2|u\rangle\langle u| - I$ reflects about $|u\rangle$. If we start with $|u\rangle$, the angle between $U_w|u\rangle$ and $U_uU_w|u\rangle$ is *amplified* to 2θ, and in general the angle between $U_w|u\rangle$ and $(U_uU_w)^k|u\rangle$ is $2k\theta$. It thus suffices to take $k = \Theta(\sqrt{n})$ to find w.

This method of alternatively applying two reflections to boost the amplitude for success can be generalized to a technique called *amplitude amplification*. For the case with some unknown number $k \in [n]$ of marked items, there is also a quadratic quantum speedup for estimating $\theta \coloneqq \arccos(k/n)$ via a technique called *amplitude estimation* [121].

In the context of searching, consider a quantum procedure $\mathcal{A}$ that returns a state $|\psi\rangle$ with t oracle queries, such that the overlap between the target state $|w\rangle$ and output state $|\psi\rangle$ is $p_{\text{succ}} \coloneqq |\langle w|\psi\rangle|^2$. By amplitude amplification and estimation [121], $O(t/\sqrt{p_{\text{succ}}})$ oracle queries suffice to amplify the overlap to constant order and to estimate p_{succ} respectively. We describe amplitude

estimation more formally:

Theorem 6.12 (Amplitude estimation)**.** *Suppose $\mathcal{O}_p$ is a unitary with $\mathcal{O}_p|0\rangle_B = |\mathsf{coin}\ p\rangle_B$. Then there is a unitary procedure $\mathsf{AE}(\epsilon, \delta)$, making $O(\frac{1}{\epsilon}\log\frac{1}{\delta})$ queries to $\mathcal{O}_p$ and $\mathcal{O}_p^\dagger$, that on input $|\mathsf{coin}\ p\rangle_B|0\rangle_P$ prepares a state of the form*

$$|\mathsf{coin}\ p\rangle_B\Big(\sum_{p'}\alpha_{p'}|p'\rangle_P + \alpha|p_\perp\rangle_P\Big), \tag{6.15}$$

where $|\alpha| \coloneqq \sqrt{1-\sum_{p'}|\alpha_{p'}|^2} \leq \delta$, $\langle p'|p_\perp\rangle = 0$ for all p', and $|p'-p| \leq \epsilon$ for all p'.

Strictly speaking, the parts of Theorem 6.12 involving δ come from measuring the output state of the original amplitude estimation procedure [121] $O(\log\frac{1}{\delta})$ times and taking the median. This can be made coherent by the principle of deferred measurement.

6.6.2 Variable-time amplitude amplification and estimation

In this section we review variable-time amplitude amplification (VTAA) and estimation (VTAE), which are essential components of our algorithm. VTAA and VTAE are procedures applied on top of so-called "variable-time" quantum algorithms, which can be formally defined as follows:

Definition 6.1 (Variable-time quantum algorithm, cf. [102, Section 3.3] and [103, Section 5.1])**.** Let $\mathcal{A}$ be a quantum algorithm in a space $\mathcal{H}$ that starts in the state $|0\rangle_{\mathcal{H}}$, the all-zeros state in $\mathcal{H}$. We say $\mathcal{A}$ is a *variable-time quantum algorithm* if the following conditions hold:

1. $\mathcal{A}$ is the product of m sub-algorithms, $\mathcal{A} = \mathcal{A}_m\mathcal{A}_{m-1}\cdots\mathcal{A}_1$.

2. $\mathcal{H}$ is a tensor product $\mathcal{H} = \mathcal{H}_C \otimes \mathcal{H}_A$, where $\mathcal{H}_C$ is a tensor product of m single-qubit registers denoted $\mathcal{H}_{C_1}, \mathcal{H}_{C_2}, \ldots, \mathcal{H}_{C_m}$.

3. Each $\mathcal{A}_j$ is a controlled unitary that acts on the registers $\mathcal{H}_{C_j} \otimes \mathcal{H}_A$ controlled on the first $j-1$ qubits of $\mathcal{H}_C$ being set to $|0\rangle$.

4. The final state of the algorithm, $\mathcal{A}|0\rangle_{\mathcal{H}}$, is perpendicular to $|0\rangle_C := |0\rangle_{C_1}|0\rangle_{C_2} \cdots |0\rangle_{C_m}$.

In each iteration of the variable-time algorithm we shall construct, we use a subroutine that we call *gapped amplitude estimation* (GAE). Standard amplitude estimation [121] performs phase estimation on a particular unitary, and GAE is essentially the same as "gapped phase estimation" [103, Lemma 22] of that unitary. We recall the standard technique of amplitude estimation [121], which we have stated in Theorem 6.12. It implies the following:

Corollary 6.13 (Gapped amplitude estimation)**.** *Suppose $\mathcal{O}_p$ is a unitary with $\mathcal{O}_p|0\rangle = |\mathsf{coin}\ p\rangle$. Then there is a unitary procedure $\mathsf{GAE}(\epsilon, \delta; l)$, making $O(\frac{1}{\epsilon}\log\frac{1}{\delta})$ queries to $\mathcal{O}_p$ and $\mathcal{O}_p^\dagger$, that on input $|\mathsf{coin}\ p\rangle_B|0\rangle_C|0\rangle_P$, prepares a state of the form*

$$|\mathsf{coin}\ p\rangle_B(\beta_0|0\rangle_C|\gamma_0\rangle_P + \beta_1|1\rangle_C|\gamma_1\rangle_P), \tag{6.16}$$

where $\beta_0, \beta_1 \in [0,1]$ satisfy $\beta_0^2 + \beta_1^2 = 1$ with $\beta_1 \leq \delta$ if $p \geq l - \epsilon$ and $\beta_0 \leq \delta$ if $p < l - 2\epsilon$.

Proof. We first run $\mathsf{AE}(\epsilon/4, \delta)$ on registers B, P. Then, in register C, we output 1 if the value stored in register P is closer to $l - \epsilon$, and output 0 if it is closer to $l - 2\epsilon$. This gives the desired unitary procedure. For convenience, we put any phase factors on the β_i into the $|\gamma_i\rangle$. □

Theorem 6.14 (Variable-time amplitude amplification and estimation [102, 103, 104])**.** *Let $\mathcal{A} = \mathcal{A}_m \cdots \mathcal{A}_1$ be a variable-time quantum algorithm on the space $\mathcal{H} = \mathcal{H}_C \otimes \mathcal{H}_F \otimes \mathcal{H}_W$. Let $|0\rangle_{\mathcal{H}}$ be*

the all-zeros state in $\mathcal{H}$ and let t_j be the query complexity of the algorithm $\mathcal{A}_j \cdots \mathcal{A}_1$. We define

$$w_j \coloneqq \left\| \Pi_{C_j} \mathcal{A}_j \cdots \mathcal{A}_1 |0\rangle_{\mathcal{H}} \right\|^2 \qquad \text{and} \qquad t_{\text{avg}} \coloneqq \sqrt{\textstyle\sum_{j=1}^{m} w_j t_j^2} \tag{6.17}$$

to be the probability of halting at step j and the root-mean-square average query complexity of the algorithm, respectively, where Π_{C_j} denotes the projector onto $|1\rangle$ in $\mathcal{H}_{C_j}$. We also define

$$p_{\text{succ}} \coloneqq \| \Pi_F \mathcal{A}_m \cdots \mathcal{A}_1 |0\rangle_{\mathcal{H}} \|^2 \quad \text{and} \quad |\psi_{\text{succ}}\rangle \coloneqq \frac{\Pi_F \mathcal{A}_m \cdots \mathcal{A}_1 |0\rangle_{\mathcal{H}}}{\| \Pi_F \mathcal{A}_m \cdots \mathcal{A}_1 |0\rangle_{\mathcal{H}} \|} \tag{6.18}$$

to be the success probability of the algorithm and the corresponding output state, respectively, where Π_F projects onto $|1\rangle$ in $\mathcal{H}_F$. Then there exists a quantum algorithm that uses $O(Q)$ queries to output the state $|\psi_{\text{succ}}\rangle$ with probability $\geq 1/2$ and a bit indicating whether it succeeds, where

$$Q \coloneqq t_m \log(t_m) + \frac{t_{\text{avg}}}{\sqrt{p_{\text{succ}}}} \log(t_m). \tag{6.19}$$

There also exists a quantum algorithm that uses $O(\frac{Q}{\epsilon} \log^2(t_m) \log\log(\frac{t_m}{\delta}))$ queries to estimate p_{succ} with multiplicative error ϵ with probability $\geq 1 - \delta$.

6.6.3 Quantum lower bounds by the adversary method

Suppose we have n multi-armed bandit oracles $\mathcal{O}_x$, $x \in [n]$, corresponding to n multi-armed bandits where the best arm is located at a different index in each. Suppose that we also have a best-arm identification algorithm $\mathcal{A}$ that uses no more than T queries to identify the best arm with probability $\geq 1 - \delta$.

The basic quantum adversary method [129, 130] considers a quantity of the form

$$s_k := \sum_{x \neq y} w_{x,y} \langle \psi_x^{(k)} | \psi_y^{(k)} \rangle, \tag{6.20}$$

where $k \in \{0, 1, \ldots, T\}$, $x, y \in [n]$, $w_{x,y} \geq 0$, and $|\psi_x^{(k)}\rangle$ is the state of $\mathcal{A}$ after the k^{th} query to the oracle $\mathcal{O}_x$.

At step $k = 0$, $\mathcal{A}$ has made no queries to the oracle, so $|\psi_x^{(0)}\rangle$ must be the same for all x. Therefore $s_0 = \sum_{x \neq y} w_{x,y}$ as $\langle \psi_x^{(0)} | \psi_y^{(0)} \rangle = 1$.

At step $k = T$, $\mathcal{A}$ must output the index of the best arm with probability $\geq 1 - \delta$. Since the location of the best arm is different for each $\mathcal{O}_x$, the states $|\psi_x^{(T)}\rangle$ must be distinguishable by a quantum measurement with probability $\geq 1 - \delta$. This means that $|\langle \psi_x^{(T)} | \psi_y^{(T)} \rangle| \leq 2\sqrt{\delta(1-\delta)}$. Therefore $|s_T| \leq 2\sqrt{\delta(1-\delta)} \cdot \sum_{x \neq y} w_{x,y}$.

Combining the above observations, we have

$$|s_0 - s_T| \geq |s_0| - |s_T| \geq (1 - 2\sqrt{\delta(1-\delta)}) \cdot \sum_{x \neq y} w_{x,y}. \tag{6.21}$$

Hence, if we can upper bound $|s_{k+1} - s_k|$ by B for some constant B, we can deduce that

$$T \geq \frac{1 - 2\sqrt{\delta(1-\delta)}}{B} \cdot \sum_{x \neq y} w_{x,y}, \tag{6.22}$$

giving a lower bound on the query complexity.

Note that we apply the quantum adversary method to multi-armed bandit oracles of the form given in (6.2), whereas most results from the literature on quantum lower bounds assume a different form of oracle. We remark that [131] treats a more general class of oracles, so it should

be possible to prove Theorem 6.11 using its results. However, we give a self-contained proof using the formulation described above as this approach is straightforward in our case.

6.7 Proof Details of the Quantum Upper Bound

6.7.1 Proof of Lemma 6.4

We first state a more detailed version of Lemma 6.4. We say that states $|\psi\rangle$ and $|\phi\rangle$ are ϵ-close if $\||\psi\rangle - |\phi\rangle\| \leq \epsilon$.

Lemma 6.15 (Full version of Lemma 1, correctness of $\mathcal{A}$)**.** *The output state $|\phi(\mathcal{A})\rangle$ of $\mathcal{A}$ is (α/n)-close to*

$$\begin{aligned}|\psi(\mathcal{A})\rangle \coloneqq \frac{1}{\sqrt{n}} \sum_{S_{\text{right}}} |i\rangle_I |\mathsf{coin}\ p_i\rangle_B |\psi_i\rangle_{C,P} |1\rangle_F \\ + \frac{1}{\sqrt{n}} \sum_{S_{\text{left}}} |i\rangle_I |\mathsf{coin}\ p_i\rangle_B |\psi_i\rangle_{C,P} |0\rangle_F \\ + \frac{1}{\sqrt{n}} \sum_{S_{\text{middle}}} |i\rangle_I |\mathsf{coin}\ p_i\rangle_B (\beta_{i,1} |\psi_{i,1}\rangle_{C,P} |1\rangle_F + \beta_{i,0} |\psi_{i,0}\rangle_{C,P} |0\rangle_F)\end{aligned}$$

for some $\beta_{i,1}, \beta_{i,0} \in \mathbb{C}$ and states $|\psi_i\rangle, |\psi_{i,j}\rangle$. In particular, we have $|p_{\text{succ}} - p'_{\text{succ}}| \leq \frac{2\alpha}{n}$ where $p_{\text{succ}} \coloneqq \|\Pi_F |\phi(\mathcal{A})\rangle\|^2$ and $p'_{\text{succ}} \coloneqq \|\Pi_F |\psi(\mathcal{A})\rangle\|^2 = \frac{1}{n}\big(|S_{\text{right}}| + \sum_{i \in S_{\text{middle}}} |\beta_{i,1}|^2\big)$.

As our proof is similar to that presented in Section 5.3 of [103], we only sketch it in a way that highlights the differences. For comparison, it may be helpful to note that our states $|i\rangle_I |\mathsf{coin}\ p_i\rangle$ are analogous to the matrix eigenstates $|\lambda\rangle$ in [103]. The controlled-NOT operation in Line 9 of our Algorithm 1 takes the place of the simulation subroutine called "W" in Lemma 23 of [103], which is much more elaborate.

We proceed with the proof sketch. Let $\mathcal{A}_{\text{main}} \coloneqq \mathcal{A}_{m+1} \cdots \mathcal{A}_1$ denote the part of $\mathcal{A}$ after initialization. We show that, for each fixed i, $\mathcal{A}_{\text{main}}|i\rangle_I|\mathsf{coin}\ p_i\rangle_B|0\rangle_{C,P,F}$ is $(\frac{\alpha}{n^{3/2}})$-close to

1. Case $i \in S_{\text{middle}}$: $|i\rangle_I|\mathsf{coin}\ p_i\rangle_B(\beta_{i,1}|\psi_i\rangle_{C,P}|1\rangle_F + \beta_{i,0}|\psi_{i,0}\rangle_{C,P}|0\rangle_F)$ for some $\beta_{i,1}, \beta_{i,0} \in \mathbb{C}$ and states $|\psi_i\rangle, |\psi_{i,j}\rangle$;
2. Case $i \in S_{\text{right}}$: $|i\rangle_I|\mathsf{coin}\ p_i\rangle_B|\psi_i\rangle_{C,P}|1\rangle_F$;
3. Case $i \in S_{\text{left}}$: $|i\rangle_I|\mathsf{coin}\ p_i\rangle_B|\psi_i\rangle_{C,P}|0\rangle_F$.

Then $|\phi(\mathcal{A})\rangle = \mathcal{A}|0\rangle_{I,B,C,P,F} = \mathcal{A}_{\text{main}}\frac{1}{\sqrt{n}}\sum_{i=1}^n |i\rangle_I|\mathsf{coin}\ p_i\rangle_B|0\rangle_{C,P,F}$ is $(\frac{1}{\sqrt{n}} \cdot n \cdot \frac{\alpha}{n^{3/2}} = \frac{\alpha}{n})$-close to $|\psi(\mathcal{A})\rangle$ as claimed.

Case $i \in S_{\text{middle}}$**.** This is trivially true because $\beta_{i,1}|\psi_{i,1}\rangle_{C,P}|1\rangle_F + \beta_{i,0}|\psi_{i,0}\rangle_{C,P}|0\rangle_F$ can represent any state on registers C, P, F.

Case $i \in S_{\text{left}}$**.**

Let $j \in [m-1]$ be such that $l_1 - 2\epsilon_j \le p_i < l_1 - \epsilon_j$. Note that this j uniquely exists by the definition of S_{left}, m, and ϵ_j. Then the state of the algorithm after the $(j-1)$st iteration of the for-loop in Line 5 is $(2(j-1)a)$-close to

$$|i\rangle_I|\mathsf{coin}\ p_i\rangle_B|0\rangle_C|\gamma_0^1\rangle_{P_1}\cdots|\gamma_0^{j-1}\rangle_{P_{j-1}}|0\rangle_{P_j\cdots P_m}|1\rangle_F, \tag{6.23}$$

where, for each i, the state $|0\rangle_{C_i}|\gamma_0\rangle_{P_i}$ corresponds to the state $|0\rangle_C|\gamma_0\rangle$ in $\mathsf{GAE}(\epsilon_j, a; l_1)$. Note that we incur an error of at most $2a$ at each iteration which comes from running $\mathsf{GAE}(\epsilon_j, a; l_1)$ (cf. the case where $\beta_1 \le a$ in Corollary 6.13). This error accumulates additively.

The state after the j^{th} iteration is $(2ja)$-close to

$$\begin{aligned}&\beta_0|i\rangle_I|\text{coin } p_i\rangle_B|0\rangle_C|\gamma_0^1\rangle_{P_1}\cdots|\gamma_0^j\rangle_{P_j}|0\rangle_{P_{j+1}\cdots P_m}|1\rangle_F\\&+\beta_1|i\rangle_I|\text{coin } p_i\rangle_B|\boldsymbol{j}\rangle_C|\gamma_0^1\rangle_{P_1}\cdots|\gamma_0^j\rangle_{P_j}|0\rangle_{P_{j+1}\cdots P_m}|1\rangle_F,\end{aligned} \tag{6.24}$$

where $\boldsymbol{j} \coloneqq 0^{j-1}10^{m-j}$ denotes a unary representation of the integer j.

At the $(j+1)$st iteration, the part of the state in the second line of Equation (6.24) is unchanged because its register C indicates "stop", but the part in the first line of Equation (6.24) changes to being $(2(j+1)a)$-close to

$$\beta_0|i\rangle_I|\text{coin } p_i\rangle_B|\boldsymbol{j+1}\rangle_C|\gamma_0^1\rangle_{P_1}\cdots|\gamma_0^j\rangle_{P_j}|\gamma_0^{j+1}\rangle_{P_{j+1}}|0\rangle_{P_{j+2}\cdots P_m}|0\rangle_F. \tag{6.25}$$

Hence, the state after the $(j+1)$st iteration is $(2(j+1)a)$-close to

$$\begin{aligned}&\beta_0|i\rangle_I|\text{coin } p_i\rangle_B|\boldsymbol{j+1}\rangle_C|\gamma_0^1\rangle_{P_1}\cdots|\gamma_0^j\rangle_{P_j}|\gamma_0^{j+1}\rangle_{P_{j+1}}|0\rangle_{P_{j+2}\cdots P_m}|0\rangle_F\\&+\beta_1|i\rangle_I|\text{coin } p_i\rangle_B|\boldsymbol{j}\rangle_C|\gamma_0^1\rangle_{P_1}\cdots|\gamma_0^j\rangle_{P_j}|0\rangle_{P_{j+1}\cdots P_m}|0\rangle_F.\end{aligned} \tag{6.26}$$

Since the C register of all parts of the state in Equation (6.26) indicates "stop", the remaining iterations $j+2,\ldots,m$ of $\mathcal{A}$ do not alter it. Hence the final state of $\mathcal{A}$ is $(2ma)$-close to the state in Equation (6.26), which is of the form

$$|i\rangle_I|\text{coin } p_i\rangle_B|\psi_i\rangle_{C,P}|0\rangle_F. \tag{6.27}$$

Note that $2ma = \frac{\alpha}{n^{3/2}}$, so the closeness of approximation is as claimed.

Case $i \in S_{\text{right}}$**.** In this case, there does not exist a $j \in [m-1]$ such that $l_1 - 2\epsilon_j \le p_i < l_1 - \epsilon_j$.

Thus a simplified version of the argument above, in which we do not have to consider different cases according to the iteration number, shows that the resulting state is $(2ma)$-close to a state of the same form as Equation (6.27) but with the F register remaining in state 1.

Lastly, we show that p_{succ} is close to p'_{succ} as claimed:

$$\begin{aligned} |p_{\text{succ}} - p'_{\text{succ}}| &= \left| \left(\sqrt{p_{\text{succ}}} + \sqrt{p'_{\text{succ}}}\right) \cdot \left(\sqrt{p_{\text{succ}}} - \sqrt{p'_{\text{succ}}}\right) \right| \\ &= \left(\sqrt{p_{\text{succ}}} + \sqrt{p'_{\text{succ}}}\right) \cdot \left| \|\Pi_F|\phi(\mathcal{A})\rangle\| - \|\Pi_F|\psi(\mathcal{A})\rangle\| \right| \\ &\leq 2\,\|\Pi_F(|\phi(\mathcal{A})\rangle - |\psi(\mathcal{A})\rangle)\| \\ &\leq 2\,\frac{\alpha}{n}. \end{aligned} \tag{6.28}$$

6.7.2 Proof of Lemma 6.5

The proof is similar to that presented in Section 5.4 of [103]. For the first claim, note first that $\mathcal{A}_0$ and $\mathcal{A}_{m+1}$ use a constant number of queries (1 and 0, respectively), so we can ignore them. For $k \in [m]$, $\mathcal{A}_k$ only uses queries to perform $\mathsf{GAE}(\epsilon_k, d; l_1)$, which takes $O(\frac{1}{\epsilon_k}\log\frac{1}{a})$ queries. Therefore t_j, the number of queries in $\mathcal{A}_j\mathcal{A}_{j-1}\cdots\mathcal{A}_1$, is of order

$$\sum_{k=1}^{j} \frac{1}{\epsilon_k}\log\left(\frac{1}{a}\right) = \sum_{k=1}^{j} 2^k \log\left(\frac{1}{a}\right) \leq 2^{j+1}\log\left(\frac{1}{a}\right) \tag{6.29}$$

because $\epsilon_k = 2^{-k}$ by definition. In addition, we have $t_m = O(\frac{1}{\Delta}\log\frac{1}{a})$ because $m = \left\lceil \log\frac{1}{\Delta} \right\rceil + 2$ by definition. The first claim follows.

For the second claim, we have

$$t_{\text{avg}}^2 = \sum_{j=1}^{m} w_j t_j^2 = \sum_{j=1}^{m} \left\| \Pi_{C_j} \mathcal{A}_j \cdots \mathcal{A}_1 \frac{1}{\sqrt{n}} \sum_{i=1}^{n} |i\rangle_I |\text{coin } p_i\rangle_B |0\rangle_C |0\rangle_P |1\rangle_F \right\|^2 t_j^2 \tag{6.30}$$

$$= \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,j} t_j^2 \tag{6.31}$$

$$= \frac{1}{n} \sum_{i=1}^{n} \tau_i^2, \tag{6.32}$$

where $w_{i,j} \coloneqq \left\| \Pi_{C_j} \mathcal{A}_j \cdots \mathcal{A}_1 |i\rangle_I |\text{coin } p_i\rangle_B |0\rangle_C |0\rangle_P |1\rangle_F \right\|^2 \in [0,1]$ and $\tau_i \coloneqq \sum_{j=1}^{m} w_{i,j} t_j^2$.

Note that $w_{i,j}$ can be thought of as the probability that $\mathcal{A}$ stops at the end of iteration j if initialized with arm i; τ_i^2 can be thought of as the squared average stopping time of $\mathcal{A}$ if initialized with arm i.

For each fixed i, we consider τ_i^2 according to the following three cases.

Case $i \in S_{\text{right}}$**.** We have $\sum_{j=1}^{m} w_{i,j} = 1$, so $\tau_i^2 \le t_m^2 = O(2^{2m} \log^2(\frac{1}{a})) = O(\frac{1}{\Delta^2} \log^2(\frac{1}{a}))$ because $m = \lceil \log \frac{1}{\Delta} \rceil + 2$ by definition.

Case $i \in S_{\text{middle}}$**.** We still have $\tau_i^2 = O(\frac{1}{\Delta^2} \log^2(\frac{1}{a}))$ as in the case $i \in S_{\text{right}}$, by exactly the same argument. But by the definition of S_{middle}, we have $l_1 - p_i \le \Delta/2$, so we can also write $\tau_i^2 = O\big(\frac{1}{(l_1 - p_i)^2} \log^2(\frac{1}{a})\big)$.

Case $i \in S_{\text{left}}$**.** For $i \in S_{\text{left}}$, let $j \in [m-1]$ be such that $l_1 - 2\epsilon_j \le p_i < l_1 - \epsilon_j$ as in the proof of Lemma 6.4.

We know that after the $(j+1)$st iteration, the state is $(ma = \alpha/n)$-close to the state in (6.26) on which the algorithm terminates. Therefore, the probability $w_{i,j+1}$ of terminating after the $(j+1)^{\text{st}}$ iteration is $1 - O((\alpha/n)^2)$. It can also be seen that the probability $w_{i,j+r}$ of terminating

after the $(j+r)^{\text{th}}$ iteration is $(1 - O((\alpha/n)^2)) \cdot O((\alpha/n)^{2(r-1)})$. Hence

$$\tau_i^2 \le t_{j+1}^2 + O\Big(\sum_{r=2}^{m-j} \Big(\frac{\alpha}{n}\Big)^{2(r-1)} t_{j+r}^2\Big) = O(t_{j+1}^2) = O\left(\frac{\log^2(\frac{1}{a})}{\epsilon_{j+1}}\right) = O\left(\frac{\log^2(\frac{1}{a})}{(l_1 - p_i)^2}\right), \tag{6.33}$$

where we used $\epsilon_{j+1} = \epsilon_j/2 \ge (l_1 - p_i)/4$ for the last inequality.

Substituting the above results into (6.32) tells us that t_{avg}^2 is of order

$$\frac{1}{n}\Big(\frac{|S_{\text{right}}|}{\Delta^2} + \sum_{i \in S_{\text{left}} \cup S_{\text{middle}}} \frac{1}{(l_1 - p_i)^2}\Big) \cdot \log^2\Big(\frac{1}{a}\Big) \tag{6.34}$$

as desired.

6.7.3 Proof of Lemma 6.6

We set the approximation parameter in $\mathcal{A}$ to be $\alpha = c\delta$ for some constant $c < 0.05$ to be determined later. Then $\alpha < 0.05$.

We apply VTAA (Theorem 6.14) on $\mathcal{A}$. This gives an algorithm that outputs a state $|\psi_{\text{succ}}\rangle$ that is $(\frac{\alpha}{n} = \frac{c\delta}{n})$-close to the (normalized) state proportional to

$$\Pi_F |\psi(\mathcal{A})\rangle = \frac{1}{\sqrt{n}}\Big(\sum_{i \in S_{\text{right}}} |i\rangle_I |\mathsf{coin}\ p_i\rangle_B |\psi_i\rangle_{C,P} |1\rangle_F + \sum_{i \in S_{\text{middle}}} \alpha_{i,1} |i\rangle_I |\mathsf{coin}\ p_i\rangle_B |\psi_{i,1}\rangle_{C,P} |1\rangle_F\Big) \tag{6.35}$$

with success probability at least $1/2$ and a bit indicating success or failure. Now, we repeat the entire procedure $O(\log \frac{1}{\delta})$ times to prepare $|\psi_{\text{succ}}\rangle$ at least once with probability $\ge 1 - \delta/2$. Once $|\psi_{\text{succ}}\rangle$ has been successfully prepared, as indicated by the algorithm, we measure its index register I. This procedure outputs an arm index in $S_{\text{right}} \cup S_{\text{middle}}$ with probability $\ge (1 - \delta/2) \cdot$

$(1 - 2c\delta/n)$ which is $\geq 1 - \delta$ for $c \leq 1/4$ sufficiently small. So, as we also need $c < 0.05$, we choose $c = 0.01$. We call this procedure $\mathsf{Amplify}(\mathcal{A}, \delta)$.

Let us consider the query complexity of $\mathsf{Amplify}(\mathcal{A}, \delta)$. We have

$$t_{m+1} = O\Big(\frac{1}{\Delta}\log\Big(\frac{1}{a}\Big)\Big) = O\Big(\frac{1}{\Delta}\log\Big(n\log\Big(\frac{1}{\Delta}\Big)\Big)\Big) \tag{6.36}$$

because $a = \frac{\alpha}{2(\lceil\log(1/\Delta)\rceil+2)n^{3/2}}$ by definition. We also have

$$p_{\text{succ}} \geq p'_{\text{succ}} - \frac{2\alpha}{n} \geq \frac{|S_{\text{right}}|}{n} - \frac{0.1}{n} > \frac{|S_{\text{right}}|}{2n}, \tag{6.37}$$

where we used the assumption $|S_{\text{right}}| > 0$ for the last inequality. Lastly, t^2_{avg} is of order given in (6.12) (reproduced in (6.34) above). Therefore, substituting all these bounds into (6.19) of Theorem 6.14, we see that $\mathsf{Amplify}(\mathcal{A}, \delta)$ has query complexity of order

$$\Big(\frac{1}{\Delta^2} + \frac{1}{|S_{\text{right}}|}\sum_{S_{\text{left}}\cup S_{\text{middle}}}\frac{1}{(l_1 - p_i)^2}\Big)\cdot\log\Big(\frac{n}{\delta}\log\frac{1}{\Delta}\Big)\cdot\log\Big(\frac{1}{\Delta}\log\Big(\frac{n}{\delta}\log\Big(\frac{1}{\Delta}\Big)\Big)\Big)\cdot\log\Big(\frac{1}{\delta}\Big). \tag{6.38}$$

We also apply VTAE (Theorem 6.14) with multiplicative accuracy ϵ and confidence δ on $\mathcal{A}$. This gives an algorithm, $\mathsf{Estimate}(\mathcal{A}, \epsilon, \delta)$, that outputs an estimate r of p_{succ} with multiplicative accuracy ϵ (i.e., $|r - p_{\text{succ}}| < \epsilon p_{\text{succ}}$) with probability $\geq 1 - \delta$. Combining $|r - p_{\text{succ}}| < \epsilon p_{\text{succ}}$ with $|p_{\text{succ}} - p'_{\text{succ}}| \leq \frac{2\alpha}{n} < \frac{0.1}{n}$ gives

$$(1 - \epsilon)\Big(p'_{\text{succ}} - \frac{0.1}{n}\Big) < r < (1 + \epsilon)\Big(p'_{\text{succ}} + \frac{0.1}{n}\Big) \tag{6.39}$$

as claimed.

The query complexity of $\mathsf{Estimate}(\mathcal{A}, \epsilon, \delta)$ is given by (6.38) times

$$\frac{1}{\epsilon} \log^2(t_{m+1}) \log\Big(\log\Big(\frac{t_{m+1}}{\delta}\Big)\Big) = O\Big(\frac{1}{\epsilon} \mathsf{poly}\Big(\log\Big(\frac{n}{\delta\Delta}\Big)\Big)\Big) \tag{6.40}$$

according to Theorem 6.14 and (6.36).

6.7.4 Proof of Lemma 6.7

From the first claim of Lemma 6.8, we see that the probability of E^c is at most $\frac{\delta}{4} \sum_{i=0}^{\infty} 2^{-i} = \delta/2$, where the geometric series arises because of Line 6. Henceforth, we assume E.

Consider the first claim. For given intervals I_2, I_1, let us write

$$\mathrm{gap}(I_2, I_1) \coloneqq \min I_1 - \max I_2. \tag{6.41}$$

At the end of iteration $i \geq 1$ (i.e., after Line 6), we have $|I_k| = (3/5)^i$ by the first claim of Lemma 6.8. At the end of iteration $\left\lceil \log_{5/3}(\frac{1}{\Delta_2}) \right\rceil + 3$, we have $|I_k| < \Delta_2/4$, so $\mathrm{gap}(I_2, I_1) > \Delta_2 - 2\Delta_2/4 = \Delta_2/2 > 2\,|I_1|$ because $p_k \in I_k$. Therefore the while loop must break at this point if it has not done so earlier. For the returned I_k, we clearly have $p_k \in I_k$ because E holds, and $\mathrm{gap}(I_2, I_1) > 2\,|I_1|$ because the while loop has broken. During the while loop, because $|I_k|$ decreases from iteration to iteration, we always have $|I_k| \geq (3/5)^{\left\lceil \log_{5/3}(\Delta_2^{-1}) \right\rceil + 3} \geq \Delta_2/8$. Note that $|I_1| = |I_2|$ because, at each iteration of the while loop, the Shrink subroutine always shrinks intervals by the same factor of $3/5$ and $|I_1| = |I_2| = 1$ initially.

Now, consider the second claim. From the first claim, we know that the while loop breaks at or before the end of iteration $\left\lceil \log_{5/3}(\Delta_2^{-1}) \right\rceil + 3$, and we always have $1/\delta_i = O(2^{\log_{5/3}(\Delta_2^{-1})}/\delta) =$

$O(\Delta_2^{-2}/\delta)$, where $\delta_i = \delta/2^{2+i}$ is the confidence parameter in Shrink at iteration i. Therefore, using the second claim of Lemma 6.8, the total number of queries used is at most

$$O(\log(\Delta_2^{-1})) \cdot O\Big(\sqrt{H} \cdot \mathsf{poly}\Big(\log\Big(\frac{n}{\Delta_2} \cdot \frac{\Delta_2^{-2}}{\delta}\Big)\Big)\Big), \tag{6.42}$$

which is $O\big(\sqrt{H} \cdot \mathsf{poly}(() \log(\frac{n}{\delta\,\Delta_2}))\big)$ as desired.

6.7.5 Proof of Lemma 6.8

Throughout, we fix $k \in \{0, 1\}$.

For the first claim, it is clear that $|J| = 3\,|I|\,/5$ because all the intervals appearing in Lines 11–14 have length 3ϵ. Our proof that $p_k \in J$ with high probability is similar to that in Section 4 of [122] so we only present a brief sketch below.

Let us write $x_j = a + j\epsilon$ for $j = 0, \ldots, 5$, so that $x_0 = a$ and $x_5 = b$. Let E be the event that both Estimates in Lines 7 and 8 return the correct result. The probability of E^c is at most δ so we restrict to the case of E in the following paragraph.

For $j \in \{1, 2\}$, we can use (6.13) in Lemma 6.6 to see that if $p_k \le x_j$, then $B_j = 0$ because $r_j \le (1 + 0.1)(\frac{k}{n+1} + \frac{0.1}{n+1}) < \frac{k+0.5}{n+1}$, whereas if $p_k \ge x_{j+2}$, then $B_j = 1$ because $r_j \ge (1 - 0.1)(\frac{k+1}{n+1} - \frac{0.1}{n+1}) > \frac{k+0.5}{n+1}$. Here we use the fact $k \in \{1, 2\}$. By considering the contrapositive of the previous two if-then statements, we establish the first claim.

For more details, we refer the reader to Section 4 of [122], in particular its Table 2 and Algorithm 1. Note that in the case of $(B_1, B_2) = (0, 1)$, we could have shrunk the interval to $[a + 2\epsilon, a + 3\epsilon]$ and still maintained $p_k \in J$, as is done in [122]. However, it is important for us to keep the shrinkage factor $(3/5)$ the same in all cases because we use this to prove correctness

in Lemma 6.7.

We now prove the second claim. Since we run Estimate with constant multiplicative error $\epsilon = 0.1$, its query complexity is of order (6.38), which is

$$\frac{1}{\Delta^2} + \frac{1}{|S_{\text{right}}|} \sum_{i \in S_{\text{left}} \cup S_{\text{middle}}} \frac{1}{(l_1 - p_i)^2} \tag{6.43}$$

up to polylog factors, where we recall that $\Delta = l_1 - l_2$. In addition, we recall

$$S_{\text{left}} \cup S_{\text{middle}} = \{i : p_i < l_1 - \Delta/8\} \tag{6.44}$$

from (6.10) and (6.11). Note that $|S_{\text{right}}| > 0$ because we appended an arm with bias $p_0 = 1$.

By assumption, $|I| \geq \Delta_2/8$. So, in view of Lines 5 and 6, we have $\Delta = 2\epsilon = 2\,|I|\,/5 \geq \Delta_2/20$. Therefore $1/\Delta^2 = O(1/\Delta_2^2)$.

We also need to compare $p_1 - p_i$ with $l_1 - p_i$ for $i \in S_{\text{left}} \cup S_{\text{middle}}$. By definition, we have $p_i < l_1 - \Delta/8$, so $l_1 - p_i > \Delta/8$. Note that we also have $|p_k - l_1| \leq |I| = 5\Delta/2$ because $p_k \in I$ by assumption and $l_1 \in I$ by definition. If $k = 1$, this says $|p_1 - l_1| \leq 5\Delta/2$. If $k = 2$, this says $|p_2 - l_1| \leq 5\Delta/2$, but we can still bound

$$|p_1 - l_1| \leq \Delta_2 + |p_2 - l_1| \leq 20\Delta + 5\Delta/2 < 25\Delta. \tag{6.45}$$

So regardless of whether $k = 1$ or $k = 2$, we have that $|p_1 - l_1| < 25\Delta$. Therefore

$$\frac{p_1 - p_i}{l_1 - p_i} = 1 + \frac{p_1 - l_1}{l_1 - p_i} < 1 + \frac{25\Delta}{\Delta/8} = 201, \tag{6.46}$$

and so $1/(l_1 - p_i)^2 = O(1/(p_1 - p_i)^2)$. Hence we have established the second claim.

6.8 Corollaries for the Fixed-budget Setting

As mentioned near the end of the main body, by using a reduction similar to that from Monte Carlo to Las Vegas algorithms, we can construct a fixed-budget algorithm from our fixed-confidence one. For completeness, we state and prove the following result:

Lemma 6.16 (Reduction to fixed confidence)**.** *Let $\mathcal{O}$ be a multi-armed bandit oracle. Suppose that for any $\delta \in (0,1)$, we have an algorithm $\mathcal{A}_c(\delta)$ that with probability $\geq 1-\delta$, terminates before using $T_c(\delta)$ queries to $\mathcal{O}$ and returns the best-arm index $i^* = 1$. Suppose that we also know $T_c(\delta)$. Then, for any positive integer T, we can construct an algorithm $\mathcal{A}_b(T)$ that returns $i^* = 1$ with probability $\geq \min_{\delta\in(0,1)} \exp\bigl(-\lfloor T/T_c(\delta)\rfloor D(\frac{1}{2}\|\delta)\bigr)$ using at most T queries to $\mathcal{O}$, where $D(p\|q)$ is the relative entropy between Bernoulli random variables with bias p and q.*

Proof. Since $T_c(\delta)$ is known, consider the modified version of the fixed-confidence algorithm where the algorithm is forced to halt and return some blank symbol "$\perp$" if the running time exceeds $T_c(\delta)$. We refer to the modified algorithm as $A'_c(\delta)$. $A'_c(\delta)$ returns the best-arm index $i^* = 1$ with probability $\geq 1-\delta$ and returns some symbol in $\{2,\ldots,n,\perp\}$ with probability $\leq \delta$.

For any T, we construct $\mathcal{A}_b(T)$ as follows. Pick some $\delta \in (0,1)$, run $A'_c(\delta)$ $m := \lfloor T/T_c(\delta)\rfloor$ times, and take a majority vote over the outcomes. The failure probability can be upper bounded by the probability that i^* is observed fewer than $m/2$ times. The Chernoff bound upper bounds the latter probability by $\exp(-mD(\frac{1}{2}\|\delta)) = \exp\bigl(-\lfloor T/T_c(\delta)\rfloor D(\frac{1}{2}\|\delta)\bigr)$. But δ was arbitrary, so we can take the δ that minimizes this upper bound. □

As a direct corollary of Theorem 6.3 and Lemma 6.16, we see that when H (therefore T_c) is

known in advance, for sufficiently large T, there is a quantum algorithm using at most T queries that returns the best arm with probability $\geq 1 - \exp(-\Omega(T/\sqrt{H}))$.

6.9 Proof Details of the Quantum Lower Bound

6.9.1 Proof of Theorem 6.11

For convenience, we reproduce the statement of the result:

Theorem 6.11. *Let $p \in (0, 1/2)$. For any biases $p_i \in [p, 1-p]$, any quantum algorithm that identifies the best arm requires $\Omega(\sqrt{H})$ queries to the multi-armed bandit oracle $\mathcal{O}$.*

Proof. We use the adversary method and consider the following n different multi-armed bandit oracles.

In the 1^{st} bandit, we assign bias p_i to arm i. Let $\eta > 0$ be a constant to be determined later. In the x^{th} bandit, $x \in \{2, \ldots, n\}$, we assign bias $p'_1 := p_1 + \eta$ to arm x and p_i to arm i for all $i \neq x$. A best-arm identification algorithm must output arm x on assignment x for all $x \in [n]$ with probability $\geq 1 - \delta$.

Following the adversary method, we consider the sum

$$s_k := \sum_{x>1} \frac{1}{\Delta'^2_x} \langle \psi_x^{(k)} | \psi_1^{(k)} \rangle \tag{6.47}$$

for $x \in [n]$, where $\Delta'_x := p'_1 - p_x$. Clearly

$$s_0 = \sum_{x>1} \frac{1}{\Delta'^2_x}. \tag{6.48}$$

We also have

$$s_T \leq \sum_{x>1} \frac{1}{\Delta_x'^2} \cdot 2\sqrt{\delta(1-\delta)}. \tag{6.49}$$

Next, we bound the difference $|s_{k+1} - s_k|$. For $i > 1$, we let

$$A_i \coloneqq \begin{pmatrix} \sqrt{1-p_i} & \sqrt{p_i} \\ \sqrt{p_i} & -\sqrt{1-p_i} \end{pmatrix}, \tag{6.50}$$

while

$$A_1 \coloneqq \begin{pmatrix} \sqrt{1-p_1'} & \sqrt{p_1'} \\ \sqrt{p_1'} & -\sqrt{1-p_1'} \end{pmatrix}, \tag{6.51}$$

where we recall $p_1' = p_1 + \eta$ by definition.

Now, let us write

$$|\psi_x^{(k)}\rangle = \sum_{z,i,b} \alpha_{x,z,i,b} |z,i,b\rangle, \quad |\psi_1^{(k)}\rangle = \sum_{z,i,b} \alpha_{1,z,i,b} |z,i,b\rangle. \tag{6.52}$$

Then

$$|\psi_x^{(k+1)}\rangle = \mathcal{O}_x |\psi_x^{(k)}\rangle = \sum_{z,b} \alpha_{x,z,x,b} |z,x\rangle A_1 |b\rangle + \sum_{i \neq x} \sum_{z,b} \alpha_{x,z,i,b} |z,i\rangle A_i |b\rangle \tag{6.53}$$

and similarly

$$|\psi_1^{(k+1)}\rangle = \mathcal{O}_1 |\psi_1^{(k)}\rangle = \sum_{z,b} \alpha_{1,z,x,b} |z,x\rangle A_x |b\rangle + \sum_{i \neq x} \sum_{z,b} \alpha_{1,z,i,b} |z,i\rangle A_i |b\rangle. \tag{6.54}$$

Then

$$|s_{k+1} - s_k| \leq \sum_{x>1} \frac{1}{\Delta_x'^2} \left| \langle \psi_x^{(k)} | \mathcal{O}_x^\dagger \mathcal{O}_1 | \psi_1^{(k)} \rangle - \langle \psi_x^{(k)} | \psi_1^{(k)} \rangle \right|. \tag{6.55}$$

Using (6.53) and (6.54), and after cancellations, we find that

$$\langle \psi_x^{(k)} | \mathcal{O}_x^\dagger \mathcal{O}_1 | \psi_1^{(k)} \rangle - \langle \psi_x^{(k)} | \psi_1^{(k)} \rangle = \sum_{z,b,b'} \alpha^*_{x,z,x,b} \alpha_{1,z,x,b'} \langle b | (A_1^\dagger A_x - \mathbb{I}) | b' \rangle. \tag{6.56}$$

With

$$\begin{aligned} \begin{pmatrix} u_x & v_x \\ -v_x & u_x \end{pmatrix} &:= A_1^\dagger A_x - \mathbb{I} \\ &= \begin{pmatrix} \sqrt{(1-p_1')(1-p_x)} + \sqrt{p_1' p_x} - 1 & \sqrt{(1-p_1')p_x} - \sqrt{p_1'(1-p_x)} \\ -\sqrt{(1-p_1')p_x} + \sqrt{p_1'(1-p_x)} & \sqrt{(1-p_1')(1-p_x)} + \sqrt{p_1' p_x} - 1 \end{pmatrix}, \end{aligned} \tag{6.57}$$

we have

$$|s_{k+1} - s_k| \leq \sum_{x>1} \sum_{z,b} \frac{|u_x|}{\Delta_x'^2} |\alpha_{x,z,x,b}| \, |\alpha_{1,z,x,b}| + \sum_{x>1} \sum_{z,b \neq b'} \frac{|v_x|}{\Delta_x'^2} |\alpha_{x,z,x,b}| \, |\alpha_{1,z,x,b'}|. \tag{6.58}$$

Clearly, $|u_x| = 1 - \sqrt{(1-p_1')(1-p_x)} - \sqrt{p_1' p_x} \leq 1 - (1-p_1') - p_x = p_1' - p_x = \Delta_x'$. It can also be seen that $|v_x| \leq \Delta_x'/c(p-\eta)$, where $c(x) := 2\sqrt{x(1-x)}$ is a monotone increasing function when $x \in [0, 1/2]$. For completeness, we prove the latter inequality as an auxiliary Lemma 6.17 immediately after this proof.

We can establish the following bounds using Cauchy-Schwarz:

$$\begin{aligned}\sum_{x>1}\sum_{z,b}\frac{|u_x|}{\Delta_x'^2}\,|\alpha_{x,z,x,b}|\,|\alpha_{1,z,x,b}| &\leq \sqrt{\sum_{x>1,z,b}\frac{|u_x|^2}{\Delta_x'^4}\,|\alpha_{x,z,x,b}|^2}\cdot\sqrt{\sum_{x>1,z,b}|\alpha_{1,z,x,b}|^2}\\ &\leq \sqrt{\sum_{x>1}\frac{1}{\Delta_x'^2}}\end{aligned} \tag{6.59}$$

and

$$\begin{aligned}\sum_{x>1}\sum_{z,b\neq b'}\frac{|v_x|}{\Delta_x'^2}\,|\alpha_{x,z,x,b}|\,|\alpha_{1,z,x,b'}| &= \sum_{b\neq b'}\sum_{x>1,z}\frac{|v_x|}{\Delta_x'^2}\,|\alpha_{x,z,x,b}|\,|\alpha_{1,z,x,b'}|\\ &\leq \sum_{b\neq b'}\sqrt{\sum_{x>1,z}\frac{|v_x|^2}{\Delta_x'^4}\,|\alpha_{x,z,x,b}|^2}\cdot\sqrt{\sum_{x>1,z}|\alpha_{1,z,x,b'}|^2}\\ &\leq \frac{2}{c(p-\eta)}\sqrt{\sum_{x>1}\frac{1}{\Delta_x'^2}}.\end{aligned} \tag{6.60}$$

Therefore, we find that

$$|s_{k+1}-s_k| \leq \left(1+\frac{2}{c(p-\eta)}\right)\sqrt{\sum_{x>1}\frac{1}{\Delta_x'^2}}. \tag{6.61}$$

Hence, from Equations (6.48), (6.49), and (6.61), we find that

$$T \geq \frac{1-2\sqrt{\delta(1-\delta)}}{1+2/c(p-\eta)}\sqrt{\sum_{x>1}\frac{1}{\Delta_x'^2}}. \tag{6.62}$$

We then set $\eta = p(p_1-p_2)/2$. Now, it can be seen that

$$c(p-\eta) = c\left(\left(1-\frac{p_1-p_2}{2}\right)p\right) \geq c(p/2) \tag{6.63}$$

because $p \le 1/2$ and $p_1 - p_2 \le 1$. Moreover, for $x > 1$,

$$\Delta'_x = p_1 + \eta - p_x = \frac{p}{2}(p_1 - p_2) + (p_1 - p_x) \le \Big(1 + \frac{p}{2}\Big)(p_1 - p_x) \le \frac{5}{4}\Delta_x \tag{6.64}$$

because $p_x \le p_2$ and $p \le 1/2$. Therefore, we find that

$$T \ge \frac{4}{5} \cdot \frac{1 - 2\sqrt{\delta(1-\delta)}}{1 + 2/c(p/2)} \sqrt{\sum_{x>1} \frac{1}{\Delta_x^2}}, \tag{6.65}$$

and hence $T = \Omega\big(\sqrt{\sum_{i=2}^{n} \frac{1}{\Delta_i^2}}\big)$. □

Lemma 6.17. *Suppose that $p_1, p_2 \in [p, 1-p]$ where $0 < p \le 1/2$. Then*

$$|\sqrt{(1-p_1)p_2} - \sqrt{(1-p_2)p_1}| \le \frac{|p_1 - p_2|}{2\sqrt{p(1-p)}}, \tag{6.66}$$

and the term in the denominator is optimal.

Proof. Note that

$$\sqrt{(1-p_1)p_2} - \sqrt{(1-p_2)p_1} = \frac{(1-p_1)p_2 - (1-p_2)p_1}{\sqrt{(1-p_1)p_2} + \sqrt{(1-p_2)p_1}} \tag{6.67}$$

$$= \frac{-(p_1 - p_2)}{\sqrt{(1-p_1)p_2} + \sqrt{(1-p_2)p_1}}. \tag{6.68}$$

Therefore, it suffices to prove

$$\sqrt{(1-p_1)p_2} + \sqrt{(1-p_2)p_1} \ge 2\sqrt{p(1-p)}. \tag{6.69}$$

Since $p_1, p_2 \in [p, 1-p]$, we have

$$(p_1 - p)(p_1 - (1-p)) \leq 0 \tag{6.70}$$

$$(p_2 - p)(p_2 - (1-p)) \leq 0 \tag{6.71}$$

$$|2p_1 - 1| \leq 1 - 2p \tag{6.72}$$

$$|2p_2 - 1| \leq 1 - 2p. \tag{6.73}$$

Equations (6.70) and (6.71) are equivalent to

$$p_1 - p_1^2 \geq p(1-p), \quad p_2 - p_2^2 \geq p(1-p). \tag{6.74}$$

Equations (6.72) and (6.73) imply

$$4p_1p_2 - 2p_1 - 2p_2 + 1 = (2p_1 - 1)(2p_2 - 1) \leq (2p-1)^2 = 4p^2 - 4p + 1, \tag{6.75}$$

which gives

$$p_1 + p_2 - 2p_1p_2 \geq 2p - 2p^2. \tag{6.76}$$

Now, we have

$$\left(\sqrt{(1-p_1)p_2}+\sqrt{(1-p_2)p_1}\right)^2 = (1-p_1)p_2+(1-p_2)p_1+2\sqrt{(1-p_1)p_2(1-p_2)p_1} \tag{6.77}$$

$$= p_1+p_2-2p_1p_2+2\sqrt{p_1(1-p_1)}\sqrt{p_2(1-p_2)} \tag{6.78}$$

$$\geq 2p-2p^2+2p(1-p) = (2\sqrt{p(1-p)})^2, \tag{6.79}$$

where the inequality comes from (6.74) and (6.76). Therefore, we have established (6.69). Note that this is optimal as taking $p_1 = p_2 = p$ makes the two sides in (6.69) equal. □

Chapter 7: Conclusion

In this dissertation, we mainly focus on the optimization aspect of the Variational Quantum Algorithms (VQAs). By characterizing sufficient and necessary conditions for the efficient training of VQAs, our analyses:

- highlight the necessity of ansatz designs: The no-go theorem in the QNN landscape (Chapter 2) indicate that, the number of parameters in the variational circuit need to scale linearly with the number of qubits without adapting QNN designs to the classification/regression tasks.

- provide a general framework for examining VQA training dynamics in the over-parameterized regime. The VQE convergence analysis in Chapter 3 can be generalized to other instantiation of VQAs following these steps:

  1. express the optimization dynamics using the linear map $\Xi_t(\cdot)$ identified in Chapter 3;
  2. define the asymptotic dynamics in the limit $p \to \infty$ by replacing $\Xi_t(\cdot)$ with the identity map $\mathsf{id}(\cdot)$;
  3. characterize the robust convergence of the asymptotic dynamics.

  In this dissertation, we showcase the framework with VQEs and QNNs.

- provide a principled way for designing ansatze and VQA algorithms: for VQEs, we tie the over-parameterization threshold to instance-dependent quantities d_{eff} and κ_{eff}. By estimating

and optimizing these quantities we come up with new VQA algorithms in Chapter 5. Similarly, for QNNs, we reveal that the convergence of QNN optimization can be accelerate by designing the readout measurements.

7.1 Limitations and Future Directions

In this section, we discuss the limitations of our works and possible future directions.

Gap between Under- and Over-parameterization Regimes. In this dissertation, we considered the regime where the number of parameters $p = \Omega(\mathsf{poly}(d))$ and $p = \Omega(\log d)$. Yet it is unclear how the dynamics transits from converging to a spurious minima to linearly converging to a almost-global minima. A closely-related recent work that shed some light on this problem is [57], where it characterizes the transition in terms of the landscape. It would be interesting to see if a similar transition can be characterized in terms of rate of convergence.

Beyond Gradient Flow. Our convergence results are established under the optimization of gradient flow, while in practice, it is the time-discretized version, the gradient descent that is commonly used for optimizing VQAs. On one hand, our results in Section 3.5 can be used to provide a theory for VQE convergence under gradient descent, as we can view the gradient descent dynamics as the gradient flow dynamics adding some non-stochastic/adversarial noise. On the other hand, it would be interesting to see if we can achieve a tighter analysis on the over-parameterization threshold following the work of [79] for the gradient descent and its variants.

Towards Better Random Initialization. While our analysis does not depend on a specific form of random initialization, we do require the randomly initialized d-dimensional variational circuit to mix to the haar measure over a subgroup of $SU(d)$. There are other random initialization

scheme (e.g. [132, 133]) being considered in the literature. It would be interesting to see if a convergence theory can be established beyond what we consider in this dissertation.

Bibliography

[1] Andy CY Li, M Sohaib Alam, Thomas Iadecola, Ammar Jahin, Doga Murat Kurkcuoglu, Richard Li, Peter P Orth, A Barış Özgüler, Gabriel N Perdue, and Norm M Tubman. Benchmarking variational quantum eigensolvers for the square-octagon-lattice kitaev model. *arXiv preprint arXiv:2108.13375*, 2021.

[2] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2):303–332, 1999.

[3] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219, 1996.

[4] Daochen Wang, Aarthi Sundaram, Robin Kothari, Ashish Kapoor, and Martin Roetteler. Quantum algorithms for reinforcement learning with a generative model. In *International Conference on Machine Learning*, pages 10916–10926. PMLR, 2021.

[5] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, Brian Burkett, Yu Chen, Zijun Chen, Ben Chiaro, Roberto Collins, William Courtney, Andrew Dunsworth, Edward Farhi, Brooks Foxen, Austin Fowler, Craig Gidney, Marissa Giustina, Rob Graff, Keith Guerin, Steve Habegger, Matthew P. Harrigan, Michael J. Hartmann, Alan Ho, Markus Hoffmann, Trent Huang, Travis S. Humble, Sergei V. Isakov, Evan Jeffrey, Zhang Jiang, Dvir Kafri, Kostyantyn Kechedzhi, Julian Kelly, Paul V. Klimov, Sergey Knysh, Alexander Korotkov, Fedor Kostritsa, David Landhuis, Mike Lindmark, Erik Lucero, Dmitry Lyakh, Salvatore Mandrà, Jarrod R. McClean, Matthew McEwen, Anthony Megrant, Xiao Mi, Kristel Michielsen, Masoud Mohseni, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Murphy Yuezhen Niu, Eric Ostby, Andre Petukhov, John C. Platt, Chris Quintana, Eleanor G. Rieffel, Pedram Roushan, Nicholas C. Rubin, Daniel Sank, Kevin J. Satzinger, Vadim Smelyanskiy, Kevin J. Sung, Matthew D. Trevithick, Amit Vainsencher, Benjamin Villalonga, Theodore White, Z. Jamie Yao, Ping Yeh, Adam Zalcman, Hartmut Neven, and John M. Martinis. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[6] Han-Sen Zhong, Hui Wang, Yu-Hao Deng, Ming-Cheng Chen, Li-Chao Peng, Yi-Han Luo, Jian Qin, Dian Wu, Xing Ding, Yi Hu, Peng Hu, Xiao-Yan Yang, Wei-Jun Zhang,

Hao Li, Yuxuan Li, Xiao Jiang, Lin Gan, Guangwen Yang, Lixing You, Zhen Wang, Li Li, Nai-Le Liu, Chao-Yang Lu, and Jian-Wei Pan. Quantum computational advantage using photons. *Science*, 370(6523):1460–1463, 2020.

[7] M Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J Coles. Variational Quantum Algorithms. *arXiv preprint arXiv:2012.09265*, 2020.

[8] John Preskill. Quantum computing in the NISQ era and beyond. *Quantum*, 2:79, 2018.

[9] Edward Farhi, Hartmut Neven, et al. Classification with quantum neural networks on near term processors. *Quantum Review Letters*, 1(2 (2020)):10–37686, 2020.

[10] Shouvanik Chakrabarti, Huang Yiming, Tongyang Li, Soheil Feizi, and Xiaodi Wu. Quantum wasserstein generative adversarial networks. *Advances in Neural Information Processing Systems*, 32, 2019.

[11] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Sergio Boixo, Michael Broughton, Bob B. Buckley, David A. Buell, Brian Burkett, Nicholas Bushnell, Yu Chen, Zijun Chen, Benjamin Chiaro, Roberto Collins, William Courtney, Sean Demura, Andrew Dunsworth, Edward Farhi, Austin Fowler, Brooks Foxen, Craig Gidney, Marissa Giustina, Rob Graff, Steve Habegger, Matthew P. Harrigan, Alan Ho, Sabrina Hong, Trent Huang, William J. Huggins, Lev Ioffe, Sergei V. Isakov, Evan Jeffrey, Zhang Jiang, Cody Jones, Dvir Kafri, Kostyantyn Kechedzhi, Julian Kelly, Seon Kim, Paul V. Klimov, Alexander Korotkov, Fedor Kostritsa, David Landhuis, Pavel Laptev, Mike Lindmark, Erik Lucero, Orion Martin, John M. Martinis, Jarrod R. McClean, Matt McEwen, Anthony Megrant, Xiao Mi, Masoud Mohseni, Wojciech Mruczkiewicz, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Hartmut Neven, Murphy Yuezhen Niu, Thomas E. O'Brien, Eric Ostby, Andre Petukhov, Harald Putterman, Chris Quintana, Pedram Roushan, Nicholas C. Rubin, Daniel Sank, Kevin J. Satzinger, Vadim Smelyanskiy, Doug Strain, Kevin J. Sung, Marco Szalay, Tyler Y. Takeshita, Amit Vainsencher, Theodore White, Nathan Wiebe, Z. Jamie Yao, Ping Yeh, and Adam Zalcman. Hartree-fock on a superconducting qubit quantum computer. *Science*, 369(6507):1084–1089, 2020.

[12] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5:4213, 2014.

[13] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

[14] Michael A Nielsen and Isaac Chuang. Quantum computation and quantum information, 2002.

[15] John Watrous. *The theory of quantum information*. Cambridge University Press, 2018.

[16] Kenji Kawaguchi. Deep learning without poor local minima. In *Advances in neural information processing systems*, pages 586–594, 2016.

[17] Simon Du and Jason Lee. On the power of over-parametrization in neural networks with quadratic activation. In *International Conference on Machine Learning*, pages 1329–1338. PMLR, 2018.

[18] Kosuke Mitarai, Makoto Negoro, Masahiro Kitagawa, and Keisuke Fujii. Quantum circuit learning. *Physical Review A*, 98(3):032309, 2018.

[19] Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4):043001, 2019.

[20] Seth Lloyd, Maria Schuld, Aroosa Ijaz, Josh Izaac, and Nathan Killoran. Quantum embeddings for machine learning. *arXiv preprint arXiv:2001.03622*, 2020.

[21] John C Napp, Rolando L La Placa, Alexander M Dalzell, Fernando GSL Brandao, and Aram W Harrow. Efficient classical simulation of random shallow 2d quantum circuits. *Physical Review X*, 12(2):021021, 2022.

[22] Mateusz Ostaszewski, Edward Grant, and Marcello Benedetti. Structure optimization for parameterized quantum circuits. *Quantum*, 5:391, 2021.

[23] Peter Auer, Mark Herbster, and Manfred KK Warmuth. Exponentially many local minima for single neurons. In *Advances in neural information processing systems*, pages 316–322, 1996.

[24] Itay Safran and Ohad Shamir. Spurious local minima are common in two-layer relu neural networks. In *International Conference on Machine Learning*, pages 4433–4441. PMLR, 2018.

[25] Chulhee Yun, Suvrit Sra, and Ali Jadbabaie. Small nonlinearities in activation functions create bad local minima in neural networks. In *International Conference on Learning Representations*, 2019.

[26] Tian Ding, Dawei Li, and Ruoyu Sun. Suboptimal local minima exist for wide neural networks with smooth activations. *Math. Oper. Res.*, 47(4):2784–2814, nov 2022.

[27] Luca Venturi, Afonso S. Bandeira, and Joan Bruna. Spurious valleys in one-hidden-layer neural network optimization landscapes. *Journal of Machine Learning Research*, 20(133):1–34, 2019.

[28] Daniel Soudry and Yair Carmon. No bad local minima: Data independent training error guarantees for multilayer neural networks. *arXiv preprint arXiv:1605.08361*, 2016.

[29] Quynh Nguyen and Matthias Hein. The loss surface of deep and wide neural networks. In *International conference on machine learning*, pages 2603–2612. PMLR, 2017.

[30] Dawei Li, Tian Ding, and Ruoyu Sun. Over-parameterized deep neural networks have no strict local minima for any continuous activations. *arXiv preprint arXiv:1812.11039*, 2018.

[31] Arthur Jacot, Clément Hongler, and Franck Gabriel. Neural tangent kernel: Convergence and generalization in neural networks. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 8580–8589, 2018.

[32] Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, Ruslan Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. *arXiv preprint arXiv:1904.11955*, 2019.

[33] Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pages 1675–1685. PMLR, 2019.

[34] Surbhi Goel, Varun Kanade, Adam Klivans, and Justin Thaler. Reliably learning the relu in polynomial time. In *Conference on Learning Theory*, pages 1004–1042. PMLR, 2017.

[35] Surbhi Goel and Adam R Klivans. Learning neural networks with two nonlinear layers in polynomial time. In *Conference on Learning Theory*, pages 1470–1499. PMLR, 2019.

[36] Kai Zhong, Zhao Song, Prateek Jain, Peter L Bartlett, and Inderjit S Dhillon. Recovery guarantees for one-hidden-layer neural networks. In *International conference on machine learning*, pages 4140–4149. PMLR, 2017.

[37] Simon S Du and Surbhi Goel. Improved learning of one-hidden-layer convolutional neural networks with overlaps. *arXiv preprint arXiv:1805.07798*, 2018.

[38] Zhihui Wang, Nicholas C. Rubin, Jason M. Dominy, and Eleanor G. Rieffel. xy mixers: Analytical and numerical results for the quantum alternating operator ansatz. *Phys. Rev. A*, 101:012320, Jan 2020.

[39] Zhihui Wang, Stuart Hadfield, Zhang Jiang, and Eleanor G. Rieffel. Quantum approximate optimization algorithm for maxcut: A fermionic view. *Phys. Rev. A*, 97:022304, Feb 2018.

[40] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature communications*, 9(1):1–6, 2018.

[41] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Leo Zhou. The Quantum Approximate Optimization Algorithm and the Sherrington-Kirkpatrick Model at Infinite Size. *arXiv e-prints*, page arXiv:1910.08187, October 2019.

[42] Herschel A Rabitz, Michael M Hsieh, and Carey M Rosenthal. Quantum optimally controlled transition landscapes. *Science*, 303(5666):1998–2001, 2004.

[43] Benjamin Russell, Herschel Rabitz, and Rebing Wu. Quantum control landscapes are almost always trap free. *arXiv preprint arXiv:1608.06198*, 2016.

[44] Bobak Toussi Kiani, Seth Lloyd, and Reevu Maity. Learning unitaries by gradient descent, 2020.

[45] Fernando GSL Brandao, Aram W Harrow, and Michał Horodecki. Local random quantum circuits are approximate polynomial-designs. *Communications in Mathematical Physics*, 346(2):397–434, 2016.

[46] Zbigniew Puchała and Jarosław Adam Miszczak. Symbolic integration with respect to the haar measure on the unitary group. *arXiv preprint arXiv:1109.4244*, 2011.

[47] Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 1996.

[48] David A Cox, John Little, and Donal O'shea. *Using algebraic geometry*, volume 185. Springer Science & Business Media, 2006.

[49] Rong Ge and Tengyu Ma. On the optimization landscape of tensor decompositions. In *Advances in Neural Information Processing Systems*, pages 3653–3663, 2017.

[50] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035, 2019.

[51] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[52] Yoshua Bengio. Rmsprop and equilibrated adaptive learning rates for nonconvex optimization. *corr abs/1502.04390*, 2015.

[53] Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

[54] Nathan Killoran, Thomas R Bromley, Juan Miguel Arrazola, Maria Schuld, Nicolás Quesada, and Seth Lloyd. Continuous-variable quantum neural networks. *Physical Review Research*, 1(3):033063, 2019.

[55] Andrea Mari, Thomas R Bromley, Josh Izaac, Maria Schuld, and Nathan Killoran. Transfer learning in hybrid classical-quantum neural networks. *Quantum*, 4:340, 2020.

[56] Ryan Sweke, Frederik Wilde, Johannes Jakob Meyer, Maria Schuld, Paul K Fährmann, Barthélémy Meynard-Piganeau, and Jens Eisert. Stochastic gradient descent for hybrid quantum-classical optimization. *Quantum*, 4:314, 2020.

[57] Eric Ricardo Anschuetz. Critical points in quantum generative models. In *International Conference on Learning Representations*, 2022.

[58] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2013.

[59] Benoît Collins and Piotr Śniady. Integration with respect to the haar measure on unitary, orthogonal and symplectic group. *Communications in Mathematical Physics*, 264(3):773–795, 2006.

[60] Dénes Petz and Jaroslav Zemánek. Characterizations of the trace. *Linear Algebra and its Applications*, 111:43–52, 1988.

[61] Dénes Petz. A survey of certain trace inequalities. *Banach Center Publications*, 30(1):287–298, 1994.

[62] Roeland Wiersema, Cunlu Zhou, Yvette de Sereville, Juan Felipe Carrasquilla, Yong Baek Kim, and Henry Yuen. Exploring entanglement and optimization within the Hamiltonian variational Ansatz. *arXiv preprint arXiv:2008.02941*, 2020.

[63] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M Chow, and Jay M Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242–246, 2017.

[64] Xuchen You and Xiaodi Wu. Exponentially many local minima in quantum neural networks. In *International Conference on Machine Learning*, pages 12144–12155. PMLR, 2021.

[65] Péter Pál Varjú. Random walks in compact groups. *arXiv preprint arXiv:1209.1745*, 2012.

[66] Lenaic Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[67] Zhiqiang Xu, Xin Cao, and Xin Gao. Convergence analysis of gradient descent for eigenvector computation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 2933–2939. International Joint Conferences on Artificial Intelligence Organization, 7 2018.

[68] Sonia G Schirmer, H Fu, and Allan I Solomon. Complete controllability of quantum systems. *Physical Review A*, 63(6):063410, 2001.

[69] Re-Bing Wu, Michael A Hsieh, and Herschel Rabitz. Role of controllability in optimizing quantum dynamics. *Physical Review A*, 83(6):062306, 2011.

[70] B. Hall and B.C. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction*. Graduate Texts in Mathematics. Springer, 2003.

[71] Joel A Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of computational mathematics*, 12(4):389–434, 2012.

[72] Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

[73] Eric R Anschuetz and Bobak T Kiani. Quantum variational algorithms are swamped with traps. *Nature Communications*, 13(1):1–10, 2022.

[74] Roi Livni, Shai Shalev-Shwartz, and Ohad Shamir. On the computational efficiency of training neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[75] Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. *CoRR*, abs/1802.05296, 2018.

[76] Abdulkadir Canatar, Blake Bordelon, and Cengiz Pehlevan. Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks. *Nature communications*, 12(1):1–12, 2021.

[77] Zixiang Chen, Yuan Cao, Quanquan Gu, and Tong Zhang. A generalized neural tangent kernel analysis for two-layer neural networks. *Advances in Neural Information Processing Systems*, 33:13363–13373, 2020.

[78] Alberto Bietti and Julien Mairal. On the inductive bias of neural tangent kernels. *Advances in Neural Information Processing Systems*, 32, 2019.

[79] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. *arXiv preprint arXiv:1811.03962*, 2019.

[80] Maria Schuld and Nathan Killoran. Quantum machine learning in feature hilbert spaces. *Physical review letters*, 122(4):040504, 2019.

[81] Hsin-Yuan Huang, Michael Broughton, Masoud Mohseni, Ryan Babbush, Sergio Boixo, Hartmut Neven, and Jarrod R. McClean. Power of data in quantum machine learning. *Nature Communications*, 12(1), may 2021.

[82] Junyu Liu, Francesco Tacchino, Jennifer R. Glick, Liang Jiang, and Antonio Mezzacapo. Representation learning via quantum neural tangent kernels. *arXiv preprint arXiv:2111.04225*, 2021.

[83] Norihito Shirai, Kenji Kubo, Kosuke Mitarai, and Keisuke Fujii. Quantum tangent kernel. *arXiv preprint arXiv:2111.02951*, 2021.

[84] Junyu Liu, Khadijeh Najafi, Kunal Sharma, Francesco Tacchino, Liang Jiang, and Antonio Mezzacapo. An analytic theory for the dynamics of wide quantum neural networks. *arXiv preprint arXiv:2203.16711*, 2022.

[85] Samet Oymak and Mahdi Soltanolkotabi. Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 1(1):84–105, 2020.

[86] Iris Cong, Soonwon Choi, and Mikhail D Lukin. Quantum convolutional neural networks. *Nature Physics*, 15(12):1273–1278, 2019.

[87] Xuchen You, Shouvanik Chakrabarti, and Xiaodi Wu. A convergence theory for over-parameterized variational quantum eigensolvers. *arXiv:2205.12481*, 2022.

[88] Vedran Dunjko and Hans J Briegel. Machine learning and artificial intelligence in the quantum domain: a review of recent progress. *Reports on Progress in Physics*, 81(7):074001, jun 2018.

[89] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[90] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

[91] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[92] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195, 2017.

[93] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. An introduction to quantum machine learning. *Contemporary Physics*, 56(2):172–185, 2015. arXiv:1409.3097

[94] Srinivasan Arunachalam and Ronald de Wolf. Guest column: a survey of quantum learning theory. *ACM SIGACT News*, 48(2):41–67, 2017. arXiv:1701.06806

[95] Vedran Dunjko and Hans J. Briegel. Machine learning & artificial intelligence in the quantum domain: a review of recent progress. *Reports on Progress in Physics*, 81(7):074001, 2018. arXiv:1709.02779

[96] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. PAC bounds for multi-armed bandit and markov decision processes. In Jyrki Kivinen and Robert H. Sloan, editors, *Computational Learning Theory*, pages 255–270, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[97] Iordanis Kerenidis, Jonas Landman, Alessandro Luongo, and Anupam Prakash. q-means: A quantum algorithm for unsupervised machine learning. In *Advances in Neural Information Processing Systems*, pages 4136–4146, 2019. arXiv:1812.03584

[98] Nathan Wiebe, Ashish Kapoor, and Krysta M. Svore. Quantum algorithms for nearest-neighbor methods for supervised and unsupervised learning. *Quantum Information & Computation*, 15(3-4):316–356, 2015. arXiv:1401.2142

[99] Vedran Dunjko, Jacob M. Taylor, and Hans J Briegel. Quantum-enhanced machine learning. *Physical Review Letters*, 117(13):130501, 2016. arXiv:1610.08251

[100] Michael A. Nielsen and Isaac L. Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2000.

[101] Ashley Montanaro. Quantum speedup of Monte Carlo methods. *Proceedings of the Royal Society A*, 471(2181):20150301, 2015.

[102] Andris Ambainis. Variable time amplitude amplification and a faster quantum algorithm for solving systems of linear equations, 2010. arXiv:1010.4458

[103] Andrew M. Childs, Robin Kothari, and Rolando D. Somma. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM Journal on Computing*, 46(6):1920–1950, 2017. arXiv:1511.02306

[104] Shantanav Chakraborty, András Gilyén, and Stacey Jeffery. The power of block-encoded matrix powers: Improved regression techniques via faster Hamiltonian simulation. In *Proceedings of the 46th International Colloquium on Automata, Languages, and Programming*, volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 33:1–33:14. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019. arXiv:1804.01973

[105] Victor Gabillon, Mohammad Ghavamzadeh, and Alessandro Lazaric. Best arm identification: A unified approach to fixed budget and fixed confidence. In *Advances in Neural Information Processing Systems*, pages 3212–3220, 2012.

[106] Kevin Jamieson, Matthew Malloy, Robert Nowak, and Sébastien Bubeck. lil'ucb: An optimal exploration algorithm for multi-armed bandits. In *Conference on Learning Theory*, pages 423–439, 2014. arXiv:1312.7308

[107] Zohar Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In *International Conference on Machine Learning*, pages 1238–1246, 2013.

[108] Shie Mannor and John N. Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research*, 5(Jun):623–648, 2004.

[109] Lijie Chen and Jian Li. On the optimal sample complexity for best arm identification, 2015. arXiv:1511.03774

[110] Lijie Chen, Jian Li, and Mingda Qiao. Towards instance optimal bounds for best arm identification. In *Conference on Learning Theory*, pages 535–592, 2017. arXiv:1608.06031

[111] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. Quantum principal component analysis. *Nature Physics*, 10(9):631, 2014. arXiv:1307.0401

[112] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. Quantum algorithms for supervised and unsupervised machine learning, 2013. arXiv:1307.0411

[113] Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. Quantum support vector machine for big data classification. *Physical Review Letters*, 113(13):130503, 2014. arXiv:1307.0471

[114] Tongyang Li, Shouvanik Chakrabarti, and Xiaodi Wu. Sublinear quantum algorithms for training linear and kernel-based classifiers. In *International Conference on Machine Learning*, pages 3815–3824, 2019. arXiv:1904.02276

[115] Mohammad H. Amin, Evgeny Andriyash, Jason Rolfe, Bohdan Kulchytskyy, and Roger Melko. Quantum Boltzmann machine. *Physical Review X*, 8(2):021050, 2018. arXiv:1601.02036

[116] Vedran Dunjko, Jacob M. Taylor, and Hans J. Briegel. Advances in quantum reinforcement learning. In *Proceedings of the 2017 IEEE International Conference on Systems, Man, and Cybernetics*, pages 282–287. IEEE, 2017. arXiv:1811.08676

[117] Vedran Dunjko, Yi-Kai Liu, Xingyao Wu, and Jacob M. Taylor. Exponential improvements for quantum-accessible reinforcement learning, 2017. arXiv:1710.11160

[118] Sofiene Jerbi, Hendrik Poulsen Nautrup, Lea M. Trenkwalder, Hans J. Briegel, and Vedran Dunjko. A framework for deep energy-based reinforcement learning with quantum speed-up, 2019. arXiv:1910.12760

[119] Balthazar Casalé, Giuseppe Di Molfetta, Hachem Kadri, and Liva Ralaivola. Quantum bandits, 2020. arXiv:2002.06395

[120] Christoph Dürr and Peter Høyer. A quantum algorithm for finding the minimum, 1996. arXiv:quant-ph/9607014

[121] Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305:53–74, 2002. arXiv:quant-ph/0005055

[122] Lin Lin and Yu Tong. Near-optimal ground state preparation, 2020. arXiv:2002.12508

[123] Andris Ambainis. A new quantum lower bound method, with an application to a strong direct product theorem for quantum search. *Theory of Computing*, 6(1):1–25, 2010. arXiv:quant-ph/0508200

[124] Yassine Hamoudi, Maharshi Ray, Patrick Rebentrost, Miklos Santha, Xin Wang, and Siyi Yang. Quantum algorithms for hedging and the sparsitron, 2020. arXiv:2002.06003

[125] Srinivasan Arunachalam and Reevu Maity. Quantum boosting. In *To appear in the Thirty-seventh International Conference on Machine Learning*, 2020. arXiv:2002.05056

[126] Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandit algorithms with supervised learning guarantees. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 19–26, 2011.

[127] Alekh Agarwal, Daniel Hsu, Satyen Kale, John Langford, Lihong Li, and Robert Schapire. Taming the monster: A fast and simple algorithm for contextual bandits. In *International Conference on Machine Learning*, pages 1638–1646, 2014.

[128] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002.

[129] Andris Ambainis. Quantum lower bounds by quantum arguments. *Journal of Computer and System Sciences*, 64(4):750–767, 2002. arXiv:quant-ph/0002066

[130] Peter Høyer and Robert Spalek. Lower bounds on quantum query complexity. *Bulletin of the EATCS*, 87:78–103, 2005. arXiv:quant-ph/0509153

[131] Aleksandrs Belovs. Variations on quantum adversary, 2015. arXiv:1504.06943

[132] Kaining Zhang, Liu Liu, Min-Hsiu Hsieh, and Dacheng Tao. Escaping from the barren plateau via gaussian initializations in deep variational quantum circuits. In *Advances in Neural Information Processing Systems*.

[133] MH Cheng, KE Khosla, CN Self, M Lin, BX Li, AC Medina, and MS Kim. Clifford circuit initialisation for variational quantum algorithms. *arXiv preprint arXiv:2207.01539*, 2022.