Information Scrambling in Quantum Neural Networks

Huitao Shen,[1] Pengfei Zhang,[2, 3, 4] Yi-Zhuang You,[5] and Hui Zhai[2, *]

[1]*Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*
[2]*Institute for Advanced Study, Tsinghua University, Beijing, 100084, China*
[3]*Institute for Quantum Information and Matter, California Institute of Technology, Pasadena, California 91125, USA*
[4]*Walter Burke Institute for Theoretical Physics, California Institute of Technology, Pasadena, California 91125, USA*
[5]*Department of Physics, University of California, San Diego, California 92093, USA*

Quantum neural networks are one of the promising applications for near-term noisy intermediate-scale quantum computers. A quantum neural network distills the information from the input wavefunction into the output qubits. In this Letter, we show that this process can also be viewed from the opposite direction: the quantum information in the output qubits is scrambled into the input. This observation motivates us to use the tripartite information, a quantity recently developed to characterize information scrambling, to diagnose the training dynamics of quantum neural networks. We empirically find strong correlation between the dynamical behavior of the tripartite information and the loss function in the training process, from which we identify that the training process has two stages for randomly initialized networks. In the early stage, the network performance improves rapidly and the tripartite information increases linearly with a universal slope, meaning that the neural network becomes less scrambled than the random unitary. In the latter stage, the network performance improves slowly while the tripartite information decreases. We present evidences that the network constructs local correlations in the early stage and learns large-scale structures in the latter stage. We believe this two-stage training dynamics is universal and is applicable to a wide range of problems. Our work builds bridges between two research subjects of quantum neural networks and information scrambling, which opens up a new perspective to understand quantum neural networks.

The neural network lies at the heart of the recent blossom of deep learning [1]. Mathematically, the neural network is a trainable mapping from the input feature to the output target. The input feature is typically represented as a high-dimensional vector. The information is distilled from the input by the neural network and is encoded into a lower-dimensional output vector. Recently, the quantum generalization of neural networks have been proposed and actively studied [2–16]. In quantum neural networks, both inputs and outputs are quantum wavefunctions. The classical mapping is replaced by a quantum channel composed by unitary evolutions and measurements. The quantum channel is parameterized and trained in classical optimization loops. As a result, these quantum neural networks are also called "parameterized quantum circuits". This hybrid quantum-classical framework can process both classical and quantum data [17]. It is considered as one of the most promising applications for near-term noisy intermediate-scale quantum devices [18]. Moreover, it has been suggested that these quantum neural networks have more expressive power than their classical counterparts [14].

Similar to classical neural networks, quantum information in the input wavefunction is distilled and encoded into the output. This process is illustrated by the forward arrow in Fig. 1(a). Intriguingly, for quantum neural networks, this process can also be viewed from the opposite direction. By deferring measurements until the end of the quantum channel [19], the information encoded in the output qubits just before the measurement is spread into the entire system by unitary transformations, as illustrated by the backward arrow in Fig. 1(a). Such processes that information is scrambled from a small system to a large one is now known as the information scrambling. The subject of information scrambling is now well-studied in contexts such as thermalization, chaos and information dynamics in quantum many-body systems and even black-hole physics [20–22]. In particular, the out-of-time-order correlation function is proposed as a powerful tool to diagnose information scrambling [23–27].

Quantum neural networks and quantum information scrambling so far are two separated research topics. The purpose of this Letter is to bridge the gap and make their connection: Information encoding in a quantum neural network and the information scrambling are the same process viewed from opposite directions.

There have been information-theoretic studies of classical neural networks. For example, the mutual information between hidden layers' intermediate results and the input or the output was studied and a universal training dynamics was found [28–30]. However, in classical neural networks, the mapping at every layer is usually not invertible and the information is generally not preserved. Due to the information loss during the process, the mutual information always decreases with the network depth. In contrast, the unitarity of the quantum evolution preserves the information perfectly. The mutual information between the input and the output of any unitary transformation is always maximal. In order to obtain nontrivial diagnosis in quantum neural networks, the key is to consider the mutual information between *subsystems* of the input and the output. This naturally leads to the tripartite information—a quantity that characterizes the information scrambling [31, 32].

In this Letter, we study the training dynamics of quantum neural networks using the tripartite information. We simultaneously monitor both the network performance and the tripartite information during training and observe empirical relations between them. Based on the behavior of these two quantities, the training process can be decomposed into two

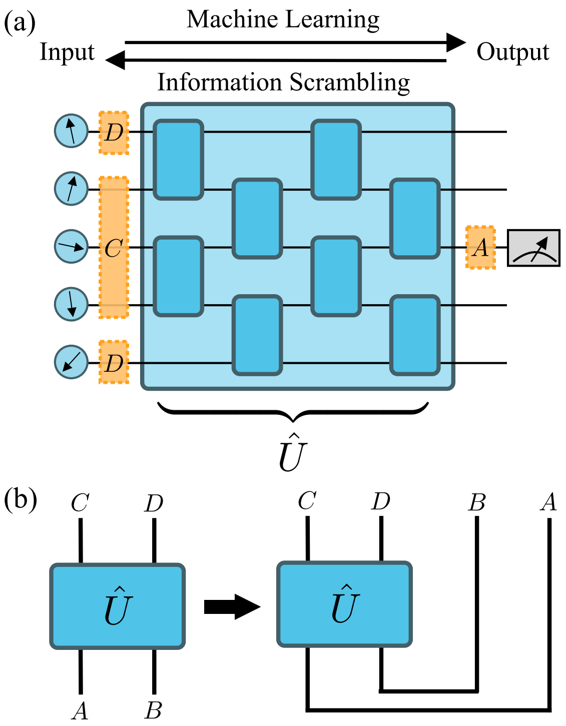

FIG. 1. (a) Schematic of a quantum circuit with brick-wall geometry. Here the network has $n = 5$ qubits and depth $l = 4$. All these two-qubit gates form a giant unitary transformation $\hat{U}$ that distills the information from the input qubits and encode it into one output qubit. The inverse process that the information of one output qubit is scrambled into input qubits by $\hat{U}^\dagger$. A is the output subsystem, C and D are input subsystems in the definition of the tripartite information. (b) Illustration for the operator-state mapping in the definition of tripartite information. Each leg may represent multiple qubits.

stages. We call the first stage as "local construction stage", and the second stage as "global relaxation stage". In the following, we present detailed analysis of the training dynamics and provide evidences to support our claim.

Tripartite Information of Quantum Neural Networks. Consider a unitary operator $\hat{U}$ in the n-qubit Hilbert space $\hat{U} = \sum_{\mathrm{i;j}=1}^{2^n} U_{\mathrm{ij}}\ |i\rangle\langle j|$, where $\{|i\rangle, i = 1, \dots, 2^{\mathrm{n}}\}$ denotes a complete set of bases in the Hilbert space. It can be regarded as a tensor with n input and n output legs. As illustrated in Fig. 1(b), we divide the output legs (qubits) to two non-overlapping subsytems A and B and similarly divide the input legs (qubits) to C and D.

The operator can be mapped to a state in the $2n$-qubit Hilbert space as $|U\rangle = \sum_{\mathrm{i;j}=1}^{2^n} U_{\mathrm{ij}}\ /\sqrt{2^{\mathrm{n}}}|i\rangle|j\rangle$. Since $|U\rangle$ is a pure state, the entanglement entropy of its subsystem is well-defined, e.g. $S(A) \equiv -\mathrm{tr}(\rho_{\mathsf{A}} \log_2 \rho_{\mathsf{A}})$ with $\rho_{\mathsf{A}} \equiv \mathrm{tr}_{\mathsf{B;C;D}}\ (|U\rangle\langle U|)$ being the reduced density matrix of subsystem A. The mutual information between the output subsystem A and the input subsystem C is $I(A, C) \equiv S(A) + S(C) - S(A \cup C)$. Similar definition can be made for $I(A, D)$ and $I(A, C \cup D)$. In this way, the tripartite information of the unitary $\hat{U}$ is defined as [31, 32]

$$I_3(A, C, D) \equiv I(A, C) + I(A, D) - I(A, C \cup D), \quad (1)$$

Because $C \cup D$ are all input qubits, it can be proved that $I(A, C \cup D) = 2|A|$, where $|A|$ is the number of qubits in subsystem A. Therefore, it is crucial to consider the mutual information between subsystems of both input and output qubits.

The strong subadditivity of the entanglement entropy leads to $I_3(A, C, D) \le 0$ for a unitary gate. The absolute value of the tripartite information $I_3(A, C, D)$ measures how much information of the subsystem A is shared by C and D simultaneously after the unitary transformation, thus quantifies how scrambled a unitary is. For example, for an identity unitary transformation $U_{\mathrm{ij}}\ = \delta_{\mathrm{ij}}$, if A is entirely contained in C or D, it is straightforward to show that $I_3(A, C, D) = 0$. As an opposite limit, for uniform Haar random unitary, local measurements can not extract any information. It follows on average $I(A, C)$ and $I(A, D)$ are exponentially small and therefore $I_3(A, C, D) = -2|A|$, which is the maximal absolute value of I_3 [32].

Having introduced the tripartite information for a general unitary transformation, we now turn to tripartite information of quantum neural networks. Here we only consider parameterized quantum circuits with brick-wall geometry. As shown in Fig. 1(a), each square represents an independent two-qubit unitary gate in the SU(4) group, and is parameterized using its 15 Euler angles [33]. During training, these parameters are optimized with classical optimization algorithms. All these two-qubit gates form a quantum circuit represented by a giant unitary transformation $\hat{U}$.

The datasets to be studied in this work have several important features. First, the input wavefunctions all have time reversal symmetry, and consequently can be represented as real vectors. Therefore we restrict two-qubit gates to SO(4) with 6 Euler angles each. Second, the output target is either a real number within $[-1, 1]$ or a binary label within $\{0, 1\}$, only one readout qubit is needed at the end of the quantum circuit. For simplicity, we always let n be odd and fix the readout qubit to be the qubit at the center, i.e. $(n + 1)/2$-th qubit.

To define tripartite information, we always fix the output subsystem A to be the central readout qubit. To respect the symmetry that A is located at the center, we always choose C to be the central $|C|$ input qubits in the circuit, and D to be the remaining input qubits. Note that under this definition, D in general contains two disconnected regions. In this way, the tripartite information $I_3(A, C, D)$ characterizes how much information of the output qubit is scrambled on the input side between the central region C and the outer region D.

Magnetization Learning. The first task is to supervisedly learn the average magnetization of a many-body wavefunction of n half spins. The dataset consists of N input-target pairs $\{(|G\ \ \rangle, M_{\mathsf{z}}\ \ ), \alpha = 1, \dots, N\}$, where the input wavefunction $|G\ \ \rangle$ is the ground state wavefunction of the parent

Hamiltonian with random long-ranged spin-spin interactions:

$$\hat{H} = \sum_{\mathsf{i;j}\ =1}^{\mathsf{n}} (J_{\mathsf{ij}}\ \sigma_{\mathsf{i}}^{\mathsf{z}} \sigma_{\mathsf{j}}^{\mathsf{z}} + K_{\mathsf{ij}}\ \sigma_{\mathsf{i}}^{\mathsf{x}} \sigma_{\mathsf{j}}^{\mathsf{x}}) + \sum_{\mathsf{i}=1}^{\mathsf{n}} (g_{\mathsf{i}} \sigma_{\mathsf{i}}^{\mathsf{x}} + h\sigma_{\mathsf{i}}^{\mathsf{z}}), \quad (2)$$

where σ_{i} represents the μ-th Pauli matrix on the i-th qubit, $\mu = x, y, z$ and $i = 1, \ldots, n$. J_{ij} , K_{ij} , g_{i} and h are all random numbers. The target is the average magnetization computed as $M_{\mathsf{z}} \equiv \langle G\ |\hat{M}_{\mathsf{z}}|G\ \rangle$, where the magnetization operator is $\hat{M}_{\mathsf{z}} \equiv \sum_{\mathsf{i}=1}^{\mathsf{n}} \sigma_{\mathsf{i}}^{\mathsf{z}}/n$. In sampling the random Hamiltonian, we ensure $J_{\mathsf{ij}} \le 0$ such that the ground state wavefunctions are either "ferromagnetic" or "paramagnetic" measured under M_{z}. h is a small pinning field randomly drawn from a distribution with zero mean, which is used to trigger the spontaneous Z_2 symmetry breaking in the ferromagnetic phase.

The quantum neural network takes the input wavefunction $|G\ \rangle$ and applies the unitary transformation $\hat{U}$ on it. The magnetization is readout by measuring σ^{x} of the central qubit. The task was such design to challenge the quantum neural network to learn how to summarize the average magnetization in the σ^{z}-basis and present the result in the σ^{x}-basis. This is essentially a regression task and the loss function to be minimized during training is the absolute error of the magnetization:

$$\mathcal{L} = \frac{1}{N} \sum_{=1}^{\mathsf{N}} \left| \langle G\ |\hat{U}^{\dagger} \sigma_{(\mathsf{n}+1)=2}^{\mathsf{x}} \hat{U}\, |G\ \rangle - M_{\mathsf{z}} \right|. \quad (3)$$

We simulate the above hybrid quantum-classical quantum neural network training algorithm. The distributions of random parameters in the Hamiltonian Eq. (2) are chosen such that M_{z} in the dataset roughly distributes uniformly within $[-1, 1]$. All two-qubit unitaries in the quantum neural network are initialized randomly. The parameters are optimized with the AMSGrad gradient descent algorithm [34]. The gradients can be computed directly thanks to the linearity of the quantum channel and are measurable in a realistic quantum neural network [7, 9, 35].

Two-stage Training. In Fig. 2, we show both the training and validation loss, along with the tripartite information, as functions of the training epoch. Both training and validation losses decrease monotonically when the training proceeds, indicating that the network can learn to compute the magnetization reasonably well without overfitting.

At the early stage of the training, the rapid improvement of the quantum neural network performance, characterized by a fast decrease of both training and validation losses, is accompanied by an almost linear increase of the tripartite information. In other words, the quantum neural network becomes less scrambled compared with the initial random unitary. This training stage terminates when the tripartite information reaches its local maximum, as indicated by the vertical dotted line in Fig. 2. In the next stage, the tripartite information decreases again, meaning that the network scrambles information faster. The network performance also improves, but with a much slower rate compared with that in the first stage.

We call the training stage before I_3 reaching the local maximum the "local construction stage", and the latter stage where I_3 decreases as the "global relaxation stage". The reason for these names will be clear after we study the training dynamics in detail below. This empirical observation that quantum neural network performance and the information scrambling is closely correlated is the main finding of this work. This correlation has been observed in most of our numerical tests with different network initializations, training algorithms, system sizes and network depths [36].

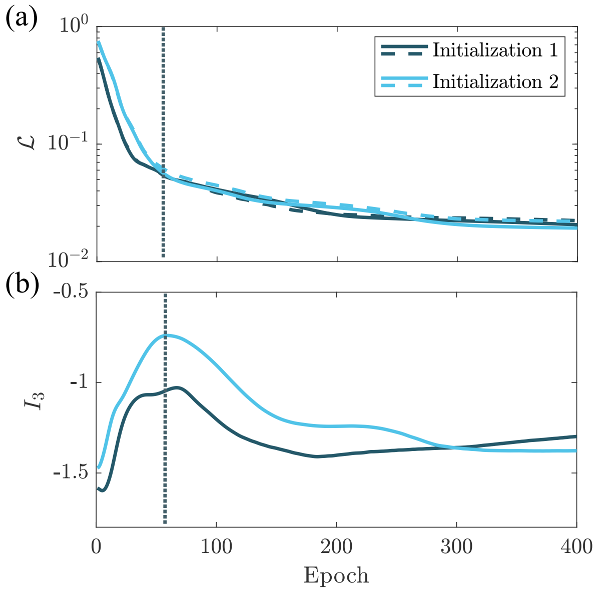

FIG. 2. (a) Training (solid) and validation (dashed) loss as functions of the training epoch. Different colors represent different random initializations. The network has $n = 9$ qubits and depth $l = 6$. The training and validation dataset contains $N = 2500$ and 500 wavefunction-magnetization pairs respectively, sampled from random Hamiltonian ensemble, where random parameters are distributed uniformly within $J_{ij}/J \in [-1, 0]$, $K_{ij}/J \in [-1, 1]$, $g_i/J \in [-3, 3]$ and $h/J \in [-0.02, 0.02]$. J is the energy unit. The learning rate is $\lambda = 10^{-2}$. (b) Tripartite information $I_3(A, C, D)$ as a function of the training epoch for different initializations. Here and in the rest of the paper the input subsystem size $|C| = 5$ unless otherwise specified. The dotted vertical line indicates the boundary between two training stages.

We also train quantum neural networks for a different task of learning the winding number of a product quantum state. Compared with the magnetization task, the input wavefunction there is a product state and is essentially classical, and the target is now a binary label instead of a real number. Despite the very different nature of this task, the empirical correlation between the neural network performance and the tripartite information still holds. All details of the winding number learning task are presented in [35].

Local Construction Stage. We claim that during the first stage when the tripartite information linearly increases, the quantum neural network learns local features of the input wavefunction. For the magnetization learning task, for example, because of the existence of ferromagnetic domain in the training wavefunction, there is some probability that any

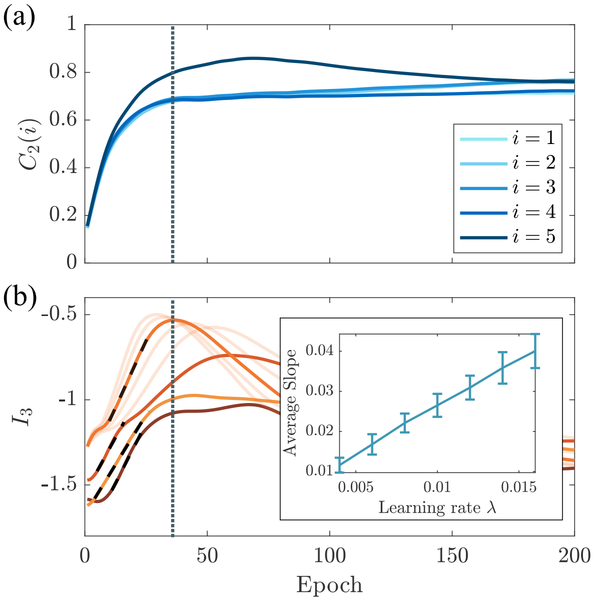

FIG. 3. (a) Two-point correlation function $C_2(i)$ as a function of the training epoch and the input qubit i. The initialization is the same as Initialization 1 in Fig. 2. (b) Tripartite information $I_3(A, C, D)$ as a function of the training epoch for different initializations and learning rates. All solid lines are trained under learning rate $\lambda = 10^{-2}$. The transparent orange lines are trained with the same initialization as the solid orange line, but with learning rates $\lambda = 6, 8, 12, 14 \times 10^{-3}$. The average slope for the four initialization shown here is plotted in the inset, as a function of the learning rates. The error bars represent $\pm\sigma$, where σ is the standard deviation of the fitted slopes for fixed learning rate but different initializations.

single spin is aligned relatively well with remaining spins in the system. Simply outputting any single-spin magnetization of the input wave function is actually a reasonable guess, the training loss can decease rapidly. For such networks where only local features are extracted, information does not need to be scrambled into the whole system. Therefore, the tripartite information increases during this stage.

To support the above claim, we compute two-point correlations between input qubits and the readout qubit:

$$C_2(i) \equiv \frac{1}{N} \sum_{=1}^{\mathsf{N}} \langle G \ | \sigma^{\mathsf{z}}_{\mathsf{i}} \hat{U}^{\dagger} \sigma^{\mathsf{x}}_{(\mathsf{n}+1)=2} \hat{U} \, |G \ \rangle \, . \tag{4}$$

If one views $\hat{U}$ as a time evolution operator, then $C_2(i)$ is simply a two-point function between two different places and two different times. In Fig. 3(a), we plot C_2 as a function of different input qubits and training epochs at early training stage. As can be seen, they increase rapidly and then saturate to large values. The increasing correlation indicates that the quantum neural network is establishing the correspondence between local input features and the output qubit. During this stage, the tripartite information also increases, and the two-point correlation function saturates when the tripartite information reaches the maximum. All these observations are consistent with our claim that during the first local construction stage, local features are extracted from the input.

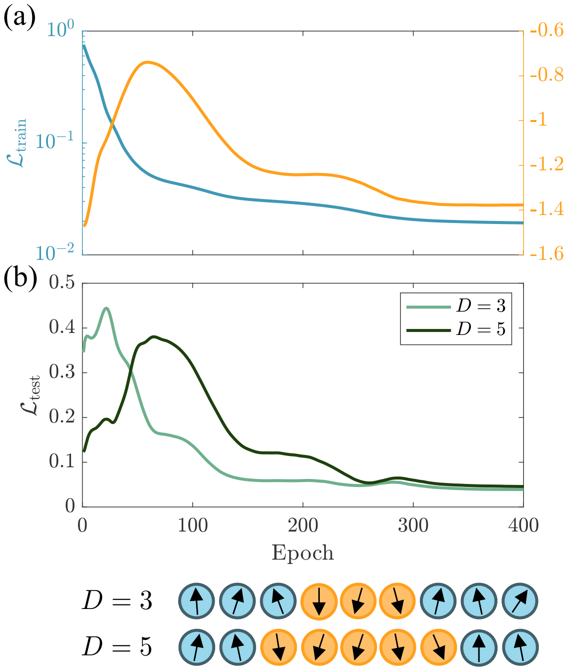

FIG. 4. (a) Training loss and tripartite information as functions of the training epoch. The initialization is the same as Initialization 1 in Fig. 2. (b) Loss function on the artificial test dataset with "ferromagnetic domain" of size D.

Before concluding this section, we point out another interesting observation that the linear increasing slope of the tripartite information is nearly a constant that is independent of the initialization, shown in Fig. 3(b). Of course, this slope depends on the learning rate of the gradient descent algorithm. As shown in the inset, the I_3-independent slope scales linearly with the learning rate.

Global Relaxation Stage. We now turn to the second stage where the tripartite information decreases and the training loss decreases with a much slower rate. We claim that during this stage, the quantum neural network learns global features of the wavefunction. To provide evidence for this claim, we test the quantum neural network in an artificial test dataset $\{(|\psi_{\mathsf{D}} \rangle, M_{\mathsf{z}} \ ), \alpha = 1, \ldots, N_{\mathsf{D}} \}$, constructed according to the following process. First, we sample ground states $|G \ \rangle$ from the random Hamiltonian of Eq. (2). Then we apply the following unitary transformation to flip a region of spins:

$$|\psi_{\mathsf{D}} \rangle = \prod_{\frac{n-D+1}{2} \leq \mathsf{i} \leq \frac{n+D}{2}} \sigma^{\mathsf{x}}_{\mathsf{i}} |G \ \rangle. \tag{5}$$

For "paramagnetic" wavefunctions $|G \ \rangle$, this transformation leaves these wavefunctions still "paramagnetic". However, for "ferromagnetic" wavefunctions $|G \ \rangle$, the transformation creates a ferromagnetic domain wall of size D, as sketched in Fig. 4 . In order to accurately compute the magnetization of such wavefunctions, the quantum neural network must be able to learn structures larger than the domain wall size D. In [35], we present an argument on why in this task, long string operators should exist in $\hat{U}^{\dagger}\sigma^{x}_{(n+1)=2}\hat{U}$ when it is expanded under the basis of product of local Pauli matrices.

In Fig. 4(b), we show losses on test datasets with $D = 3$ and 5, as functions of the training epoch. In the later stage of the training, although the training loss is decreasing slowly, the tripartite information can decrease rather drastically, accompanied with the rapid decreases of losses on these test datasets. Moreover, the larger the average domain wall size is, the later the test loss begins to decease. This means that the tripartite information deceasing is associated with the performance improvement on data with large domain structures. It naturally explains why the unitary has to become more scrambled during this stage. Since such data are rare in the training dataset, it also explains why the improvement of performance with respect to training loss is slow.

Discussion and Outlook. In summary, we apply an information-theoretic measure of the quantum information scrambling, namely the tripartite information, to diagnose the learning process of quantum neural networks. We find strong correlation between this metric and the loss function, and identify a two-stage training dynamics of quantum neural networks. We show that the neural network establishes local correlations in the early stage and builds up global structures in the later stage. We believe this two-stage scenario is applicable to a wide range of quantum machine learning problems.

We would also like to shed some physical insights onto this two-stage dynamical process. First, it is reminiscent of the annealing process. For instance, when cooling a spin system towards a ferromagnetic ground state, a fast process is to reach local equilibrium by forming ferromagnetic domains, and a slow process is to remove the domain walls and to reach global equilibrium. Second, the process can also be understood in terms of operator growth in the context of many-body quantum chaos. In fact, the formation of string operators in magnetization learning task is already discussed above. The time scales when the two stages end are direct analogs of the dissipation time and the scrambling time there, and the latter is believed to be longer than the former in a generic many-body system.

Finally we believe that the connection between the information scrambling and the quantum neural network is profound. The connection can find broader applications in quantum machine learning much beyond the neural network structure discussed here, including revealing the underlying mechanism of quantum machine learning and guiding quantum machine learning architecture design.

Acknowledgment. We thank Yingfei Gu for discussions. HS thanks IASTU for hosting his visit to Beijing, where key parts of this work were done. PZ acknowledges support from the Walter Burke Institute for Theoretical Physics at Caltech. This work is supported by Beijing Outstanding Young Scientist Program (HZ), MOST under Grant No. 2016YFA0301600 (HZ) and NSFC Grant No. 11734010 (HZ).

[*] hzhai@tsinghua.edu.cn

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning* (MIT Press, 2016).

[2] Marcello Benedetti, John Realpe-Gómez, Rupak Biswas, and Alejandro Perdomo-Ortiz, "Quantum-Assisted Learning of Hardware-Embedded Probabilistic Graphical Models," Phys. Rev. X **7**, 041052 (2017).

[3] E. Torrontegui and J. J. García-Ripoll, "Unitary quantum perceptron as efficient universal approximator," EPL (Europhysics Letters) **125**, 30004 (2019).

[4] Marcello Benedetti, Delfina Garcia-Pintos, Oscar Perdomo, Vicente Leyton-Ortega, Yunseong Nam, and Alejandro Perdomo-Ortiz, "A generative modeling approach for benchmarking and training shallow quantum circuits," npj Quantum Inf. **5**, 45 (2019).

[5] Edward Farhi and Hartmut Neven, "Classification with Quantum Neural Networks on Near Term Processors," arXiv:1802.06002.

[6] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven, "Barren plateaus in quantum neural network training landscapes," Nat. Commun. **9**, 4812 (2018).

[7] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, "Quantum circuit learning," Phys. Rev. A **98**, 032309 (2018).

[8] William Huggins, Piyush Patil, Bradley Mitchell, K Birgitta Whaley, and E Miles Stoudenmire, "Towards quantum machine learning with tensor networks," Quantum Sci. Technol. **4**, 024001 (2019).

[9] Maria Schuld, Alex Bocharov, Krysta Svore, and Nathan Wiebe, "Circuit-centric quantum classifiers," arXiv:1804.00633.

[10] Edward Grant, Marcello Benedetti, Shuxiang Cao, Andrew Hallam, Joshua Lockhart, Vid Stojevic, Andrew G Green, and Simone Severini, "Hierarchical quantum classifiers," npj Quantum Inf. **4**, 65 (2018).

[11] Jin-Guo Liu and Lei Wang, "Differentiable learning of quantum circuit Born machines," Phys. Rev. A **98**, 062324 (2018).

[12] Guillaume Verdon, Jason Pye, and Michael Broughton, "A Universal Training Algorithm for Quantum Deep Learning," arXiv:1806.09729.

[13] Jinfeng Zeng, Yufeng Wu, Jin-Guo Liu, Lei Wang, and Jiangping Hu, "Learning and inference on generative adversarial quantum circuits," Phys. Rev. A **99**, 052306 (2019).

[14] Yuxuan Du, Min-Hsiu Hsieh, Tongliang Liu, and Dacheng Tao, "The Expressive Power of Parameterized Quantum Circuits," arXiv:1810.11922.

[15] Kerstin Beer, Dmytro Bondarenko, Terry Farrelly, Tobias J Osborne, Robert Salzmann, and Ramona Wolf, "Efficient Learning for Deep Quantum Neural Networks," arXiv:1902.10445.

[16] Matthew J. S. Beach, Roger G. Melko, Tarun Grover, and Timothy H. Hsieh, "Making Trotters Sprint: A Variational Imaginary Time Ansatz for Quantum Many-body Systems," arXiv:1904.00019.

[17] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd, "Quantum machine learning," Nature **549**, 195–202 (2017).

[18] John Preskill, "Quantum Computing in the NISQ era and beyond," Quantum **2**, 79 (2018).

[19] Michael A. Nielsen and Isaac .L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, 2010).

[20] Ehud Altman, "Many-body localization and quantum thermalization," Nat. Phys. **14**, 979–983 (2018).

[21] Xiao-Liang Qi, "Does gravity come from quantum information?" Nat. Phys. **14**, 984–987 (2018).

[22] Brian Swingle, "Unscrambling the physics of out-of-time-order correlators," Nat. Phys. **14**, 988–990 (2018).

[23] A I Larkin and Yu N Ovchinnikov, "Quasiclassical Method in the Theory of Superconductivity," Sov. Phys. JETP **28**, 1200–1205 (1969).

[24] Alexei Kitaev, "Hidden correlations in the hawking radiation and thermal noise," A talk given at Fundamental Physics Prize Symposium, 2014.

[25] Stephen H. Shenker and Douglas Stanford, "Black holes and the butterfly effect," J. High Energy Phys. **2014**, 67 (2014).

[26] Juan Maldacena, Stephen H. Shenker, and Douglas Stanford, "A bound on chaos," J. High Energy Phys. **2016**, 106 (2016).

[27] Ruihua Fan, Pengfei Zhang, Huitao Shen, and Hui Zhai, "Out-of-time-order correlation for many-body localization," Sci. Bull. **62**, 707 – 711 (2017).

[28] Ravid Shwartz-Ziv and Naftali Tishby, "Opening the Black Box of Deep Neural Networks via Information," arXiv:1703.00810.

[29] Andrew Michael Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan Daniel Tracey, and David Daniel Cox, "On the information bottleneck theory of deep learning," in *International Conference on Learning Representations* (2018).

[30] Ziv Goldfeld, Ewout Van Den Berg, Kristjan Greenewald, Igor Melnyk, Nam Nguyen, Brian Kingsbury, and Yury Polyanskiy, "Estimating information flow in deep neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 97, edited by Kamalika Chaudhuri and Ruslan Salakhutdinov (PMLR, Long Beach, California, USA, 2019) pp. 2299–2308.

[31] Alexei Kitaev and John Preskill, "Topological Entanglement Entropy," Phys. Rev. Lett. **96**, 110404 (2006).

[32] Pavan Hosur, Xiao-Liang Qi, Daniel A. Roberts, and Beni Yoshida, "Chaos in quantum channels," J. High Energy Phys. **2016**, 4 (2016).

[33] P Dita, "Factorization of unitary matrices," J. Phys. A **36**, 2781–2789 (2003).

[34] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar, "On the Convergence of Adam and Beyond," in *International Conference on Learning Representations* (2018).

[35] See Supplemental Material, which includes Ref. [37], for further results on magnetization learning, results of winding number learning, and details of gradient calculation and measurement.

[36] For network initializations, we require initial unitaries to be scrambled enough such that initial $I_3(A,C,D) \lesssim -1$ (-1 is about is half of the negative-most value). For training algorithms, we require these algorithms to be gradient-based. For network depths, we require the networks to be not too shallow.

[37] Pengfei Zhang, Huitao Shen, and Hui Zhai, "Machine Learning Topological Invariants with Neural Networks," Phys. Rev. Lett. **120**, 066401 (2018).

Supplemental Material for “Information Scrambling in Quantum Neural Networks”

Huitao Shen,[1] Pengfei Zhang,[2] Yi-Zhuang You,[3] and Hui Zhai[2,*]

[1] *Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*
[2] *Institute for Advanced Study, Tsinghua University, Beijing, 100084, China*
[3] *Department of Physics, University of California, San Diego, CA 92093, USA*

In this supplemental material, we present more results of magnetization learning, results of winding number learning, and details of gradient calculation and measurement.

I. MAGNETIZATION LEARNING

In this section, we provide more details of magnetization learning and present an argument on why in magnetization learning, long string operators should exist in $\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U}$ when it is expanded under the basis of product of local Pauli matrices.

A. Learning Task Details

Figure 1 shows the distribution of magnetization M_z^α in the training and validation datasets. The magnetization distributions within the training and validation set are similar. There are roughly equal number of wavefuntions that are “ferromagnetic” ($|M_z| \geq 0.5$) or “paramagnetic” ($|M_z| < 0.5$).

For the AMSGrad algorithm [1], momentum parameters are always $\beta_1 = 0.9$ and $\beta_2 = 0.999$ throughout this work.

B. Explicit Construction of Unitary that Learns Magnetization

Generally, it is impossible to find an unitary $\hat{U}$ such that

$$\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U} = \hat{M}_z, \tag{1}$$

because the L.H.S. and the R.H.S. of the above equality have different eigenvalues. As a result, we can only expect the above equality to hold at the level of expectation

$$\langle\psi|\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U}|\psi\rangle = \langle\psi|\hat{M}_z|\psi\rangle\,, \tag{2}$$

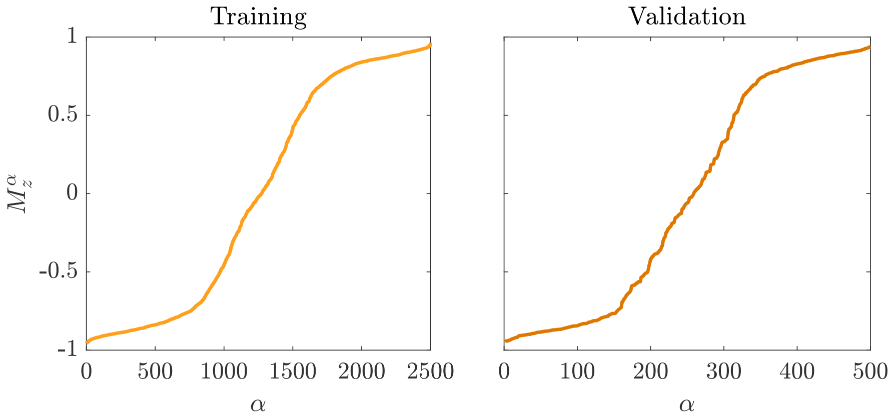

FIG. 1. Distribution of magnetization M_z^α in the training and validation sets.

[*] hzhai@tsinghua.edu.cn

within a subset of states $\{|\psi\rangle\}$ that are of interest [1]. In this section, we present an explicit construction of $\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U}$ to the magnetization learning problem when the subset of states are eigenstates of $\hat{M}_z \equiv \sum_{i=1}^n \sigma^z_i/n$. The purpose of this construction is to use an explicit example to demonstrate why it is usually necessary to have string operators in $\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U}$ for a quantum neural network that learns magnetization.

Denote the eigenstates of $\hat{M}_z$ as $|m,i\rangle$ such that $\hat{M}_z\,|m,i\rangle = m\,|m,i\rangle$. Here $m \in [-1,1]$ is the eigenvalue, which is also the average magnetization. $i = 1,\ldots,d_m$ represents the state in the degenerate eigenspace and d_m is the degeneracy. The states are orthonormal $\langle m,i|m',i'\rangle = \delta_{mm'}\delta_{ii'}$ and complete $\sum_m d_m = 2^n$. In general $d_m > 1$ unless $m = \pm 1$, where all spins are polarized to the same direction. For degenerate subspaces, note that the choice of $|m,i\rangle$ for fixed m but different i is not unique.

In the following, we construct matrix elements of $\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U}$ under $|m,i\rangle$ basis such that

$$\langle m,i|\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U}|m,i\rangle = m, \tag{3}$$

for all m and i. Consider the two-dimensional subspace spanned by $|m,i\rangle$ and $|-m,i\rangle$ for all m and i. Within this subspace, we set

$$\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U} = \sin\theta\sigma^x + \cos\theta\sigma^z, \tag{4}$$

where $\theta = \arccos m$. It is straightforward to verify the constraint Eq. (3) is satisfied and the eigenvalues are ± 1. Under this construction, half of $\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U}$'s eigenvalues are $+1$ and half are -1. It is then not hard to see that there must exist some $\hat{U}$ such that $\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U}$ has the matrix elements under $|m,i\rangle$ basis as constructed.

Although the above matrix is constructed explicitly on a particular choice of basis, it is straightforward to verify that the following basis-independent constraint holds

$$\langle m|\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U}|m\rangle = m, \tag{5}$$

where $|m\rangle \equiv \sum_{i=1}^{d_m} c_i\,|m,i\rangle$ is any linear combination of eigenstates within the same degenerate eigenspace. $\sum_{i=1}^{d_m}|c_i|^2 = 1$.

Because the choice of basis within a degenerate subspace is not unique, our constructions above are not unique either. Nevertheless, generally $|m,i\rangle$ and $|-m,i\rangle$ are related to each other by a string of local Pauli matrices whose length is of order of system size n. A particular choice is that $|-m,i\rangle = \prod_{j=1}^n \sigma^x_i\,|m,i\rangle$ such that the two states are related by a global spin-flip operator, which is a string operator of length n. Because of Eq. (4), such string operator must exist in $\hat{U}^\dagger\sigma^x_{(n+1)/2}\hat{U}$.

II. WINDING NUMBER LEARNING

In this section we present the results of winding number learning task, where empirical correlation between the neural network performance and the tripartite information are also found.

Dataset. The input data consist of N product states of n qubits, where each qubit represents a vector on the xz plane of the Bloch sphere. The target is the winding number of these vectors by treating the n qubits as vectors on an one-dimensional Brillouin zone [2]. Formally, the dataset consists of N input-target pairs $\{(|H^\alpha\rangle, w^\alpha), \alpha = 1,\ldots,N\}$, where the input wavefunction $|H^\alpha\rangle = \prod_{i=1}^n |\psi^\alpha(k_i)\rangle$, $k_i = 2\pi(i-1)/(n-1)$, and $\psi^\alpha(k)$ is the ground state of the following random two-band Hamiltonian in one-dimensional Brillouin zone $k \in [0,2\pi)$ with chiral symmetry $\sigma_y H(k)\sigma_y = -H(k)$:

$$H(k) = h_x(k)\sigma^x + h_z(k)\sigma^z. \tag{6}$$

Here the coefficient $h_\mu(k)$, $\mu = x,z$ is represented in terms of Fourier components up to p-th harmonic:

$$h_\mu(k) = \sum_{n=0}^{p}\cos(nk)c^\mu_n + \sum_{n=1}^{p}\sin(nk)s^\mu_n, \tag{7}$$

where c^μ_n and s^μ_n are random numbers.

The learning target is the discrete version of winding number:

$$w^\alpha = \frac{1}{2\pi}\sum_{i=1}^{n}\operatorname{Im}\ln\left[e^{i(\phi^\alpha(k_i)-\phi^\alpha(k_{i+1}))}\right], \tag{8}$$

[1] Note that the subset is in general not a subspace as linear combinations in general break the equality Eq. (2).

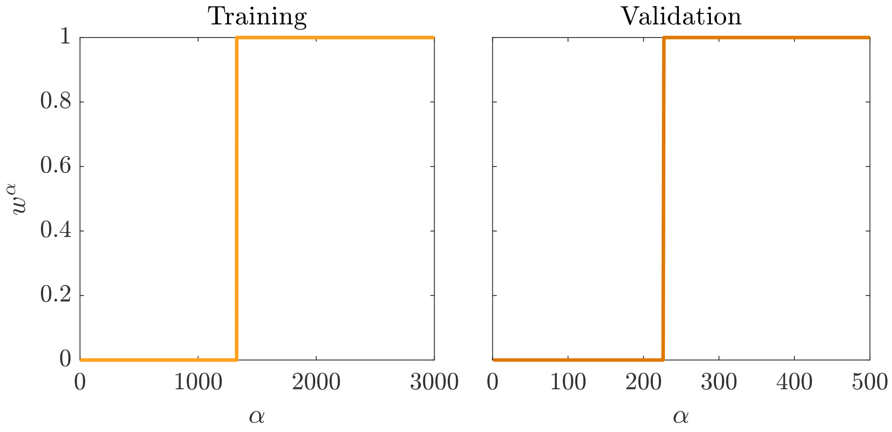

FIG. 2. Distribution of winding number w^α in the training and validation sets.

where $\phi^\alpha(k)$ is defined as the argument of the following complex number:

$$e^{i\phi^\alpha(k)} = \frac{h_z^\alpha(k) + ih_x^\alpha(k)}{\sqrt{h_z^\alpha(k)^2 + h_x^\alpha(k)^2}}. \tag{9}$$

The branch cut for the logarithm in Eq. (8) is along the negative direction of the x axis such that $\phi(k) - \phi(k') \in [-\pi, \pi)$.

Task. In the following, we set the harmonic cutoff $p = 1$. c_n^μ and s_n^μ are sampled from a uniform distribution between $[-1/3, 1/3]$ for $n = 0$ and $[-1, 1]$ for $n > 0$. We then post-select data with winding number $w = 0, 1$ and discard those with $w = -1$. In this way, the task becomes binary classification. The parameters are chosen such that there are roughly equal number of data with $w = 0$ and 1, as shown in Fig. 2.

The quantum neural network takes the input wavefunction $|H^\alpha\rangle$ and applies the unitary transformation $\hat{U}$ on it. The probability that the $w^\alpha = 1$ is readout by measuring σ^x of the central qubit:

$$p^\alpha = \frac{1 + \langle H^\alpha| \hat{U}^\dagger \sigma^x_{(n+1)/2} \hat{U} |H^\alpha\rangle}{2}. \tag{10}$$

Therefore, the loss function to be minimized is the negative binary cross-entropy:

$$\mathcal{L} = \frac{1}{N} \sum_{\alpha=1}^{N} \left[-w^\alpha \ln p^\alpha - (1 - w^\alpha) \ln(1 - p^\alpha) \right]. \tag{11}$$

A more sensible metric is the prediction accuracy. Let the prediction of the winding number be $o^\alpha \equiv (1 + \mathrm{sgn}(p_a))/2$. The prediction accuracy is then

$$\mathcal{A} \equiv \frac{1}{N} \sum_{\alpha=1}^{N} |o^\alpha - w^\alpha|. \tag{12}$$

Results. In Fig. 3, we present the training loss and accuracy for the winding number learning task, along with the tripartite information. We confirm the validation loss and accuracy is similar to that in the training set. The network depth l is larger than that in the magnetization learning as we suspect the winding learning task is more difficult. However, using a shallower network will not affect the performance significantly.

First, the quantum neural network manages to learn distinguish wavefunctions with winding number $w = 0$ and 1, as the final accuracy is more than 90%. Second, the trend of the loss function and the tripartite information is similar to that in magnetization learning: At the early stage of the training, the loss decreases rapidly and the tripartite information increases. In the later stage, the tripartite information decreases again. However, we note the tripartite information is more volatile in the later stage than that in the magnetization learning. We leave the in-depth understanding of this behavior for future research.

Compared with the magnetization task, the input wavefunction here is a product state and is essentially classical, and the target is now a binary label instead of a real number. Despite the very different nature of this task, the empirical correlation between the neural network performance and the tripartite information still holds. This suggests the generality of the two-stage training dynamics of quantum neural networks.

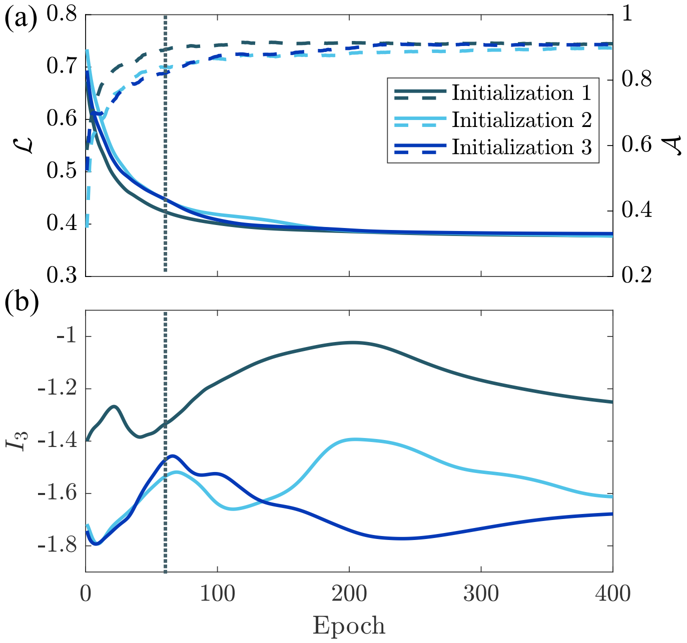

FIG. 3. (a) Training loss (solid) and accuracy (dashed) as functions of the training epoch. Different colors represent different random initializations. The network has $n = 9$ qubits and depth $l = 8$. The training and validation dataset contains $N = 3000$ and 500 wavefunction-winding number pairs respectively, sampled from random wavefunctions defined in the main text. The learning rate is $\lambda = 10^{-2}$. (b) Tripartite information $I_3(A, C, D)$ as a function of the training epoch for different initializations. Here the input subsystem size $|C| = 5$. The dotted vertical line indicates the boundary between two training stages.

III. GRADIENTS IN QUANTUM NEURAL NETWORKS

In this section, we report the method of computing gradients of quantum neural networks in this work.

A. In Classical Simulations

A schematic of the quantum neural network with $n = 5$ qubits and depth $l = 4$ is shown in Fig. 4. The i-th two-qubit gate in the d-th layer is denoted as $\hat{U}_i^d$. Assuming n is odd, here $i = 1, 2 \ldots (n-1)/2$. It follows the giant unitary $\hat{U}$ is the composition of $\hat{U}_i^d$:

$$\hat{U} = \left(\prod_{i=1}^{(n-1)/2} \hat{U}_i^l\right) \ldots \left(\prod_{i=1}^{(n-1)/2} \hat{U}_i^2\right) \left(\prod_{i=1}^{(n-1)/2} \hat{U}_i^1\right) \equiv \prod_{d=1}^{l} \left(\prod_{i=1}^{(n-1)/2} \hat{U}_i^d\right). \tag{13}$$

The order of unitaries within a layer does not matter because these unitaries are applied on non-overlapping qubits.

In general, each two-qubit gate $\hat{U}_i^d$ is a 4×4 matrix in the SU(4) group and can be parametrized by 15 parameters. However, as explained in the main text, in this work we restrict $\hat{U}_i^d$ to $\mathrm{SO}(4)$ with 6 Euler angles: Generally, a matrix in $\mathrm{SO}(4)$ can be parametrized by a vector $\boldsymbol{\theta}$ with 6 components [3]:

$$\hat{U}_{\mathrm{SO}(4)} = O_{34}(\theta_1)O_{23}(\theta_2)O_{12}(\theta_3)O_{34}(\theta_4)O_{23}(\theta_5)O_{34}(\theta_6). \tag{14}$$

Here $O_{ij}(\theta) \equiv \exp(\theta J_{ij})$ is a rotation in the ij plane: J_{ij} an antisymmetric matrix with ij (ji) element equal to 1 (-1) and all other elements zero. As a result there are $l(n-1)/2$ independent vectors $\boldsymbol{\theta}_i^d$ and thus $6l(n-1)/2$ independent parameters in total to fully describe the quantum neural network.

To be concrete, in the following, we use magnetization learning as the example. The winding number learning is similar. The loss function in magnetization learning is

$$\mathcal{L} = \frac{1}{N}\sum_{\alpha=1}^{N} \left| \langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \hat{U} | G^\alpha \rangle - M_z^\alpha \right|. \tag{15}$$

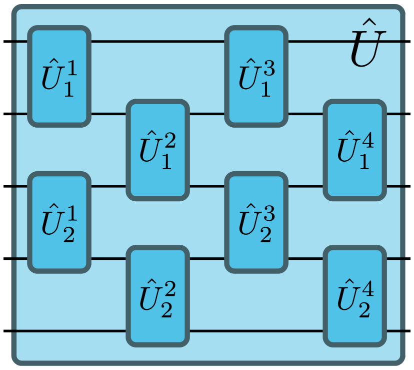

FIG. 4. Schematic of a quantum circuit with brick-wall geometry. Here the network has $n = 5$ qubits and depth $l = 4$. All these two-qubit gates form a giant unitary transformation $\hat{U}$. The i-th two-qubit gate in the d-th layer is denoted as $\hat{U}_i^d$

The gradient of $\mathcal{L}$ with respect to $\theta^d_{j,a}$, $a = 1, \dots, 6$ is

$$\frac{\partial \mathcal{L}}{\partial \theta^d_{j,a}} = \frac{1}{N} \sum_{\alpha=1}^{N} \operatorname{sgn}\left(\langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \hat{U} | G^\alpha \rangle - M_z^\alpha \right) \frac{\partial \langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \hat{U} | G^\alpha \rangle}{\partial \theta^d_{j,a}}. \tag{16}$$

The gradient of the network output can be further simplified as

$$\begin{aligned}
&\frac{\partial}{\partial \theta^d_{j,a}} \langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \hat{U} | G^\alpha \rangle \\
&= \langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \frac{\partial \hat{U}}{\partial \theta^d_{j,a}} | G^\alpha \rangle + \text{h.c.} \\
&= \langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \frac{\partial}{\partial \theta^d_{j,a}} \left[\prod_{d=1}^{l} \left(\prod_{i=1}^{(n-1)/2} \hat{U}_i^d \right) \right] | G^\alpha \rangle + \text{h.c.} \\
&= \langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \left(\prod_{i=1}^{(n-1)/2} \hat{U}_i^l \right) \dots \left(\hat{U}_1^d \hat{U}_2^d \dots \frac{\partial \hat{U}_j^d}{\partial \theta^d_{j,a}} \dots \hat{U}^d_{\frac{n-1}{2}} \right) \dots \left(\prod_{i=1}^{(n-1)/2} \hat{U}_i^1 \right) | G^\alpha \rangle + \text{h.c.},
\end{aligned} \tag{17}$$

where, $\partial \hat{U}_j^d / \partial \theta^d_{j,a}$ can be further simplified using Eq. (14). For example,

$$\frac{\partial \hat{U}_j^d}{\partial \theta^d_{j,4}} = O_{34}(\theta^d_{j,1}) O_{23}(\theta^d_{j,2}) O_{12}(\theta^d_{j,3}) J_{34} O_{34}(\theta^d_{j,4}) O_{23}(\theta^d_{j,5}) O_{34}(\theta^d_{j,6}). \tag{18}$$

Gradients with respect to other components a can be computed in the similar way by adding an additional corresponding J matrices.

In this work, we directly compute the gradient according to Eqs. (16), (17) and (18) in the classical simulation.

B. In Real Quantum Neural Networks

In a real quantum neural network, this gradient could instead be determined through the measurement of the following Hermitian operator:

$$\hat{g}^d_{j,a} = \sigma^x_{(n+1)/2} \frac{\partial \hat{U}}{\partial \theta^d_{j,a}} \hat{U}^\dagger + \text{h.c.}. \tag{19}$$

It is straightforward to see that

$$\langle G^\alpha | \hat{U}^\dagger \hat{g}^d_{j,a} U | G^\alpha \rangle = \frac{\partial}{\partial \theta^d_{j,a}} \langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \hat{U} | G^\alpha \rangle .$$

However, this operator is generally non-local and is hard to measure.

Alternatively, one could perform the following three measurements [4, 5]:

1. Measure the output of the quantum neural network normally with the original parameter $\boldsymbol{\theta}_i^d$. The result is denoted as $o_1 \equiv \langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \hat{U} | G^\alpha \rangle$;

2. Measure the output of the quantum neural network with $\theta^d_{j,a}$ replaced by $\theta^d_{j,a} + \pi/4$. The result is denoted as o_2;

3. Measure the output of the quantum neural network with $\theta^d_{j,a}$ replaced by $\theta^d_{j,a} + \pi/2$. The result is denoted as o_3.

It follows the desired gradient is

$$\frac{\partial}{\partial \theta^d_{j,a}} \langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \hat{U} | G^\alpha \rangle = 2o_2 - o_1 - o_3. \tag{20}$$

The reason is that if we focus on some specific $\theta^d_{j,a}$, we have

$$o_1 = \left\langle \dots O^\dagger_{p,p+1}(\theta^d_{j,a}) \dots O_{p,p+1}(\theta^d_{j,a}) \dots \right\rangle, \tag{21}$$

$$\begin{aligned} o_2 &= \left\langle \dots O^\dagger_{p,p+1}(\theta^d_{j,a} + \pi/4) \dots O_{p,p+1}(\theta^d_{j,a} + \pi/4) \dots \right\rangle \\ &= \left\langle \dots \left[(1 + J_{p,p+1}) O_{p,p+1}(\theta^d_{j,a})\right]^\dagger \dots (1 + J_{p,p+1}) O_{p,p+1}(\theta^d_{j,a}) \dots \right\rangle /2, \end{aligned} \tag{22}$$

$$\begin{aligned} o_3 &= \left\langle \dots O^\dagger_{p,p+1}(\theta^d_{j,a} + \pi/2) \dots O_{p,p+1}(\theta^d_{j,a} + \pi/2) \dots \right\rangle \\ &= \left\langle \dots \left[J_{p,p+1} O_{p,p+1}(\theta^d_{j,a})\right]^\dagger \dots J_{p,p+1} O_{p,p+1}(\theta^d_{j,a}) \dots \right\rangle. \end{aligned} \tag{23}$$

Here $p(p+1)$ is the rotation plane associated with a. As a result:

$$2o_2 - o_1 - o_3 = \left\langle \dots O^\dagger_{p,p+1}(\theta^d_{j,a}) \dots J_{p,p+1} O_{p,p+1}(\theta^d_{j,a}) \dots \right\rangle + \text{h.c.} = \frac{\partial}{\partial \theta^d_{j,a}} \langle G^\alpha | \hat{U}^\dagger \sigma^x_{(n+1)/2} \hat{U} | G^\alpha \rangle. \tag{24}$$

The above method can be easily generalized to $\mathrm{SU}(4)$ as well.

[1] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar, "On the Convergence of Adam and Beyond," in *International Conference on Learning Representations* (2018).

[2] Pengfei Zhang, Huitao Shen, and Hui Zhai, "Machine Learning Topological Invariants with Neural Networks," Phys. Rev. Lett. **120**, 066401 (2018).

[3] P Dita, "Factorization of unitary matrices," J. Phys. A **36**, 2781–2789 (2003).

[4] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, "Quantum circuit learning," Phys. Rev. A **98**, 032309 (2018).

[5] Maria Schuld, Alex Bocharov, Krysta Svore, and Nathan Wiebe, "Circuit-centric quantum classifiers," arXiv:1804.00633.