Toward Trainability of Deep Quantum Neural Networks

Kaining Zhang\textsuperscript{1}, Min-Hsiu Hsieh\textsuperscript{2}, Liu Lin\textsuperscript{1}, and Dacheng Tao\textsuperscript{1}

\textsuperscript{1}School of Computer Science, Faculty of Engineering, University of Sydney, Australia and 
\textsuperscript{2}Hon Hai Quantum Computing Research Center, Taipei, Taiwan

(Dated: January 3, 2022)

Quantum Neural Networks (QNNs) with random structures have poor trainability due to the exponentially vanishing gradient as the circuit depth and the qubit number increase. This result leads to a general belief that a deep QNN will not be feasible. In this work, we provide the first viable solution to the vanishing gradient problem for deep QNNs with theoretical guarantees. Specifically, we prove that for circuits with controlled-layer architectures, the expectation of the gradient norm can be lower bounded by a value that is independent of the qubit number and the circuit depth. Our results follow from a careful analysis of the gradient behaviour on parameter space consisting of rotation angles, as employed in almost any QNNs, instead of relying on impractical 2-design assumptions. We explicitly construct examples where only our QNNs are trainable and converge, while others in comparison cannot.

I. INTRODUCTION

Neural Networks\textsuperscript{4} using gradient-based optimizations have dramatically advanced research in discriminative models, generative models, and reinforcement learning. To efficiently utilize parameters and practically improve trainability, neural networks with dedicated architectures\textsuperscript{2} are introduced for different tasks, including convolutional neural networks\textsuperscript{8} for image tasks, recurrent neural networks\textsuperscript{4} for time series analysis, and graph neural networks\textsuperscript{9} for tasks related to graph-structured data. Recently, neural architecture search\textsuperscript{6} was proposed to improve the performance of the networks by optimizing their neural structures.

Despite the success in many fields, the development of neural network algorithms is still prone to limitations caused by the large computational resources required for model training. In recent years, quantum computing has emerged as a promising approach to this problem, and has evolved into a new interdisciplinary field known as quantum machine learning (QML)\textsuperscript{7,8}. Specifically, variational quantum circuits\textsuperscript{9} have been explored as efficient protocols for quantum chemistry\textsuperscript{10,11} and combinatorial optimizations\textsuperscript{11,12}. Compared to the classical circuit models, quantum circuits have greater expressive power\textsuperscript{13,14,15} and proven quantum advantage\textsuperscript{16}. Due to their robustness against noises\textsuperscript{16}, variational quantum circuits have attracted significant interest for the hope of achieving the practical quantum supremacy on near-term quantum computers\textsuperscript{17,18}.

Quantum Neural Networks (QNNs)\textsuperscript{19,20} are the special kind of quantum-classical hybrid algorithms that run on trainable quantum circuits. Recently, small-scale QNNs have been implemented on real quantum computers\textsuperscript{8,21} for supervised learning tasks. Inspired by the classical optimizations of neural networks, a natural strategy to train QNNs is to exploit the gradient of the loss function\textsuperscript{22,23}. However, recent works\textsuperscript{24} show that N-qubit quantum circuits with random structures and large depth \((D = \text{poly}(N))\) form approximate unitary 2-designs\textsuperscript{25}, and the partial derivative vanishes to zero exponentially with respect to \(N\).

The vanishing gradient problem is usually referred to as the Barren Plateaus\textsuperscript{24}, which could affect the trainability of QNNs in three folds. 1) Simply using a gradient-based method like Stochastic Gradient Descent (SGD) to train the QNN takes a large number of iterations. 2) The estimation of derivatives requires an extremely large number of samples from the quantum output to guarantee a relatively accurate direction\textsuperscript{26}. 3) The objective function tends to have a flat surface\textsuperscript{26} with globally vanishing gradients, which hardly have practical implications.

A. Related Works

The barren plateau phenomenon of QNNs is first reported in random ansatzes\textsuperscript{24}. Specifically, for \(N\)-qubit random quantum circuits with global 2-design distributed unitaries, the expectation of the derivative to the objective function is zero, and the variance of the derivative vanishes to zero with the rate exponential in the number of qubits. Given the fact that random circuits form approximate 2-designs\textsuperscript{25}, QNNs with random ansatzes are essentially untrainable. Subsequently, Refs.\textsuperscript{28,91} relax the global 2-design assumption to the local case and prove the trainability of shallow alternating layered circuits with local observables. Their results imply that in the low-depth \(D = \text{O}(\log N)\) case, the norm of the gradient could be \(\frac{1}{\text{poly}(N)}\), which is better than previous exponential vanishing results. Similar results hold for the quantum convolutional neural network (QCNN)\textsuperscript{29} which adopts a tree tensor structure\textsuperscript{32} with \(\log N\) depth. Besides, most existing works show a negative relationship between trainability and circuit complexity. Specifically, states generated from QNNs, which satisfy the volume law\textsuperscript{33}, could lead to the barren plateaus landscape. QNNs with high expressibility\textsuperscript{27,31} could also suffer from the vanishing gradient problem. For the depolarizing channel case, noisy QNNs with linear depth could have an exponentially small gradient\textsuperscript{30}. The vanishing
TABLE I. Existing theoretical results of the barren plateaus problem. We use $N$ and $D$ to denote the number of qubits and the circuit depth. We denote by $m$ the number of qubits acted on by local gates. We use $\varepsilon$ to measure the distance between circuit states and Haar random states \cite{27}. We denote by $q < 1$ the noise parameter.

| Structure               | Distribution         | Bounds on the gradient                              |
|-------------------------|----------------------|-----------------------------------------------------|
| Random circuit \cite{24} | Global 2-design      | $\text{Var} \left[ \frac{\partial f}{\partial \theta_j} \right] \approx O(2^{-3N})$ |
| Alternating layered circuit \cite{25} | Local 2-design      | $\text{Var} \left[ \frac{\partial f}{\partial \theta_j} \right] \geq O(2^{-m(D+2)})$ |
| QCNN \cite{29}           | Local 2-design      | $\text{Var} \left[ \frac{\partial f}{\partial \theta_j} \right] \geq O(\frac{1}{\text{poly}(N)})$ |
| Arbitrary circuit \cite{27} | Circuit distribution | $\text{Var} \left[ \frac{\partial f}{\partial \theta_j} \right] \leq 2^{-O(N)} + O(\varepsilon^2)$ |
| D-layer noisy circuit \cite{30} | -                   | $\text{Var} \left[ \frac{\partial f}{\partial \theta_j} \right] \leq O(N^{1/2}q^D)$ |

The gradient phenomenon is inherently related to the concentration nature of measure for quantum states \cite{35}, which cannot be solved using gradient free \cite{36} or higher order \cite{37} methods. We summarize existing theoretical results about barren plateaus problem in Table I.

Recently, some techniques have been proposed to address the barren plateaus problem, including the special initialization strategy \cite{38} and the layerwise training method \cite{39}. We remark that these techniques rely on the assumption of low-depth quantum circuits. Specifically, Ref. \cite{38} initializes parameters such that the initial quantum circuit is equivalent to an identity matrix ($D = 0$). Ref. \cite{39} trains subsets of parameters for each layer, so that a low-depth circuit is optimized during the training.

Notice that almost all the theoretical analysis on the trainability of QNNs is based on the assumption that the learning circuits satisfy the unitary 2-designs or the Haar distributions. However, QNNs in most scenarios are tuned in the parameter space consisting of rotation angles followed by CNOT or CZ gates, instead of the entire unitary space \cite{35}. Thus, earlier claims that deep QNNs are doomed to fail are not on solid ground. In the work, we provide a first analysis of the gradient behaviour of the quantum neural network within the parameter space. Most importantly, we theoretically prove the existence of deep QNNs whose gradient norm is independent of the circuit depth $D$.

The rest of this paper is organized as follows. Preliminaries of QNNs are introduced in Section II. The theoretical results are presented in Section III. We show numerical results in Section IV which includes finding the ground state of the Ising model (Section IV A) and the binary classification (Section IV B). We summarize our conclusion in Section V.

II. PRELIMINARY

Quantum neural networks, also known as variational quantum algorithms, perform the optimization task which involves a parameterized quantum circuit. More specifically, we aim to minimize the function $f$ with respect to $\theta$.

$$f(\theta; \rho_{in}) = \text{Tr} \left[ O V(\theta) \rho_{in} V(\theta)^\dagger \right],$$

where $O$ denotes the quantum observable and $\rho_{in}$ denotes the density matrix of the input state. We use the linear combination of tensor products of Pauli matrices

$$\{\sigma_0, \sigma_1, \sigma_2, \sigma_3\} = \{I, X, Y, Z\}$$

to represent the quantum observable. Equation (1) provides a general formulation of the loss in QNNs. For the quantum machine learning (QML) scenario, $\rho_{in}$ encodes the information of the training data, while the observable $O$ could be a simple $\sigma_3$ for classification tasks \cite{19, 41} or a complex form for quantum kernel methods \cite{42, 43}. For the variational quantum eigensolver scenario, such as the quantum chemistry and the quantum simulation, $\rho_{in}$ is usually initialized as $(|0\rangle|0\rangle)^{\otimes N}$, while $O$ encodes the Hamiltonian of the whole system which is written as the linear combination of Pauli tensor products. For convenience, we define the locality of a Pauli product observable as the number of qubits such that the corresponding component in the observable is not the identity.

Different from the input state $\rho_{in}$ and the observable $O$, parameters $\theta$ in QNN are trainable, which makes the QNN suitable for various optimizers \cite{44, 45}. Generally, we could deploy the parameter as the phase of the time evolution with arbitrary Hamiltonians. However, a practical tactic is to encode each parameter as the phase of single qubit gates $\{e^{-i\theta_k\sigma_k}, k \in \{1, 2, 3\}\}$ while employing two-qubit gates $\{\text{CX}, \text{CZ}\}$ among them to generate quantum entanglement. This strategy has been frequently used in existing quantum neural networks \cite{9, 19, 21, 47, 50} since the model suits near-term quantum computers. We remark that the partial derivative could be calculated subsequently by using the parameter shifting rule \cite{22, 23},

$$\frac{\partial f}{\partial \theta_j} = f(\theta_+; \rho_{in}) - f(\theta_-; \rho_{in}),$$

where $\theta_+$ and $\theta_-$ are different from $\theta$ only at the $j$th parameter: $\theta_j \rightarrow \theta_j \pm \frac{\pi}{4}$. Thus, the gradient of $f$ could be obtained by estimating quantum observables, which allows the utilization of quantum computers for fast optimizations using gradient-based methods \cite{44, 45}.
QNNs) with controlled-layer (CL) architectures and the dark blue block, and a parameterized unitary operation $W_{\ell}$ on the remaining $N-S$ qubits (light blue block). We allow arbitrary parameterized unitaries the qubit and $\ell$ $j$ sequence to implement arbitrary unitaries in $U_S$ remaining $N_S$ parameterized quantum gates on the first $L$ pairs are allowed, is applied. Then we perform parameterized unitaries in the $\ell$-th CL block by $\theta_\ell = (\theta_{1,\ell}, \cdots, \theta_{S,\ell}, \theta'_\ell)$, where $\theta'_\ell$ denotes the parameters on the remaining $N - S$ qubits.

III. MAIN RESULTS

In this section, we introduce quantum neural networks (QNNs) with controlled-layer (CL) architectures and the corresponding trainability results. Namely, we prove that the expectation of the square of the gradient for the CL-QNN is shown in Figure 1. The circuit is the analogy of arbitrary deep CL-QNNs with limited CL blocks. In the beginning with an input state $\rho_{in}$, followed by $L$ layers of CL blocks. In the $\ell$-th CL block, a CZ operator layer (orange color) and the arbitrary parameterized unitary operation $W_{\ell}'(\theta'_\ell)$ (light yellow block) consists of a CZ operation layer (orange block), single qubit rotations $R_X R_Y R_X$ on the first $S$ qubits, and a parameterized unitary operation $W_{\ell}(\theta_{\ell})$ on the remaining $N-S$ qubits (light blue block). Operations $W_{\ell}$ and $W_{\ell}'(\theta'_\ell)$ could have different structures for different blocks, and the ansatz $W_{\ell}(\theta_{\ell})$ could be arbitrarily deep. We denote parameters of single qubit rotations on the $j$th qubit in the $\ell$th CL block by $\theta_{\ell,j} = (\theta_{1,\ell,j}, \theta_{2,\ell,j}, \cdots, \theta_{S,\ell,j})$. We denote parameters in the $\ell$-th CL block by $\theta_\ell = (\theta_{1,\ell}, \cdots, \theta_{S,\ell}, \theta'_\ell)$, where $\theta'_\ell$ denotes the parameters on the remaining $N - S$ qubits.

A. Controlled-layer QNNs

Now we discuss the CL-QNNs in detail. The architecture of the CL-QNN is shown in Figure 1. The circuit begins with an input state $\rho_{in}$, followed by $L$ layers of CL blocks. In the $\ell$-th CL block, a CZ operator layer (denoted as $CZ_{\ell}$), where CZ gates on arbitrary qubit pairs are allowed, is applied. Then we perform parameterized quantum gates on the first $S$ qubits and the remaining $N - S$ qubits separately. For each of the first $S$ qubits, we employ $R_X(\theta_{j,\ell}^{(3)}) R_Y(\theta_{j,\ell}^{(2)}) R_X(\theta_{j,\ell}^{(1)})$ gates sequence to implement arbitrary unitaries in $U(2)$ individually, where $j \in \{1, \cdots, S\}$ denotes the index of the qubit and $\ell \in \{1, \cdots, L\}$ denotes the index of the CL block. We allow arbitrary parameterized unitaries $W_{\ell}'(\theta'_\ell) \in U(2^{N - S})$ on the remaining $N - S$ qubits. Thus, the $\ell$-th CL block is defined by

$$V_{\ell}(\theta_\ell) = \left( \bigotimes_{j=1}^{S} W_{j,\ell}(\theta_{j,\ell}) \right) \bigotimes W_{\ell}'(\theta'_\ell) \right) CZ_{\ell},$$

where $W_{j,\ell}(\theta_{j,\ell}) \equiv R_X(\theta_{j,\ell}^{(3)}) R_Y(\theta_{j,\ell}^{(2)}) R_X(\theta_{j,\ell}^{(1)})$ and $\theta_\ell = (\theta_{1,\ell}, \cdots, \theta_{S,\ell}, \theta'_\ell)$. After all unitary operations, we measure the first $S$ qubits to reveal the information from the quantum circuit. Specifically, the objective function is the expectation of the measurement result with the associated $S$-local observable

$$\sigma_i = \sigma(\iota_1, \cdots, \iota_S, 0, \cdots, 0)$$

$$= \sigma_{i_1} \otimes \cdots \otimes \sigma_{i_S} \otimes \sigma_0 \otimes \cdots \otimes \sigma_0,$$

where $i_j \neq 0, \forall j \in \{1, \cdots, S\}$.

We remark that the controlled-layer architecture shown in Figure 1 is a general framework. Arbitrary structures could be employed for operations $CZ_{\ell}$ and $W_{\ell}'$, to generate different distributions for specific tasks. Many existing quantum neural networks can be viewed as special cases of CL-QNNs. For example, the layerwise learning circuit can be employed for operations $CZ_{\ell}$ and $W_{\ell}'$, to generate different distributions for specific tasks. Many existing quantum neural networks can be viewed as special cases of CL-QNNs. For example, the layerwise learning circuit could be arbitrarily deep. We denote the local quantum observable condition is naturally satisfied for QNNs associated with quantum machine learning, combinatorial optimizations, and quantum simulations.
B. Trainability of CL-QNNs

We provide main theoretical results of this work in Theorems 1 and 2. Specifically, we prove that both the square of the loss $f$ and the norm of its gradient are lower bounded by $\Omega(8^{-LS})$. Both bounds are independent of the qubit number $N$ and the circuit depth $D$, where the latter can be arbitrarily large when employing complex unitaries $W_i$ in Figure 1. Proofs of Theorems 1 and 2 are provided in Appendices B and C, respectively.

Theorem 1. Consider the $N$-qubit $L$-block variational quantum circuit $V(\theta)$ defined in Figure 7 and the cost function $f(\theta) = \text{Tr} \left[ \sigma_i V(\theta) \rho_{in} V(\theta)^\dagger \right]$, where $\sigma_i$ is a S-local observable. Then the following formula holds,

$$E_{\theta} f^2 \geq \frac{(\text{Tr}[\sigma_3|i\rho_{in}])^2}{8LS},$$

where the index $3|i = (3,3,\cdots,3,0,\cdots,0)$ contains $S$ non-zero components. The expectation is taken independently for all parameters in $\theta$ with the uniform distribution in $[0,2\pi]$.

Theorem 2. Consider the $N$-qubit $L$-block variational quantum circuit $V(\theta)$ defined in Figure 7 and the cost function $f(\theta) = \text{Tr} \left[ \sigma_i V(\theta) \rho_{in} V(\theta)^\dagger \right]$, where $\sigma_i$ is a $S$-local observable. Then the following formula holds,

$$E_{\theta} \|\nabla_{\theta} f\|^2 \geq \frac{12(L-1)S}{8LS} \left(\text{Tr}[\sigma_3|i\rho_{in}]\right)^2,$$

where index $3|i = (3,3,\cdots,3,0,\cdots,0)$ contains $S$ non-zero components. The expectation is taken independently for all parameters in $\theta$ with the uniform distribution in $[0,2\pi]$.

We remark that the trace term in Eqs. (4) and (5) is independent from $N$ for many input states, e.g., $\text{Tr}[\sigma_3|i(0)(0)^{\otimes N}] = 1$.

From the geographic view, the value $E_{\theta} \|\nabla_{\theta} f\|^2$ characterizes the global steepness of the function surface in the parameter space. Optimizing the objective function $f$ using gradient-based methods could be hard if the norm of the gradient vanishes to zero. Thus, the lower bound in Eq. (6) provides a theoretical guarantee for the optimization of the objective function, which is the necessary condition for obtaining a good trainability of QNNs on related tasks. Different from existing works [24, 28, 29, 38] that deal with shallow quantum circuits, we provide a positive result on the trainability of deep quantum circuit with certain structures.

From the technical view, we provide a new theoretical framework for proving Eqs. (4) and (5). Instead of the unitary 2-design assumptions which do not accord with practical QNNs training scenarios, we consider the uniform distribution in the parameter space, in which each parameter in $\theta$ varies continuously in $[0,2\pi]$. Our assumption suits quantum circuits that encode parameters in the phase of single qubit rotations, and fits the analysis to the loss landscape within the parameter space. Our framework could be extended with further assumptions on current circuit architectures in future works.

Generally, the uniform distribution of parameters could

FIG. 2. Existing QNNs as special cases of CL-QNNs. Figure 2(a) shows the structure of the layerwise learning circuit [39]. Figure 2(b) shows the structure of the quantum entanglement circuit [51-53]. Figure 2(c) shows the structure of the alternating-layered circuit [39]. The deep blue block denotes the single qubit rotations and the orange block denotes the CZ operator layer, as defined in Figure 1.

FIG. 3. Bloch sphere of states $U(0)$, where $U = R_Y(\theta_2)R_Z(\theta_1)$ for Figure 3(a) and $U \in U(2)$ for Figure 3(b). For the left figure, we sample from the uniform distribution of $\theta_2, \theta_1 \in [0, 2\pi]^2$; while for the right figure, we sample from the Haar distribution of the unitary space. Each figure contains 10000 samples.
induce a distribution that differs from the Haar distribution \([0, 2\pi]\), as illustrated by an example in Figure 3. Specifically, Figure 3(a) corresponds to the unitary \(U = \text{R}_Y(\theta_2)\text{R}_X(\theta_1)\), where \(\theta_1, \theta_2\) are sampled from \([0, 2\pi]\) uniformly; while Figure 3(b) corresponds to the unitary \(U\), which is sampled from the Haar distribution. We can see concentrations of samples around states \(|i\pm\rangle = \frac{|0\rangle \pm |1\rangle}{\sqrt{2}}\) in Figure 3(a), which differs from uniformly distributed samples in Figure 3(b).

**IV. APPLICATIONS**

We provide numerical analysis on the trainability of different QNNs using the PennyLane Python package [57]. We propose two optimization tasks, quantum simulation of the Ising model and the supervised learning for the binary classification. In the first task, we demonstrate that the CL-QNNs could perform better than the randomly structured QNNs. Moreover, in the second task of binary classification, we are able to show that only CL-QNNs with sufficiently deep circuits are trainable while the hardware-efficient QNNs [58] and the randomly structured QNNs are not.

**A. Ising model**

In the first task, we aim to find the ground state energy of the transverse-field Ising model [59] around the critical point with periodic boundary conditions, i.e.

\[
H = -\frac{1}{N} \sum_{i=1}^{N} Z_i Z_{i+1} - \frac{1}{N} \sum_{i=1}^{N} X_i,
\]

where we employ the periodic boundary condition \(Z_{N+1} = Z_0\). We adopt the loss function defined by Eq. (1) with the corresponding observable in Eq. (6). We compare the performance of CL-QNNs with that of randomly structured QNNs. Both QNNs contain \(N = 16\) qubits and begin from the initial state \((|0\rangle |0\rangle)^{\otimes N}\). We set the number of CL blocks \(L = 6\) and perform CZ gates on all neighboring

![Figure 4](image-url)
qubits $(1,2),(2,3),\cdots,(N-1,N),(N,1)$. Unitaries on the remaining $N-S$ qubits in Figure 1 are employed as the tensor product of $R_X R_Y R_X$ gate sequences. Overall, we generate a CL-QNN with 270 single qubit gates and 75 CZ gates. To make a fair comparison, the number of single qubit gates and CZ gates in the random QNN are set to be the same as that in the CL-QNN. Initial parameters in both QNNs are sampled independently from a uniform distribution in $[0,2\pi]$. We employ two different optimizers, i.e., stochastic gradient descent (SGD) [60] and stochastic Adam (SGD with adaptive momentum) [61], where the randomness is introduced by finite measurements. In the experiment, we set the number of measurements to be 100 for each Pauli observable, and the number of training iterations is 200. The learning rate is 0.15 for SGD, while for stochastic Adam, the learning rate is 0.01.

The numerical results of the Ising model task are shown in Figures 4(a) and 4(d). The loss during the training iteration is illustrated in Figures 4(a) and 4(b) for different QNNs. For the SGD case, we notice the convergence of the loss after 75 iterations, while the loss of random-QNN does not converge and the final value is much higher than that of the CL-QNN. For the stochastic Adam case, the loss of the CL-QNN converges after 75 iterations, while the loss of random-QNN does not converge with a much higher final value. We record the $\ell_2$-norm of the gradient during the training in Figures 4(c) and 4(d). The gradient norm for the CL-QNN is larger than 0.2 at the initial point and converges during training. On the other hand, the gradient norm of the random-QNN is much smaller than that of the CL-QNN at the initial point and then exceeds the CL-QNN during training. The gradient norm of the random-QNN after training indicates that the loss of the random-QNN converges to its stationary point slower than that of the CL-QNN.

### B. Binary classification

In the second task, we aim to classify two kinds of wines based on their features. The wine dataset [62] consists of 138 instances derived from three different cultivars. Each of the instances has 13 features which measure different chemical components. In this task, we choose two kinds of wines and divide the origin dataset into the training and test set. Both datasets contain 58 samples, and the number of samples from two different classes are the same. Specifically, we denote $A = \{(x^{(i)},y^{(i)})\}$ as the set of input data, where $y^{(i)} = \pm 1$ for different labels. Qubit embedding [63] is employed to encode the information of datasets into quantum states, i.e.,

$$
\rho_{\text{in}}(\mathbf{x}) = \bigotimes_{j=1}^{\dim(\mathbf{x})} R_Y(x_j)|0\rangle\langle 0|R_Y(x_j)^{\dagger},
$$

where $\dim(\mathbf{x}) = 13$ is the dimension of $\mathbf{x}$. We optimize the $\ell_1$-norm loss defined by

$$
f_{\text{QML}}(\theta) = \frac{1}{|A|} \sum_{i=1}^{|A|} |\text{Tr}(OV(\theta)\rho_{\text{in}}(x^{(i)})V(\theta)^{\dagger}) - y^{(i)}|,
$$

where $O = \sigma_3 \otimes \sigma_0 \cdots \otimes \sigma_0$, and $|A|$ is the size of the dataset. Since the aim of the binary classification is to arrange the right label for the unseen data, we will record the classification error

$$
e(\theta) = \frac{1}{|A|} \sum_{i=1}^{|A|} \text{sign}\left(\text{Tr}(OV(\theta)\rho_{\text{in}}(x^{(i)})V(\theta)^{\dagger}) - y^{(i)}\right)
$$

on the test dataset.

We compare the performance of CL-QNNs with that of hardware-efficient QNNs (HE-QNNs) in Figure 5 and randomly structured QNNs. The qubit number $N = 13$ suffices to encode all the features of the wine dataset. For CL-QNNs, we set the number of CL blocks $L = 4$ and perform CZ gates on all neighboring qubits $(1,2),(2,3),\cdots,(N-1,N),(N,1)$. The number $S = 1$ by considering the observable in Eq. (7). Unitaries on the remaining $N-S$ qubits (the light blue block in Figure 1) are chosen to be the hardware-efficient ansatz in Figure 5 with the depth $L_{\text{HE}} = 5$. Thus, the depth of the parameterized part of the CL-QNN is $D_{\text{CL}} = 3L_{\text{HE}}L = 60$. To make a fair comparison, the number of single qubit gates, CZ gates, and CNOT gates in random QNNs are set to be the same as that in the CL-QNN, and the parameterized circuit depth of the HE-QNNs is set to be the same as that in the CL-QNN, i.e., $L_{\text{HE}} = 60$. Parameters in different QNNs are initialized independently from a uniform distribution in $[0,2\pi]$. We employ SGD and stochastic Adam optimizers, where the randomness is introduced by finite measurements and the batch training strategy. We set the number of measurements to be 100 and the

![FIG. 5. The hardware-efficient circuit [63] employed in the binary classification task, where the ansatz in the light blue block is repeated for $L_{\text{HE}}$ times. The deep blue block denotes the $R_X R_Y R_X$ sequence as defined in Figure 1.](image)
FIG. 6. Numerical results of the binary classification on the wine dataset [62]. Figures 6(a) and 6(b) show the loss (7) of the training set with stochastic gradient descent and stochastic Adam optimizers. Figures 6(c) and 6(d) show the error (8) of the test set with stochastic gradient descent and stochastic Adam optimizers. Figures 6(e) and 6(f) show the $\ell_2$-norm of the corresponding gradient during the training. Red, blue, and black lines denote the average of 5 rounds for CL-QNN, the Random-QNN, and the hardware-efficient QNN (HE-QNN), respectively.

batch size to be 8. The number of training iterations is 200. The learning rates of SGD and stochastic Adam optimizers are 0.01.

The numerical results of the binary classification task are shown in Figures 6(a)-6(f). For the SGD case, we notice the convergence of the training loss and the test error corresponding to the CL-QNN after 150 iterations, while the loss of the random-QNN and the HE-QNN does not show clear convergence and the loss value is much higher than that of the CL-QNN. For the stochastic Adam
case, the training loss and the test error of the CL-QNN converges after 75 iterations, while the loss of random-QNN and HE-QNN do not show clear convergence with a much higher final value. We record the $\ell_2$-norm of the gradient during the training in Figures 6(e) and 6(f). The gradient norm for the CL-QNN is larger than 1.0 at the initial point and converges during training. On the other hand, the gradient norm of the random-QNN and the HE-QNN are much smaller than that of the CL-QNN at the initial point before the latter converges.

In conclusion, CL-QNNs of moderate size showed much better trainability for finding the ground state of the Ising model and the binary classification when compared with random-QNNs and HE-QNNs. These numerical experiments accord with the theorems obtained in the previous section.

V. CONCLUSION

In this work, we analyze the vanishing gradient problem in quantum neural networks. We prove that the gradient norm of $N$-qubit quantum neural networks with a controlled-layer structure is lower bounded by $\Omega(8^{-L^2})$, where $L$ is the number of CL blocks in the circuit and $S$ is the number of qubits corresponding to the quantum observable. We remark that the circuit depth in each CL block could be arbitrary. Thus, the bound guarantees the trainability of deep CL-QNNs.

Our theoretical framework does not require the unitary 2-design assumption as in previous works; hence, it will be more applicable to real-world quantum neural networks for near-term quantum computers. When compared with randomly structured QNNs and hardware-efficient QNNs, which are known to suffer from the barren plateaus problem, CL-QNNs show better trainability for finding the ground state of the Ising model and the binary classification task. We hope that the paper could inspire future works on the trainability of deep QNNs with different architectures and other variational quantum algorithms.

[1] Robert Hecht-Nielsen. Theory of the backpropagation neural network. In Neural networks for perception, pages 65–93. Elsevier, 1992.
[2] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. Nature, 521(7553):436–444, 2015.
[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems, volume 25. Curran Associates, Inc., 2012.
[4] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. Recurrent neural network regularization. arXiv:1409.2329.
[5] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. IEEE Transactions on Neural Networks, 20(1):61–80, 2009.
[6] Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. Neural architecture search: a survey. Journal of Machine Learning Research, 20(55):1–21, 2019.
[7] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. Nature, 549(7671):195–202, 2017.
[8] Vojtěch Havlíček, Antonio D Córcoles, Kristan Temme, Jerry M Chow, and Jay M Gambetta. Supervised learning with quantum-enhanced feature spaces. Nature, 567(7747):209–212, 2019.
[9] Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. 4(4):043001, 2019.
[10] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M Chow, and Jay M Gambetta. Hardware-efficient variational eigensolver for small molecules and quantum magnets. Nature, 549(7671):242–246, 2017.
[11] Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. Quantum approximate optimization algorithm: performance, mechanism, and implementation on near-term devices. Phys. Rev. X, 10:021067, 2020.
[12] Juneseo Lee, Alicia B. Magann, Herschel A. Rabitz, and Christian Arenz. Progress toward favorable landscapes in quantum combinatorial optimization. Phys. Rev. A, 104:032401, 2021.
[13] Yuxuan Du, Min-Hsiu Hsieh, Tongliang Liu, and Dacheng Tao. Expressive power of parametrized quantum circuits. Phys. Rev. Research, 2:033125, 2020.
[14] Dan Shepherd and Michael J Bremer. Temporally unstructured quantum computation. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 465(2105):1413–1439, 2009.
[15] Sergey Bravyi, David Gosset, and Robert König. Quantum advantage with shallow circuits. Science, 362(6412):308–311, 2018.
[16] Yuxuan Du, Min-Hsiu Hsieh, Tongliang Liu, Dacheng Tao, and Nana Liu. Quantum noise protects quantum classifiers against adversaries. Phys. Rev. Research, 3:023153, 2021.
[17] John Preskill. Quantum Computing in the NISQ era and beyond. Quantum, 2:79, 2018.
[18] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. Nature, 574(7779):505–510, 2019.
[19] Maria Schuld, Alex Bocharov, Krysta M. Svore, and Nathan Wiebe. Circuit-centric quantum classifiers. Phys. Rev. A, 101:032308, 2020.
[20] Kerstin Beer, Dmytro Bondarenko, Terry Farrelly, Tobias J Osborne, Robert Salzmann, Daniel Scheirermann, and Ramona Wolf. Training deep quantum neural networks. Nat. Commun., 11(1):1–6, 2020.
[21] He-Liang Huang, Yuxuan Du, Ming Gong, Youwei Zhao, Yulin Wu, Chaoyue Wang, Shaowei Li, Futian Liang, Jin Lin, Yu Xu, Rui Yang, Tongliang Liu, Min-Hsiu Hsieh, Hui Deng, Hao Rong, Cheng-Zhi Peng, Chao-Yang Lu, Yu-Ao Chen, Dacheng Tao, Xiaobo Zhu, and Jian-Wei Pan. Experimental quantum generative adversarial networks for image generation. *Phys. Rev. Applied*, 16:024051, 2021.

[22] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. Quantum circuit learning. *Phys. Rev. A*, 98:032309, 2018.

[23] Jun Li, Xiaodong Yang, Xinhua Peng, and Chang-Fu Sun. Hybrid quantum-classical approach to quantum optimal control. *Phys. Rev. Lett.*, 118:150503, 2017.

[24] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nat. Commun.*, 9(1):1–6, 2018.

[25] Aram W Harrow and Richard A Low. Random quantum circuits are approximate 2-designs. *Commun. Math. Phys.*, 291(1):257–302, 2009.

[26] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *Proceedings of the 35th International Conference on Machine Learning*. Proceedings of Machine Learning Research, 2018.

[27] Zoë Holmes, Kunal Sharma, M Cerezo, and Patrick J. Coles. Connecting ansatz expressibility to gradient magnitudes and barren plateaus. *arXiv:2101.02138*, 2021.

[28] Marco Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J. Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nat. Commun.*, 12(1):1–12, 2021.

[29] Arthur Pesah, M. Cerezo, Samson Wang, Tyler Volkoff, Andrew T. Sornborger, and Patrick J. Coles. Absence of barren plateaus in quantum convolutional neural networks. *Phys. Rev. X*, 11:041011, 2021.

[30] Samson Wang, Enrico Fontana, Marco Cerezo, Kunal Sharma, Akira Sone, Lukasz Cincio, and Patrick Coles. Noise-induced barren plateaus in variational quantum algorithms. *Nat. Commun.*, 2021.

[31] A V Uvarov and J D Biamonte. On barren plateaus and cost function locality in variational quantum algorithms. 54(24):245301, may 2021.

[32] Kaining Zhang, Min-Hsiu Hsieh, Liu Liu, and Dacheng Tao. Toward trainability of quantum neural networks. *arXiv:2011.06258*, 2020.

[33] Carlos Ortiz Marrero, Mária Kieferová, and Nathan Wiebe. Entanglement-induced barren plateaus. *PRX Quantum*, 2:040316, Oct 2021.

[34] Zoë Holmes, Andrew Arrasmith, Bin Yan, Patrick J. Coles, Andreas Albrecht, and Andrew T. Sornborger. Barren plateaus preclude learning scramblers. *Phys. Rev. Lett.*, 126:190501, may 2021.

[35] Markus P Müller, David Gross, and Jens Eisert. Concentration of measure for quantum states with a fixed expectation value. *Commun. Math. Phys.*, 303(3):785–824, 2011.

[36] Andrew Arrasmith, M. Cerezo, Piotr Czarnik, Lukasz Cincio, and Patrick J. Coles. Effect of barren plateaus on gradient-free optimization. *Quantum*, 5:558, 2021.

[37] M Cerezo and Patrick J Coles. Higher order derivatives of quantum neural networks with barren plateaus. 6(3):035006, 2021.

[38] Edward Grant, Leonard Wossnig, Mateusz Ostaszewski, and Marcello Benedetti. An initialization strategy for addressing barren plateaus in parametrized quantum circuits. *Quantum*, 3:214, 2019.

[39] Andrea Skolik, Jarrod R McClean, Masoud Mohseni, Patrick van der Smagt, and Martin Leib. Layerwise learning for quantum neural networks. *Quantum Mach. Intell.*, 3(1):1–11, 2021.

[40] Although the assumption of a uniform distribution in the parameter space could generate approximate 2-designs for certain cases [43] with linear depths, the corresponding analysis requires that all gates are tunable, which does not hold for circuits employing CNOT or CZ gates.

[41] Edward Grant, Marcello Benedetti, Shuxiang Cao, Andrew Hallam, Joshua Lockhart, Vid Stojanovic, Andrew G. Green, and Simone Severini. Hierarchical quantum classifiers. *npj Quantum Inf.*, 4(1), 2018.

[42] Hsin-Yuan Huang, Michael Broughton, Masoud Mohseni, Ryan Babbush, Sergio Boixo, Hartmut Neven, and Jarrod R McClean. Power of data in quantum machine learning. *Nat. Commun.*, 12(1):1–9, 2021.

[43] Xinbiao Wang, Yuxuan Du, Yong Luo, and Dacheng Tao. Towards understanding the power of quantum kernels in the NISQ era. *Quantum*, 5:331, 2021.

[44] D. Zhu, N. M. Linke, M. Benedetti, K. A. Landsman, N. H. Nguyen, C. H. Alderate, A. Perdomo-Ortiz, N. Korda, A. Garfoot, C. Brecque, L. Egan, O. Perdomo, and C. Monroe. Training of quantum circuits on a hybrid quantum computer. *Sci. Adv.*, 5(10), 2019.

[45] James Stokes, Josh Izac, Nathan Killoran, and Giuseppe Carleo. Quantum Natural Gradient. *Quantum*, 4:269, 2020.

[46] Ryan Sveke, Frederik Wilde, Johannes Meyer, Maria Schuld, Paul K. Faehrmann, Barthélémy Meynard-Piganeau, and Jens Eisert. Stochastic gradient descent for hybrid quantum-classical optimization. *Quantum*, 4:314, 2020.

[47] Yuxuan Du, Min-Hsiu Hsieh, Tongliang Liu, Shan You, and Dacheng Tao. Learnability of quantum neural networks. *PRX Quantum*, 2:040337, 2021.

[48] Yuxuan Du, Tao Huang, Shan You, Min-Hsiu Hsieh, and Dacheng Tao. Quantum circuit architecture search: error mitigation and trainability enhancement for variational quantum solvers. *arXiv:2010.10217*, 2020.

[49] Yuxuan Du and Dacheng Tao. On exploring practical potentials of quantum auto-encoder with advantages. *arXiv:2106.15334*, 2021.

[50] Yuxuan Du, Yang Qian, and Dacheng Tao. Accelerating variational quantum algorithms with multiple quantum processors. *arXiv:2106.12819*, 2021.

[51] Chih-Chieh Chen, Masaya Watabe, Kodai Shiba, Masaru Sugiura, Katsuyoshi Sakamoto, and Tomohiro Sogabe. On the expressibility and overfitting of quantum circuit learning. *ACM Transactions on Quantum Computing*, 2(2), 2021.

[52] Tóibíás Háug, Kishor Bharti, and MS Kim. Capacity and quantum geometry of parametrized quantum circuits. *arXiv:2102.01659*, 2021.

[53] Tóibíás Háug and MS Kim. Optimal training of variational quantum algorithms without barren plateaus. *arXiv:2104.14543*, 2021.

[54] Alba Cervera-Lierta. Exact Ising model simulation on a quantum computer. *Quantum*, 2:114, 2018.

[55] Guido Pagano, Aniruddha Bapat, Patrick Becker, Katherine S. Collins, Arinjoy De, Paul W. Hess, Harvey B.
Appendix A: Technical Lemmas

In this section we provide some technical lemmas. Consider the objective function of the QNN defined as
$$ f(\theta) = \langle O \rangle = \text{Tr}[(O \cdot V(\theta))\rho_n V(\theta)^\dagger], $$
where $\theta$ encodes all parameters which participate the circuit as $e^{-i\theta_j \sigma_j}$, $j \in 1, 2, 3$, $\rho_n$ denotes the input state and $O$ is an arbitrary quantum observable.

**Lemma 1.** The partial derivative of the function $f(\theta)$ with respect to the parameter $\theta_j$ could be calculated by
$$ \frac{\partial f}{\partial \theta_j} = \text{Tr}[O \cdot V(\theta_+\rho_n V(\theta_+)^\dagger)] - \text{Tr}[O \cdot V(\theta_-\rho_n V(\theta_-)^\dagger)], \tag{A1} $$
where $\theta_+ = \theta + \frac{\pi}{4} e_j$ and $\theta_- = \theta - \frac{\pi}{4} e_j$.

**Proof.** First we assume that the circuit $V(\theta)$ consists of $p$ parameters, and could be written in the sequence:
$$ V(\theta) = V_p(\theta_p) \cdot V_{p-1}(\theta_{p-1}) \cdots V_1(\theta_1), $$
such that each block $V_j$ contains only one parameter.

The parameter shifting rule [22, 23] provides a gradient calculation method for the single parameter case. Consider the observable $O' = V_{j+1}^\dagger \cdots V_3^\dagger O V_2 \cdots V_{j+1}$ and the input state $\rho'_n = V_{j-1}^\dagger \cdots V_1 \rho_n V_1^\dagger \cdots V_{j-1}^\dagger$. The gradient of $f_j(\theta_j) = \text{Tr}[O' \cdot U(\theta_j)\rho'_n U(\theta_j)^\dagger]$ could be calculated as
$$ \frac{df_j}{d\theta_j} = f_j(\theta_j + \frac{\pi}{4}) - f_j(\theta_j - \frac{\pi}{4}). $$
Thus, by inserting the form of $O'$ and $\rho'_n$, we could obtain
$$ \frac{\partial f}{\partial \theta_j} = \frac{df_j}{d\theta_j} = f_j(\theta_j + \frac{\pi}{4}) - f_j(\theta_j - \frac{\pi}{4}) = \text{Tr}[O \cdot V(\theta_+\rho_n V(\theta_+)^\dagger)] - \text{Tr}[O \cdot V(\theta_-\rho_n V(\theta_-)^\dagger)]. $$

**Lemma 2.** Let $\theta$ be a variable with uniform distribution in $[0, 2\pi]$. Let $G$ be an arbitrary hermitian unitary, and define $W = e^{-i\theta G}$. Let $O$ be an arbitrary quantum observable. Then
$$ \mathbb{E}_\theta \text{Tr}[O W \rho_1 W^\dagger] = \text{Tr}[O_1 \rho_1], $$
where $\rho_1, \rho_2$ are two quantum states, and $O_{1,2} = \frac{1}{2} O \pm \frac{i}{2} GOG$, respectively.

**Proof.** By replacing the term
$$ W = e^{-i\theta G} = I \cos \theta - iG \sin \theta,$$
we have
\[ \text{Tr}[OW\rho_1 W^\dagger] = \text{Tr}[O(I\cos \theta - iG\sin \theta)\rho_1(I\cos \theta + iG\sin \theta)] = \text{Tr}[(O_1\rho_1) + \text{Tr}(O_2\rho_1)\cos 2\theta - \sin 2\theta\text{Tr}[iO_2G\rho_1]], \]

where hermitians \( O_1 \) and \( O_2 \) are commuting and anti-commuting parts of \( O \) with respect to \( G \), respectively, i.e.
\[
O_1 + O_2 = O, \\
O_1G - GO_1 = \frac{OG + GO}{2} - \frac{GO + OG}{2} = 0, \\
O_2G + GO_2 = \frac{OG - GO}{2} + \frac{GO - OG}{2} = 0.
\]

We remark that \( iO_2G \) could serve as a hermitian observable. Similar formulation also holds for \( \rho_2 \), i.e.
\[
\text{Tr}[OW\rho_2 W^\dagger] = \text{Tr}[(O_1\rho_2) + \text{Tr}(O_2\rho_2)\cos 2\theta - \sin 2\theta\text{Tr}[iO_2G\rho_2]].
\]

Combining Eqs. (A3) and (A4), we have
\[
\mathbb{E}_\theta \left( \text{Tr}[OW_1W^\dagger]\right) = \text{Tr}[O(I\cos \theta - iG\sin \theta)\rho_1(I\cos \theta + iG\sin \theta)] = \text{Tr}[(O_1\rho_1)[(O_1\rho_1) + \text{Tr}(O_2\rho_1)\cos 2\theta - \sin 2\theta\text{Tr}[iO_2G\rho_1]]]
\]
\[
= \text{Tr}[(O_1\rho_1) + \text{Tr}(O_2\rho_1)\cos 2\theta - \sin 2\theta\text{Tr}[iO_2G\rho_1]] + \frac{1}{2}\text{Tr}[O_2\rho_1]\text{Tr}[O_2\rho_2]
\]
\[
= \frac{1}{2}\text{Tr}[O_2\rho_1]\text{Tr}[O_2\rho_2] + \frac{1}{2}\text{Tr}[iO_2G\rho_1]\text{Tr}[iO_2G\rho_2],
\]

where Eq. (A5) is obtained by calculating expectation terms.

**Lemma 3.** Let \( \theta \) be a variable with uniform distribution in \([0, 2\pi]\). Let \( G \) be an arbitrary hermitian unitary, and denote \( W_\pm = e^{-i(\theta_\pm + 2G)\theta}}. Let \( O \) be an arbitrary hermitian observable and \( \rho \). Then
\[
\mathbb{E}_\theta \left( \text{Tr}[O_W\rho W^\dagger]\right) - \text{Tr}[O_W\rho W^\dagger] = 2\text{Tr}[O_2\rho]^2 + 2\text{Tr}[iO_2G\rho]^2,
\]

where \( \rho \) is a quantum state and \( O_2 = \frac{1}{2}O - \frac{1}{2}GOG \).

**Proof.** First we denote \( \theta_\pm = \theta \pm \frac{\pi}{4} \) for convenience. By replacing terms
\[ W_\pm = e^{-i\theta_\pm G} = I\cos \theta_\pm - iG\sin \theta_\pm, \]
we obtain
\[
\text{Tr}[OW_\pm\rho W^\dagger] - \text{Tr}[OW_\pm\rho W^\dagger] = \text{Tr}[O(I\cos \theta_\pm - iG\sin \theta_\pm)\rho(I\cos \theta_\pm + iG\sin \theta_\pm)]
\]
\[
- \text{Tr}[O(I\cos \theta_\pm - iG\sin \theta_\pm)\rho(I\cos \theta_\pm + iG\sin \theta_\pm)].
\]

\[ = \text{Tr}[O\rho] + \text{Tr}[O_2\rho] - \sin 2\theta_\pm \text{Tr}[iO_2G\rho], \]

where hermitians \( O_{1,2} = \frac{1}{2}O \pm \frac{1}{2}GOG \) are commuting and anti-commuting parts of \( O \) with respect to \( G \), respectively, i.e.
\[
O_1 + O_2 = O, \\
O_1G - GO_1 = \frac{OG + GO}{2} - \frac{GO - OG}{2} = 0, \\
O_2G + GO_2 = \frac{OG - GO}{2} + \frac{GO - OG}{2} = 0.
\]

We remark that \( iO_2G \) could be served as a hermitian observable. Thus, we have
\[
\mathbb{E}_\theta \left( \text{Tr}[O_W\rho W^\dagger] - \text{Tr}[O_W\rho W^\dagger]\right) = 2\text{Tr}[O_2\rho]^2 + 2\text{Tr}[iO_2G\rho]^2,
\]

where Eq. (A7) is obtained by calculating expectation terms.

**Appendix B: Proof of Theorem 1**

*Proof.* We first introduce several notations for convenience.

![FIG. 7. Decomposition of a controlled layer block into five layers.](image)

Denote
\[ V_X(\theta_\ell^{(1)}) \equiv R_X(\theta_\ell^{(1)}) \otimes \cdots \otimes R_X(\theta_\ell^{(S)}) \otimes \sigma_0 \otimes \cdots \otimes \sigma_0, \]
\[ V_Y(\theta_\ell^{(2)}) \equiv R_Y(\theta_\ell^{(2)}) \otimes \cdots \otimes R_Y(\theta_\ell^{(S)}) \otimes \sigma_0 \otimes \cdots \otimes \sigma_0, \]
\[ V_X(\theta_\ell^{(3)}) \equiv R_X(\theta_\ell^{(3)}) \otimes \cdots \otimes R_X(\theta_\ell^{(S)}) \otimes \sigma_0 \otimes \cdots \otimes \sigma_0, \]

where \( \theta_\ell^{(i)} = (\theta_\ell^{(i,1)}, \cdots, \theta_\ell^{(i,S)}) \). Then a controlled-layer block in Figure 7 could be divided into
where the parameter \( \theta = (\theta^{(1)}, \theta^{(2)}, \theta^{(3)}). \) We abbreviate the term \( I^{\otimes S} \otimes W_{\ell}^{(i)}(\theta^{(j)}) \) as \( W_{\ell}^{(i)}(\theta^{(j)}) \) in the following for convenience.

In the following proof, we consider \( V_X(\theta^{(1)}_\ell), V_Y(\theta^{(2)}_\ell), V_Z(\theta^{(3)}_\ell), W_{\ell}^{(i)}(\theta^{(j)}) \) and CZ \( \ell \) as separate layers. We denote by \( \rho_k \ell \) the state after the \( \ell \)th gate, \( \forall k \in \{0, 1, \cdots, 5L\}, \)

\[
\rho_k \ell := \begin{cases} 
\frac{1}{\sqrt{\ell}} \sum_{i=1}^{\ell} V_i(\theta^{(1)}_\ell) \rho_{\text{in}}(i) \sum_{i=1}^{\ell} V_i(\theta^{(1)}_\ell) & (k = 5\ell), \\
CZ_{\ell+1} \rho_{5\ell} \rho_{5\ell+1} & (k = 5\ell + 1), \\
V_X(\theta^{(1)}_{\ell+1}) \rho_{5\ell+1} V_X(\theta^{(1)}_{\ell+1})^\dagger & (k = 5\ell + 2), \\
V_Y(\theta^{(2)}_{\ell+1}) \rho_{5\ell+2} V_Y(\theta^{(2)}_{\ell+1})^\dagger & (k = 5\ell + 3), \\
V_Z(\theta^{(3)}_{\ell+1}) \rho_{5\ell+3} V_Z(\theta^{(3)}_{\ell+1})^\dagger & (k = 5\ell + 4). 
\end{cases}
\]

Now we begin to prove Eq. \((\text{B}4)\). We denote all parameters in the circuit by \( \theta = (\theta^{(1)}, \theta^{(2)}, \theta^{(3)}, \theta_L) \). Rewrite the formulation of \( f(\theta) \) in detail:

\[
E_{\theta} \left( \text{Tr} \left[ \sigma_i V(\theta) \rho_{\text{in}} V(\theta)^\dagger \right] \right)^2 \geq \frac{1}{2} E_{\theta} \left( \text{Tr} \left[ \sigma_i V(\theta) \rho_{\text{in}} V(\theta)^\dagger \right] \right)^2 
\]

where \( 3|i; 2 \) denotes the index by replacing non-zero elements of \( i = (i_1, \cdots, i_N) \) with 3 if the original value is 2. Eq. \((\text{B}11)\) is derived by using Lemma \(2 \) for the RX rotation case for \( S \) times. Eqs. \((\text{B}12)\) and \((\text{B}13)\) are derived similarly by using Lemma \(2 \). Eq. \((\text{B}14)\) is derived by noticing that for \( j, k \in \{0, 3\} \), the term \( \sigma_j \otimes \sigma_k \) remains the same after the CZ operation.

Now we move forward from Eq. \((\text{B}14)\). By applying Eqs. \((\text{B}7)-(\text{B}14)\) inductively for \( L - 1 \) times, we obtain

\[
\text{Eq. (B14)} \ \geq \ \left( \frac{1}{2} \right)^{3LS} \text{Tr} \left[ \sigma_{3|i; 6\ell - 4} \right]^2. \tag{B15}
\]

Thus, we prove Equation \((4)\). \(\square\)

Appendix \(C\): Proof of Theorem \(2\)

\textbf{Proof.} First, we notice that the norm of the whole gradient is lower bounded by that of particle derivatives summed over a part of parameters, i.e.

\[
E_{\theta} \left\| \nabla_{\theta} f \right\|^2 \geq \sum_{l=1}^{L-1} \sum_{n=1}^L \sum_{j=1}^3 \frac{\left( \frac{\partial f}{\partial \theta^{(j)}_{\ell, n}} \right)^2}{E_{\theta} \left( \frac{\partial f}{\partial \theta^{(j)}_{\ell, n}} \right)^2} = 4 \left( \text{Tr} \left[ \sigma_{3|i; 6\ell} \rho_{6\ell - 4} \right]^2 \right)^2 \tag{C1}
\]

holds for all \( \ell \in \{1, \cdots, L - 1\}, n = \{1, \cdots, S\}, \) and \( j \in \{1, 2, 3\}. \)

Now we begin to prove Eq. \((C2)\). Our main idea is to integrate the square of the partial derivative of \( f \) with respect to \( \theta = (\theta_1, \cdots, \theta_L) \) by using Lemma \(2 \) and Lemma \(3 \). For convenience, we follow the definition of \( V_X(\cdot) \) in Eqs. \((\text{B}1)-(\text{B}3)\), \( V_Y(\cdot) \) in Eq. \((\text{B}2)\), CZ \( \ell \) and \( \rho_k \) in Eq. \((\text{B}5)\). In addition, we denote

\[
\rho_{k, t, n, j, z} := \begin{cases} 
\rho_k(\theta_1, \cdots) \quad (k \leq 5\ell - 4), \\
\rho_k(\theta_1, \cdots, \theta_{t-1}, \theta_{t, n, j, z}, \theta_{t+1}, \cdots) & (k > 5\ell - 4), 
\end{cases} \tag{C3}
\]

where \( \rho_k \) follows from the definition in Eq. \((\text{B}5)\), and \( \theta_{t, n, j, z} \) differs from \( \theta_t \) by plus or minus \( \frac{\pi}{4} \) on the component \( \theta^{(j)}_{\ell, n}. \)
Next, we rewrite the formulation in Eq. (C2).
\[
\mathbb{E}_\theta \left( \frac{\partial}{\partial \theta_{\ell,n}} \text{Tr} \left[ \sigma_i V(\theta) \rho_{\text{in}} V(\theta)^\dagger \right] \right)^2 \tag{C4}
\]
\[
= \mathbb{E}_\theta \cdots \mathbb{E}_\theta \left( \frac{\partial}{\partial \theta_{\ell,n}} \text{Tr} \left[ \sigma_i \rho_{\text{FL}} \right] \right)^2 \tag{C5}
\]
\[
= \mathbb{E}_\theta \cdots \mathbb{E}_\theta \left( \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,-} \right] \right)^2 \tag{C6}
\]
Eq. (C4) follows from the formulation of the cost function \( f \). Eq. (C5) follows from Eq. (B5). Eq. (C6) follows from the parameter-shift rule in Lemma 1.
We proceed from Eq. (C6). Firstly, by taking the expectation with the uniform distribution of parameters in \( \theta_{\ell} \), we obtain
\[
\mathbb{E}_\theta \left( \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,-} \right] \right)^2 = \mathbb{E}_\theta \cdots \mathbb{E}_\theta \left( \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,-} \right] \right)^2 \tag{C7}
\]
\[
= \mathbb{E}_\theta \cdots \mathbb{E}_\theta \left( \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,-} \right] \right)^2 \tag{C8}
\]
\[
= \mathbb{E}_\theta \cdots \mathbb{E}_\theta \left( \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,-} \right] \right)^2 \tag{C9}
\]
where Eq. (C7) follows from the definition of \( \rho_{k,\ell,n,\pm} \) in Eq. (C3). Eq. (C8) is obtained by using Eq. (B10). Eq. (C9) follows from the definition of \( \rho_{k,\ell,n,\pm} \) in Eq. (C3) again.
Then, we have
\[
\geq \frac{1}{2^{5S}} \mathbb{E}_{\theta_{L}^{(1)} \theta_{L}^{(2)}} \left( \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,-} \right] \right)^2 \tag{C10}
\]
\[
= \frac{1}{2^{5S}} \mathbb{E}_{\theta_{L}^{(1)} \theta_{L}^{(2)}} \left( \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,-} \right] \right)^2 \tag{C11}
\]
\[
= \frac{1}{2^{5S}} \mathbb{E}_{\theta_{L}^{(1)} \theta_{L}^{(2)}} \left( \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,-} \right] \right)^2 \tag{C12}
\]
\[
= \frac{1}{2^{5S}} \mathbb{E}_{\theta_{L}^{(1)} \theta_{L}^{(2)}} \left( \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{\text{FL},\ell,n,-} \right] \right)^2 \tag{C13}
\]
\[
\geq \frac{\mathbb{E}_\theta \cdots \mathbb{E}_\theta \left( \text{Tr} \left[ \sigma_i \rho_{5L,\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{5L,\ell,n,-} \right] \right)^2}{2^{3S}} \tag{C14}
\]
\[
= \frac{\mathbb{E}_\theta \cdots \mathbb{E}_\theta \left( \text{Tr} \left[ \sigma_i \rho_{5L,\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{5L,\ell,n,-} \right] \right)^2}{2^{3S}} \tag{C15}
\]
where \( 3i; 2 \) denotes the index by replacing non-zero elements of \( i = (i_1, \cdots, i_N) \) with 3 if the original value is 2. Eq. (C10) is derived by using Lemma 2 for the \( R_X \) case. Eq. (C11) follows from the definition of \( \rho_{k,\ell,n,\pm} \) in Eq. (C3). Eq. (C12) is obtained by using Lemma 2 for the \( R_Y \) case. Eq. (C13) follows from the definition of \( \rho_{k,\ell,n,\pm} \) in Eq. (C3). Eq. (C14) is obtained by using Lemma 2 for the \( R_X \) case. Eq. (C15) is derived by noticing that for \( j, k \in \{0, 3\} \), the term \( \sigma_j \otimes \sigma_k \) remains the same after the CZ operation.
Next, we repeat the derivation in Eqs. (C7) - (C15) inductively for parameters \( \theta_{\ell_2}, \cdots, \theta_{\ell_{\ell_1}} \), which yields
\[
\mathbb{E}_\theta \left( \frac{\partial f}{\partial \theta_{\ell,n}} \right)^2 \geq \frac{\mathbb{E}_\theta \cdots \mathbb{E}_\theta \left( \text{Tr} \left[ \sigma_i \rho_{5L,\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{5L,\ell,n,-} \right] \right)^2}{2^{3(L-\ell)S}} \tag{C16}
\]
\[
= \frac{1}{8(L-\ell)S} \mathbb{E}_{\theta_1} \cdots \mathbb{E}_{\theta_\ell} \left( \text{Tr} \left[ \sigma_i \rho_{5L,\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{5L,\ell,n,-} \right] \right)^2 \tag{C17}
\]
\[
= \frac{1}{8(L-\ell)S} \mathbb{E}_{\theta_1} \cdots \mathbb{E}_{\theta_{\ell-1}} \theta_{\ell}^{(1)} \theta_{\ell}^{(2)} \theta_{\ell}^{(3)} \left( \text{Tr} \left[ \sigma_i \rho_{5L,\ell,n,+} \right] - \text{Tr} \left[ \sigma_i \rho_{5L,\ell,n,-} \right] \right)^2 \tag{C18}
\]
Eq. (C17) is derived by using the definition of \( \rho_{k,\ell,n,\pm} \) in Eq. (C3). Eq. (C18) is derived by using Eq. (B10). Now we proceed to integrate parameter layers \( \theta_{\ell}^{(1)}, \theta_{\ell}^{(2)}, \theta_{\ell}^{(3)} \) and \( \theta_{\ell-1}, \cdots, \theta_{\ell_1} \). Remark that the parameter \( \theta_{\ell}^{(n)} \) is applied on the nth qubit with the observable \( \sigma_3 \) in Eq. (C18). Since \( \sigma_3 \) anticommutes with \( \sigma_1 \) and \( \sigma_2 \), Hamiltonians of parameterized single qubit gates in the circuit, we could apply Lemma 2 and Lemma 3 to simplify the formulation. Thus, Eq. (C18) could be further bounded
\[
\geq \frac{\mathbb{E}_\theta \cdots \mathbb{E}_\theta \left( \text{Tr} \left[ \sigma_i \rho_{5L-\ell+1} \right] \right)^2}{8(L-\ell+1)S} \tag{C19}
\]
\[
\geq \frac{4 \text{Tr} \left[ \sigma_i \rho_{5L} \right]^2}{8LS} \tag{C20}
\]
Eq. (C19) is obtained by noticing the similar collapsed formulation of Lemmas 2 and 3 when the gate Hamiltonian anticommutes with the observable. Eq. (C20) follows similar to Eq. (B15). Thus, we prove Eq. (C2).