

Virtual quantum neural networks

Benchi Zhao,¹ Xuanqiang Zhao,¹ Yinan Li,^{2,3,4,5} Yingzhou Li,^{6,7,*} and Giulio Chiribella^{1,8,9,†}

¹*QICI Quantum Information and Computation Initiative, School of Computing and Data Science,
The University of Hong Kong, Pokfulam Road, Hong Kong*

²*School of Artificial Intelligence, Wuhan University, Wuhan, China*

³*Hubei Center for Applied Mathematics, Wuhan, China*

⁴*Hubei Key Laboratory of Computational Science, Wuhan, China*

⁵*Wuhan Institute of Quantum Technology, Wuhan, China*

⁶*School of Mathematical Sciences, Shanghai Key Laboratory for Contemporary Applied Mathematics, Fudan University*

⁷*Key Laboratory of Computational Physical Sciences, Ministry of Education*

⁸*Quantum Group, Department of Computer Science, University of Oxford,
Wolfson Building, Parks Road, Oxford, OX1 3QD, United Kingdom*

⁹*Perimeter Institute for Theoretical Physics, 31 Caroline Street North, Waterloo, Ontario, Canada*

Quantum neural networks are a prominent model of quantum machine learning. Their training consists in the minimization of a given loss function over a parametrized family of quantum circuits, mathematically described by unitary operators, or, more generally, completely positive linear maps. In this work, we extend the notion of quantum neural network, using random sampling and classical data processing to enlarge the optimization space in a way that includes linear combinations of completely positive maps. Our extended model, called *virtual quantum neural networks*, leverages its enlarged optimization space to achieve increased expressivity and improved noise robustness. These benefits are illustrated in three representative tasks: quantum error mitigation, binary classification, and estimation of ground-state energies. Overall, virtual quantum neural networks offer a flexible learning paradigm that expands the space of achievable computations and strengthens the applications of near-term quantum hardware.

I. INTRODUCTION

The present stage of quantum computing is known as the noisy intermediate-scale quantum (NISQ) era [1]. In this stage, noisy quantum computations of moderate depth can be implemented on a few hundred qubits, starting to probe a new regime beyond the reach of classical numerical simulations. Exploiting these capabilities, several applications of NISQ quantum processors have been identified, including optimization [2, 3], chemistry and material science [4–6], and cryptography [7, 8], a broad class of NISQ algorithms has been designed [9–15].

A notable approach to quantum computing in the NISQ era is provided by quantum neural networks (QNNs) [16–20] (see also [21] for a review in the broader context of quantum machine learning). In a QNN, an achievable family of quantum circuits is specified by a set of parameters, which are optimized according to a loss function evaluated on classical data obtained from measurements. This quantum–classical optimization loop is appealing for near-term applications because it offers adaptability to hardware limitations while retaining sufficient expressivity for realizing non-trivial quantum computations. It provides a prominent model of quantum machine learning (QML) [22, 23] and underpins all variational quantum algorithms (VQAs) [24], supporting QML primitives such as quantum classifiers [25, 26], data compression [27–29], and generative models [30, 31].

Technically, a QNN is described by a parametrized family of unitary operators, or, more generally, a parametrized fam-

ily of completely positive trace-preserving (CPTP) maps. The minimization of the loss function is then performed on the family of CPTP maps reachable by varying the QNN parameters. This framework, however, entails a fundamental limitation, since the optimization is by default constrained to a subset of CPTP maps achievable on the given quantum hardware. This limitation restricts the expressivity of the model, typically restricting it to computations represented by shallow circuits, and in turn affects its practical applications.

In this paper, we develop a more general model that lifts the above limitations by sampling over multiple QNNs and by classically postprocessing the measurement outcomes obtained in different runs. The combined use of random sampling and classical postprocessing allows us to access a broader set of maps, consisting of all possible linear combinations of the CPTP maps achievable by the sampled QNNs. This extension provides a larger optimization space, increasing the expressivity of the model and potentially unlocking new quantum advantages.

We call our extended QNN model a *virtual QNN* (VQNN) and illustrate its quantum-classical optimization loop in Fig. 1. Compared with conventional QNNs, VQNNs framework offers two key advantages: enhanced expressivity and improved robustness against quantum noise. We demonstrate these features in three representative quantum machine learning tasks, including both supervised and unsupervised learning. First, we consider the problem of quantum error mitigation, introducing a technique of variational quantum error mitigation (VQEM). Numerical results show that the mitigation strategies obtained via VQNNs are near-optimal when implemented with noiseless parameterized quantum circuits, and remain effective even in the presence of circuit-level noise. Second, we develop a binary classification algorithm, which

* yingzhouli@fudan.edu.cn

† giulio@hku.hk

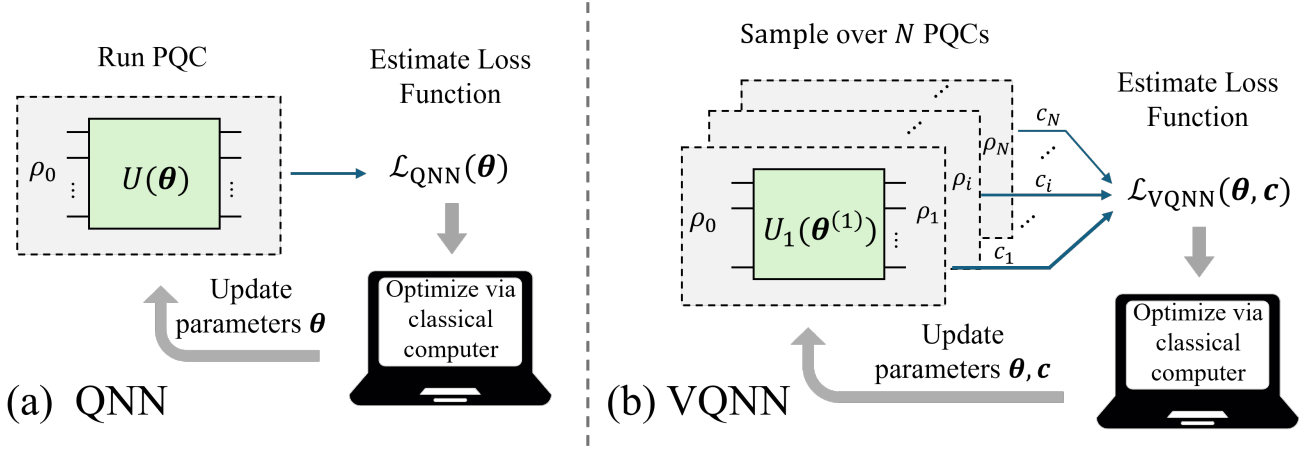


Fig 1. **(a) Training of a quantum neural network.** The network is designed to implement a parametrized quantum circuit (PQC), depending on a set of parameters θ . In the noiseless case, the PQC is represented by a unitary operator $U(\theta)$, which is applied to a quantum system, initially in the state ρ_0 . Then, a measurement is performed, and the statistics of the outcomes is used to estimate the value of a loss function $\mathcal{L}_{\text{QNN}}(\theta)$. Finally, the classical optimization method is used to update the parameters in the PQC to minimize the loss function. **(b) Training of a virtual quantum neural network.** N parametrized quantum circuits (PQCs) are randomly sampled and applied to the input state ρ_0 , producing output states $\rho_1, \dots, \rho_N$. Quantum measurements performed on the output states are then used to estimate the value of the loss function $\mathcal{L}_{\text{VQNN}}(\theta, \mathbf{c})$, which depends on the parameters of the sampled circuits, and on a set of coefficients $c_1, \dots, c_N$, used to aggregate the data from the N different branches. The value of the loss function is then fed into a classical optimization algorithm used to update the value of the parameters in the PQCs.

we call the virtual quantum classifier (VQC). For both training and testing data sets, VQC consistently achieves higher prediction accuracy than conventional QNN-based classifiers. Finally, we apply the VQNN framework to the problem of ground-state energy estimation, proposing a virtual variational quantum eigensolver (VVQE). With noisy circuits, VVQE yields significantly more accurate estimates of the ground-state energy compared with standard QNN approaches. Taken together, these results establish VQNNs as a flexible paradigm for quantum machine learning, which enhances expressivity and reduces sensitivity to noise, thereby advancing the prospects for future practical applications.

The rest of the paper is structured as follows: in Section II, we introduce the VQNN model and discuss its general features. The applications are then presented in Section III, where we apply the VQNN model to two supervised learning tasks, namely quantum noise mitigation (Subsection III A) and quantum classification (Subsection III B), and to an unsupervised learning task, namely the estimation of the ground state energy of a binuclear molecule (Subsection III C). Finally, Section IV provides a discussion of our findings and draws the conclusions.

II. VIRTUAL QUANTUM NEURAL NETWORKS

A. Standard QNNs

Before introducing our extended notion, here we briefly summarize the standard notion of QNN [24], illustrated in Fig. 1 (a). A standard QNN is physically implemented by a parametrized quantum circuit, acting on a given quantum

system, typically consisting of a finite number of qubits. In the noiseless case, the circuit is described by a parametrized unitary operator $U(\theta)$, acting on the Hilbert space $\mathcal{H}$ of the given system and depending on a set of parameters θ . In the noisy case, the circuit is described by a parametrized quantum channel, that is, a completely positive, trace-preserving linear map $\mathcal{C}(\theta)$ mapping linear operators on the Hilbert space $\mathcal{H}$ into linear operators on a (possibly different) Hilbert space $\mathcal{K}$, and again, depending on a set of parameters θ . In the following, we will denote by $L(\mathcal{H})$ ($L(\mathcal{K})$) the set of linear operators from the Hilbert space $\mathcal{H}$ ($\mathcal{K}$) to itself.

The training of the QNN, illustrated in Fig. 1 (a), is aimed at minimizing the value of a loss function. In most cases, the loss function is a function of the expectation value of a given observable. In the noiseless case, this type of loss function can be written as

$$\mathcal{L}_{\text{QNN}}(\theta) := f(\text{Tr}[O U(\theta) \rho_0 U^\dagger(\theta)]) , \quad (1)$$

where $\rho_0 \in L(\mathcal{H})$ is a density matrix ($\text{Tr}[\rho_0] = 1$, and $\langle \psi | \rho_0 | \psi \rangle \geq 0, \forall |\psi\rangle \in \mathcal{H}$), representing a state of the quantum system, $O \in L(\mathcal{K})$ is a Hermitian operator ($O^\dagger = O$), representing an observable, and $f : \mathbb{R} \rightarrow \mathbb{R}$ is a real-valued function depending the task at hand. A basic requirement for training is that the loss function is lower bounded, so that a finite minimum exists, in the same way as in the noiseless case. Other requirements, such as differentiability, may be further imposed depending on the minimization algorithm used to minimize the loss function.

In the noisy case, the loss function takes the more general form

$$\mathcal{L}_{\text{QNN}}(\theta) = f(\text{Tr}[O \mathcal{C}(\theta)(\rho_0)]) , \quad (2)$$

where $\rho_0 \in L(\mathcal{H})$ is a quantum state of the input system, $O_k \in L(\mathcal{K})$ is an observable on the output system, and $f : \mathbb{R} \rightarrow \mathbb{R}$ is a real-valued function associated to the task at hand.

Both in the noiseless and noisy case, the value of the loss function can be directly estimated by initializing the input system in the state ρ_0 , running the circuit, and performing a measurement of the observable O on the final state. The estimated value of the loss function is then fed into a classical optimizer, which updates the parameters θ , searching for the value that minimizes the loss, in the same way as in the noiseless case.

B. Noiseless VQNNs

We are now ready to introduce the framework of VQNNs. We will start from its basic noiseless version, illustrated in Fig. 1 (b). In the noiseless case, a VQNN is specified by N real coefficients $c_1, \dots, c_N$, and by N unitary operators $U_1(\theta^{(1)}), \dots, U_N(\theta^{(N)})$, where $\theta^{(i)}$ are the parameters of the i -th circuit, for $i \in \{1, \dots, N\}$. Each of these unitary operators represents a parametrized quantum circuit (PQC) acting on a given quantum system with Hilbert space $\mathcal{H}$.

The loss function is defined in terms of an observable O , mathematically described by a Hermitian operator $O \in L(\mathcal{H})$: explicitly, the loss function is

$$\mathcal{L}_{\text{VQNN}}(\theta, c) := f \left(\sum_{i=1}^N c_i \text{Tr} \left[O U_i(\theta^{(i)}) \rho_0 U_i^\dagger(\theta^{(i)}) \right] \right), \quad (3)$$

where we used the notations $c := (c_1, \dots, c_N)$ and $\theta := (\theta^{(1)}, \dots, \theta^{(N)})$, for the coefficients and the parameters, respectively. In the above equation, $\rho_0 \in L(\mathcal{H})$ is a density matrix, representing a state of the quantum system, while the function $f : \mathbb{R} \rightarrow \mathbb{R}$ depends on the task at hand, as in the case of standard QNNs.

The loss function Eq. (3) can be empirically estimated from measurement data. More precisely, measurement data can be used to obtain an empirical estimate of the loss function's argument

$$x(\theta, c) := \sum_{i=1}^N c_i \text{Tr} \left[O U_i(\theta^{(i)}) \rho_0 U_i^\dagger(\theta^{(i)}) \right]. \quad (4)$$

The estimation procedure is as follows. First, the quantum system is initialized in the state ρ_0 . Then, it undergoes the unitary gate $U_i(\theta^{(i)})$ with probability $p_i := |c_i|/\|c\|_1$, with

$$\|c\|_1 := \sum_{j=1}^N |c_j|. \quad (5)$$

Due to the action of the gate, the system ends up in the state $\rho_i(\theta) := U_i(\theta^{(i)}) \rho_0 U_i^\dagger(\theta^{(i)})$. Finally, the expectation value of the observable O is estimated, e.g. by performing a von Neumann measurement on the eigenstates of the operator O , which yields an outcome o_k with probability $p_k^{(i)} := \text{Tr}[P_k \rho_i(\theta)]$ where o_k is one of the eigenvalues of O and P_k is the projector on the corresponding eigenspace.

The above procedure is repeated for S times, every time sampling one of the unitary gates $(U_1(\theta^{(1)}), \dots, U_N(\theta^{(N)}))$ according to the probability distribution $(p_1, \dots, p_N)$. After the S rounds of sampling have been concluded, the measurement outcomes are used to compute the linear combination

$$\zeta_S = \frac{\|c\|_1}{S} \sum_{s=1}^S \frac{c_{i(s)}}{|c_{i(s)}|} o_{k(s)}, \quad (6)$$

where the index $s \in \{1, \dots, S\}$ labels the round, $k(s)$ is the label of the measurement outcome obtained in the s -th round, and $i(s)$ is the label of the unitary gate chosen in the s -th round.

For large S , the law of large numbers implies that the quantity ζ_S in Eq. (6) converges with high probability to $x(\theta, c)$, the argument of the loss function. Quantitatively, Hoeffding's inequality [32] guarantees that ζ_S has probability at least $1 - \delta$ to deviate at most ε from $x(\theta, c)$, provided that the number of sampling rounds is at least

$$S = 2 \frac{\|c\|_1^2 \|O\|_\infty^2}{\varepsilon^2} \ln \left(\frac{2}{\delta} \right), \quad (7)$$

$\|O\|_\infty := \max_k |\rho_k|$ is the operator norm of the observable O . Note that, for a fixed observable O and error parameters ε and δ , the sampling time S depends only on $\|c\|_1$. For this reason, $\|c\|_1$ is called the *sampling overhead* [33–35]. Finally, the value of ζ_S is plugged into the function f , thereby providing an estimate of the loss function $\mathcal{L}(\theta, c) = f(x(\theta, c))$.

After estimating the loss function $\mathcal{L}(\theta, c)$, classical optimization methods [18, 36–39] are used to update the parameters of the PQCs in order to minimize the loss function. This classical-quantum optimization loop is iterated until the loss function converges to a minimum. In this work, the search for the minimum has been carried out using *Adam* [40], a popular optimization algorithm based on stochastic gradient descent.

It is worth mentioning that the training of standard QNNs may be subject to the problem of barren plateaus [41], arising when the gradients of the loss function vanish exponentially with the size of the system. A similar issue can affect also VQNNs; however, the additional flexibility of the VQNN framework does not generally come at the price of aggravating the problem of barren plateaus, as discussed in Appendix A.

C. Noisy VQNNs

In general, NISQ circuits are noisy, and instead of implementing unitary operators, they implement quantum channels, mathematically described by CPTP maps. The formulation of the VQNN model in the noisy setting is a straightforward modification of the noiseless formulation: a noisy VQNN is specified by N real coefficients $c_1, \dots, c_N$, and by N parametrized CPTP maps $\mathcal{C}_1(\theta^{(1)}), \dots, \mathcal{C}_N(\theta^{(N)})$, each of which transforms an input system, with Hilbert space $\mathcal{H}$, into an output system, with Hilbert space $\mathcal{K}$.

The training consists in the minimization of the loss function

$$\mathcal{L}_{\text{VQNN}}(\boldsymbol{\theta}, \mathbf{c}) := f \left(\sum_{i=1}^N c_i \text{Tr} \left[O \mathcal{C}_i(\boldsymbol{\theta}^{(i)}) (\rho_0) \right] \right), \quad (8)$$

where $\rho_0 \in L(\mathcal{H})$ is a density matrix on the input system, $O \in L(\mathcal{K})$ is an observable on the output system, and $f : \mathbb{R} \rightarrow \mathbb{R}$ is a real-valued function depending on the problem at hand.

The loss function can be empirically estimated by preparing the input system in the initial state ρ_0 , applying the channel $\mathcal{C}_i(\boldsymbol{\theta}^{(i)})$ with probability $p_i = |c_i|/\|\mathbf{c}\|_1$, measuring the observable O , and postprocessing the outcomes, in the same way as it was described in the noiseless case.

D. Comparison to standard QNNs

A standard QNN can be seen as the special case of a VQNN with a single branch ($N = 1$), corresponding to a single parametric family of quantum channels $\mathcal{C}_1(\boldsymbol{\theta}^{(1)})$, with the corresponding coefficient set to the default value $c_1 = 1$. Since there is only one branch, here we will drop the branch index, writing $\boldsymbol{\theta}$ and $\mathcal{C}(\boldsymbol{\theta})$ in place of $\boldsymbol{\theta}^{(1)}$ and $\mathcal{C}_1(\boldsymbol{\theta}^{(1)})$, respectively. In this case, the loss function (2) is simply equal to the function f applied to the expectation value of the observable O on the output state $\mathcal{C}(\boldsymbol{\theta})(\rho_0)$.

For a general VQNN, instead, the loss function Eq. (8) can be rewritten as

$$\mathcal{L}_{\text{VQNN}}(\boldsymbol{\theta}, \mathbf{c}) := f(\text{Tr}[O \mathcal{M}(\boldsymbol{\theta}, \mathbf{c})(\rho_0)]) , \quad (9)$$

with

$$\mathcal{M}(\boldsymbol{\theta}, \mathbf{c}) := \sum_{i=1}^N c_i \mathcal{C}_i(\boldsymbol{\theta}^{(i)}) . \quad (10)$$

The key difference between a standard QNN and a VQNN can be seen by comparing the loss functions in Eq. (2) and Eq. (9), respectively. For a standard QNN, the loss function is minimized over a parametric family of quantum channels, mathematically described by CPTP maps. For a virtual QNN, instead, the loss function is minimized over a larger parametric family, containing all linear combinations of the CPTP maps associated to the N branches. In general, such linear combinations are Hermitian-preserving (HP) maps, that is, linear maps that transform Hermitian operators into Hermitian operators. HP maps are also known as *virtual quantum operations*, and have applications in several quantum tasks, including entanglement detection [42–44], reduced dynamics of open quantum systems [45–47], and quantum error mitigation [33, 48–52].

In summary, the loss function for a VQNN is equal to a function f over the expectation value of the observable O on the output of the virtual quantum operation $\mathcal{M}(\boldsymbol{\theta}, \mathbf{c})$, applied to the initial state ρ_0 . By extending the optimization space from a parametric family of CPTP maps to a larger

parametric family of HP maps, VQNNs offer increased expressivity and performance by leveraging a larger optimization space. The price to pay for these enhancements is an increased sampling time, according to the sampling overhead γ , and an increased classical computational effort, required for minimizing the cost function over a larger set of parameters. In other words, the VQNNs framework can be seen as a way to overcome some of the limitations of the available quantum hardware by investing additional resources into sampling and classical computation. It is also worth noting that the N branches of the VQNN can be executed in parallel, meaning that the VQNN framework is naturally suitable for the distributed computing paradigm.

E. Further extensions

A further extension of our model is obtained by sampling over quantum circuits that include measurement devices. Mathematically, a quantum circuit with M possible outcomes is represented by a quantum instrument, that is, an indexed set of completely positive maps $(\mathcal{C}_j)_{j=1}^M$, with the map $\mathcal{C}_j$ corresponding to the j -th outcome. The normalization condition for a quantum instrument is that the sum $\sum_{j=1}^M \mathcal{C}_j$ is a CPTP map.

In the most general case, a VQNN can be implemented by sampling over N quantum instruments, each of which is parametrized by a set of parameters. The interest of this generalization is that the outcomes of the instruments can be used in the postprocessing stage, thereby computing linear combinations of the expectation values conditional on different measurement outcomes. This outcome-dependent postprocessing allows one to compute linear combinations of the form

$$x(\boldsymbol{\theta}, \mathbf{c}) := \sum_{i=1}^N \sum_{j=1}^M c_{ij} \text{Tr} \left[O \mathcal{C}_{ij}(\boldsymbol{\theta}^{(i)}) (\rho_0) \right] , \quad (11)$$

where $(\mathcal{C}_{ij}(\boldsymbol{\theta}_i))_{j=1}^M$ denotes the i -th quantum instrument, $(c_{ij})_{j=1}^M$ are the corresponding coefficients, $\boldsymbol{\theta} := (\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_N)$ are the parameters of the N circuits, and $\mathbf{c} := (c_{ij})_{1 \leq i \leq N, 1 \leq j \leq M}$ is the matrix of all coefficients. The training of the VQNN is then designed to minimize the value of a loss function of the form $\mathcal{L}(\boldsymbol{\theta}, \mathbf{c}) := f(x(\boldsymbol{\theta}, \mathbf{c}))$ where $f : \mathbb{R} \rightarrow \mathbb{R}$ is a function depending on the task at hand.

Computing the linear combination (11) by classical postprocessing is equivalent to simulating the linear map

$$\mathcal{M}(\boldsymbol{\theta}, \mathbf{c}) := \sum_{i=1}^N \sum_{j=1}^M c_{ij} \mathcal{C}_{ij}(\boldsymbol{\theta}^{(i)}) , \quad (12)$$

This expression is more general than the expression in Eq. (10), which consisted in a linear combination of *trace-preserving* maps, and, as such, rescaled the trace of every matrix ρ by a fixed amount $\lambda := \sum_{i=1}^N c_i$. In contrast, Eq. (12) is a linear combination of *trace non-increasing* maps, and, as such, it can be an arbitrary HP map, without any further

constraint on the trace. Hence, the extension to quantum instruments provides a further enlargement of the optimization space.

Finally, it is worth mentioning that a special case of the VQNN model is to use quantum circuits that act non-trivially only on a subsystem of the total quantum system. In this case, the quantum circuits are used to compute the expectation value

$$x(\boldsymbol{\theta}, \mathbf{c}) := \sum_{i=1}^N \sum_{j=1}^M c_{ij} \text{Tr} \left[O \left(C_{ij}(\boldsymbol{\theta}^{(i)}) \otimes \mathcal{I}_R \right) (\rho_0) \right], \quad (13)$$

where the input quantum system is divided into two subsystems Q and R , with the identity channel $\mathcal{I}_R$ acting on subsystem R , and the sampled quantum instruments acting on subsystem Q , transforming it into a new system Q' . By choosing ρ_0 and O to be an arbitrary state of system QR and an arbitrary observable on system $Q'R$, this scheme allows one to realize every possible test on the HP map $\mathcal{M}(\boldsymbol{\theta}, \mathbf{c})$ [53], thereby allowing for a broader set of tasks compared to the basic formulation without the reference system.

III. APPLICATIONS

In this section we apply the VQNN framework to three different problems, including two supervised learning tasks (quantum noise mitigation and quantum classification) and an unsupervised learning task (ground-state energy estimation). For simplicity, we will consider VQNNs realized by sampling over $N = 2$ quantum circuits, which is already sufficient to achieve non-trivial Hermitian-preserving maps [33, 51]. In all these applications, we will use the *QUAIRKIT* platform [54] to conduct numerical experiments to demonstrate the advantage of VQNN on training quantum neural networks with noisy quantum devices.

A. Mitigating unknown quantum noise

Quantum error mitigation [33, 51, 52] is a promising approach to estimate the expectation values of quantum observables in the presence of noise. The existing protocols, however, require the action of the noise to be known, either *a priori* or via quantum process characterization. Here we use the VQNN framework to develop a quantum error mitigation algorithm that can be used in the presence of unknown, uncharacterized noise.

1. Standard quantum error mitigation

Estimating the expectation values of quantum observables is a key step in several quantum algorithms [24, 55, 56]. In the ideal scenario, the expectation value $\text{Tr}[O\rho]$ of a given observable O on a given quantum state ρ is estimated by directly performing measurements on the state ρ . In practice,

however, quantum noise is inevitable, and one has access to a noisy version of the quantum state ρ , denoted by $\mathcal{N}(\rho)$ where $\mathcal{N} : L(\mathcal{H}_A) \rightarrow L(\mathcal{H}_B)$ is a CPTP map. Then, the problem of quantum error mitigation is to estimate the expectation value $\text{Tr}[O\rho]$ by performing measurements on the noisy state $\mathcal{N}(\rho)$.

The key idea of quantum error mitigation is to invert the action of the noise, by applying the inverse map $\mathcal{N}^{-1}$. This map is generally Hermitian-preserving and trace-preserving (HPTP) map, but not CPTP. As such, it cannot be implemented directly on a quantum device, but it can be simulated by sampling and postprocessing. To simulate the map $\mathcal{N}^{-1}$, one decomposes it into a linear combination of CPTP maps

$$\mathcal{N}^{-1} = \sum_{i=1}^N c_i \mathcal{C}_i, \quad (14)$$

where c_i are real coefficients and $\mathcal{C}_i$ are CPTP maps [33, 49, 50]. Then, the map $\mathcal{N}^{-1}$ is simulated by sampling over the CPTP maps $\mathcal{C}_1, \dots, \mathcal{C}_N$, and by postprocessing the expectation values to compute the linear combination.

The optimal error mitigation protocol corresponds to the decomposition that minimizes the sampling overhead, and can be found by solving the following minimization problem:

$$\min_{c_i, \mathcal{C}_i} \left\{ \|\mathbf{c}\|_1 \mid \sum_i c_i \mathcal{C}_i = \mathcal{N}^{-1}, c_i \in \mathbb{R}, \mathcal{C}_i \in \text{CPTP} \right\}, \quad (15)$$

where CPTP denotes the set of all CPTP maps from $L(\mathcal{H}_B)$ to $L(\mathcal{H}_A)$. It is important to note that the optimization problem in Eq. (15) is defined in terms of a given CPTP map $\mathcal{N}$, which specifies the action of the noise. In the following, we will extend the framework of quantum error mitigation to the scenario where the map $\mathcal{N}$ is initially unknown, corresponding to an uncharacterized quantum noise.

2. Variational quantum error mitigation

We now use the VQNN framework to develop variational algorithms for mitigation of an initially unknown noise process. The first step to construct such a protocol is to rephrase the optimization problem in Eq. (15) in a way that facilitates variational optimization. We start by observing that the error mitigation condition $\mathcal{N}^{-1} = \sum_i c_i \mathcal{C}_i$ is equivalent to $\mathcal{I} = \mathcal{N}^{-1} \circ \mathcal{N} = \sum_i c_i \mathcal{C}_i \circ \mathcal{N}$, which in turn is equivalent to

$$\Phi^+ = \sum_i c_i (\mathcal{I}_A \otimes \mathcal{C}_i \circ \mathcal{N})(\Phi^+), \quad (16)$$

where $\Phi^+ := |\Phi^+\rangle\langle\Phi^+|$ is the projector onto the Bell state $|\Phi^+\rangle = \sum_{i=1}^{d_A} |i\rangle \otimes |i\rangle / \sqrt{d_A}$, and d_A is the dimension of system's A Hilbert space. The derivation of Eq. (16) is given in Appendix B.

Eq. (16) guarantees a perfect inversion of the noise, corresponding to an ideal error mitigation. In practice, however, a perfect inversion may not be possible, either because the map $\mathcal{N}$ is not invertible, or because the operations accessible on the given quantum hardware do not include the channels $(\mathcal{C}_i)_{i=1}^N$

needed to invert the noise. In this setting, Eq. (16) can only hold approximately, and the goal is to find the best approximation achievable with the existing hardware. To assess the quality of an approximation in Eq. (16), one can use a measure of distance between the ideal target state Φ^+ and the operator

$$\sigma := \sum_i c_i \sigma_i, \quad \sigma_i := (\mathcal{I}_A \otimes \mathcal{C}_i \circ \mathcal{N})(\Phi^+). \quad (17)$$

In general, this operator may not be positive, and therefore may not be a quantum state. We can regard it as a “virtual quantum state,” which can be simulated by random sampling and classical data processing.

Many choices of distance measures are possible, including for example the fidelity and the diamond norm. For variational optimization, however, it is important to choose a distance measure that can be easily estimated from experimental data, so that one can feed the estimate of the distance into the classical algorithm that optimizes the parameters of the channels $(\mathcal{C}_i)_{i=1}^N$. For this purpose, a suitable choice is provided by the Hilbert-Schmidt (HS) distance, used in earlier works on quantum process tomography [57–59] and, more recently, in quantum state learning and purification [15]. Mathematically, the HS distance between the operators Φ^+ and σ is

$$\begin{aligned} \|\Phi^+ - \sigma\|_2^2 &:= \text{Tr}[(\Phi^+ - \sigma)^2] \\ &= \text{Tr}[(\Phi^+)^2] + \text{Tr}[\sigma^2] - 2 \text{Tr}[\Phi^+ \sigma] \\ &= 1 + \sum_{i,j=1}^N c_i c_j \text{Tr}[\sigma_i \sigma_j] - 2 \sum_{i=1}^N c_i \text{Tr}[\Phi^+ \sigma_i]. \end{aligned} \quad (18)$$

Crucially, the overlaps $\text{Tr}[\sigma_i \sigma_j]$ and $\text{Tr}[\Phi^+ \sigma_i]$ can be empirically estimated from quantum measurements. Specifically, for the first overlap $\text{Tr}[\sigma_i \sigma_j]$, one can be estimated by applying the swap test [55, 60, 61] on the states σ_i and σ_j for arbitrary $i, j \in \{1, \dots, N\}$. Moreover, if the control SWAP operation is unavailable, one can estimate the overlap by local operation and classical operations instead [62]. For the other overlap, $\text{Tr}[\Phi^+ \sigma_i]$, the value can also be estimated from the correlations of local Pauli measurements, using the formula $\text{Tr}[\Phi^+ \sigma_i] = (1 + \text{Tr}[(X \otimes X) \sigma_i] - \text{Tr}[(Y \otimes Y) \sigma_i] + \text{Tr}[(Z \otimes Z) \sigma_i])/4$ (this formula follows from the decomposition of the Bell state as a linear combination of Pauli products, namely $\Phi^+ = (I \otimes I + X \otimes X - Y \otimes Y + Z \otimes Z)/4$).

We are now ready to formulate the error mitigation task in the VQNN framework. The basic scenario is illustrated in Fig. 2. Here, we let the channels $(\mathcal{C}_i)_{i=1}^N$ depend on variational parameters $(\theta_i)_{i=1}^N$, respectively, and train our VQNN to minimize the approximation error. We propose two algorithms, corresponding to two different ways to include the sampling overhead into the optimization. The first algorithm is based on the minimization of the loss function

$$\mathcal{L}_{\text{VQEM}}^{(1)}(\theta, c) = \|c\|_1 + \alpha \|\Phi^+ - \sigma(\theta)\|_2^2, \quad (19)$$

where $\sigma(\theta) := \sum_i c_i (\mathcal{I}_A \otimes \mathcal{N}_i(\theta_i))(\Phi^+)$ is the virtual state produced by the VQNN, and $\alpha \geq 0$ is a weight determining

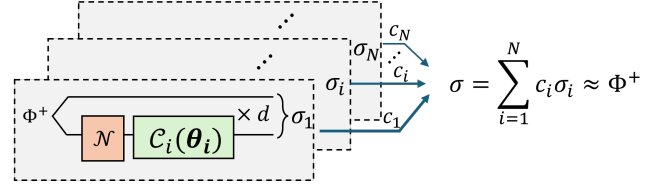


Fig 2. **Variational quantum error mitigation protocol.** Multiple parametrized quantum channels $\mathcal{C}_1(\theta_1), \dots, \mathcal{C}_N(\theta_N)$ are randomly sampled, producing output states $\rho_1, \dots, \rho_N$, respectively. At the postprocessing stage, the output states are combined linearly with coefficients $(c_i)_{i=1}^N$, yielding the virtual state $\sigma = \sum_{i=1}^N c_i \rho_i$. The goal of the variational protocol is to find parameters such that the virtual state σ is close to the Bell state Φ^+ .

the relative importance of the sampling overhead $\|c\|_1$ and the approximation error $\|\Phi^+ - \sigma(\theta)\|_2$. In the following, we will refer to α as the *penalty strength*, as it quantifies how strong is the penalty associated to an approximate realization of the error mitigation condition.

The second algorithm aims at directly minimizing the approximation error, and corresponds to the loss function

$$\mathcal{L}_{\text{VQEM}}^{(2)}(\theta) = \|\Phi^+ - \sigma(\theta)\|_2^2. \quad (20)$$

In this case, the minimization is performed under the constraint that the sampling overhead must not exceed a given value γ_* , hereafter called the *sampling overhead budget*. Mathematically, this constraint amounts to the fact that the coefficients c are restricted to be in the N -dimensional hypercube defined by the inequality $\sum_{i=1}^N |c_i| \leq \gamma_*$. Explicit descriptions of the two algorithms in pseudo-code are provided in Appendix C.

In the following we apply the general framework of variational quantum error mitigation to two special cases, corresponding to the mitigation of single-qubit noisy channels using two different forms of the parametrized channels $(\mathcal{C}_i(\theta_i))_{i=1}^N$. In the first case, we take the parametrized channels to be a subset of single-qubit channels generated by bounded-depth unitary gates acting on a limited number of qubits. This scenario models the situation in which a small-size, but otherwise ideal quantum circuit is used to mitigate an initially unknown error. In the second case we consider the realistic case in which also the realization of the quantum circuit is subject to noise. We take these two cases as an opportunity to illustrate our two variational algorithms, corresponding to the two loss functions (19) and (20), respectively.

3. VQEM with bounded-size unitary circuits

Here we consider the problem of mitigating an initially unknown single-qubit noisy channel $\mathcal{N}$. To this purpose, here we will use our first VQEM algorithm, designed to minimize the loss function (19) with a given penalty strength α . To test the performance of our algorithm, we pick the noisy channel $\mathcal{N}$ from two possible families channels: depolarizing channels,

corresponding to CPTP maps of the form

$$\mathcal{D}_p(\rho) = (1-p)\rho + p \frac{I_d}{d}, \quad (21)$$

(where d is the dimension of the system and I_d is the $d \times d$ identity matrix), and amplitude damping channels, corresponding to CPTP maps of the form

$$\mathcal{A}_q(\rho) = A_0 \rho A_0^\dagger + A_1 \rho A_1^\dagger \quad (22)$$

with $A_0 \equiv |0\rangle\langle 0| + \sqrt{1-q}|1\rangle\langle 1|$, and $A_1 \equiv \sqrt{q}|0\rangle\langle 1|$. In the above expressions the parameters p and q range between 0 and 1 and quantifies the noise level.

To mitigate the noise, we use parametrized channels obtained by running bounded-size unitary circuits on the target qubit and on a bounded number of auxiliary qubits. Explicitly, we consider parametrized channels of the form

$$C_i(\theta_i)(\rho) = \text{Tr}_{A_1, \dots, A_{n-1}} \left[\mathcal{U}_i(\theta_i) (\rho \otimes |0\rangle\langle 0|^{\otimes(n-1)}) \right], \quad (23)$$

where $A_1, \dots, A_{n-1}$ are $n-1$ auxiliary qubits, all initialized in the default state $|0\rangle$, and, for each i , $\mathcal{U}_i(\theta_i)$ is an n -qubit unitary channel a strong entangled layer structure [13, 14, 19, 63, 64], corresponding to a unitary gate decomposed into a sequence of d layers, with each layer containing only single qubit rotations about the X and Z axes, and CNOT gates on pairs of neighboring qubits. For $n = 3$, the strong entangled layer structure is illustrated in Fig. 3 (a).

In our numerical experiments, we set the number of qubits to $n = 3$, the number of layers to $d = 10$, and the number of parametrized channels to $N = 2$. The first set of experiments consists in the minimization of the loss function in Eq. (19), setting the penalty strength to $\alpha = 1000$. The results are shown in the TABLE I, where we show the values of approximation error and the sampling overhead for the optimal variational circuit. The approximation error, quantified by the HS norm, appears to be generally small, showing that the strong entangled layer structure is an effective ansatz for variational error mitigation of depolarizing and amplitude damping channels.

It is also interesting to observe that sampling overhead of our variational algorithms is comparable to the absolute minimum of the sampling overhead when the minimization is run over the set of all quantum channels (see Appendix D for more details). In other words, the circuits identified by our VQEM algorithm are nearly optimal in terms of sampling overhead.

It is worth noting that the quality of the approximation decreases for higher values of the noise levels p and q . However, this issue can be fixed by tuning the penalty strength α , thereby increasing the weight of the HS norm in the loss function. In Fig. 4, we analyze the case of on amplitude damping with noise level $q = 0.3$, showing the approximation error and the sampling overhead as functions of the penalty strength. As anticipated, the approximation error decreases monotonically with the penalty strength. Notably, also sampling overhead obtained by running VQEM still converges to the minimum sampling overhead over all possible quantum channels.

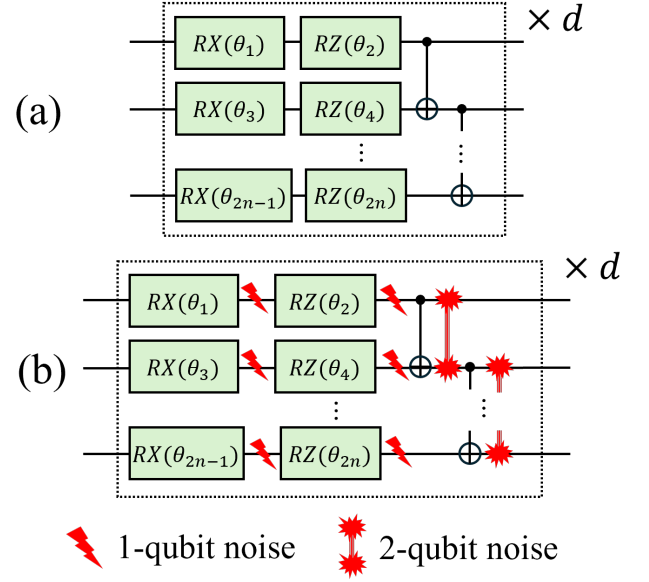


Fig 3. **Variational circuits for error mitigation.** (a) A noiseless ansatz for the PQC is constructed with rotation-X gates, rotation-Z gates and control-NOT gates. (b) A noisy ansatz of PQC is constructed by letting the single-qubit gates and two-qubit gates be followed by single-qubit and two-qubit noise processes, respectively.

4. VQEM with bounded-size noisy circuits

We now extend our numerical experiments to the scenario in which the parametrized quantum circuits are also subject to noise. As the task, we consider the mitigation of single-qubit depolarizing noise. For the variational optimization, we use our second VQEM algorithm, corresponding to the minimization of the approximation error for a fixed amount of sampling overhead.

To mitigate the noise, we use parametrized channels of the form

$$C_i(\theta_i)(\rho) = \text{Tr}_{A_1, \dots, A_{n-1}} \left[\tilde{\mathcal{U}}_i(\theta_i) (\rho \otimes |0\rangle\langle 0|^{\otimes(n-1)}) \right], \quad (24)$$

where $A_1, \dots, A_{n-1}$ are $n-1$ auxiliary qubits, all initialized in the default state $|0\rangle$, and, for each i , $\tilde{\mathcal{U}}_i(\theta_i)$ is an n -qubit noisy channel, corresponding to a noisy quantum circuit with a strong entangled layer structure. To model noisy circuit, we consider an ideal unitary circuit with strong entangled layer structure, and append a noisy channel to each single-qubit and each two-qubit gate in the circuit, as it was often done in the recent literature [49, 65, 66]. For $n = 3$, a noisy circuit with strong entangled layer structure is illustrated in Fig. 3 (b).

In the specific numerical experiments performed here, we set the noise on single-qubit gates and two-qubit gates to be depolarizing noise with noise levels $p = 0.01$ and $p = 0.02$, respectively. For the variational circuits, we set the total number of qubits to $n = 3$, the number of layers to $d = 5$, and the number of variational channels to $N = 2$. We run our second VQEM algorithm, which minimizes the HS distance with a

Noise model	Depolarizing $\mathcal{D}_p$			Amplitude Damping $\mathcal{A}_q$		
Noise level p	0.1	0.2	0.3	0.1	0.2	0.3
HS norm	9.53×10^{-7}	1.33×10^{-6}	1.55×10^{-6}	2.62×10^{-6}	3.34×10^{-6}	4.53×10^{-6}
Sampling overhead by VQEM	1.1667	1.372	1.644	1.222	1.500	1.857
Minimum sampling overhead	1.1667	1.375	1.643	1.218	1.49467	1.849

TABLE I. **Approximation error and sampling overhead of VQEM on depolarizing and amplitude damping noise.** The approximation error (quantified by the HS norm) and the sampling overhead are shown for different values of the noise level. For comparison, we also include the minimum value of the sampling overhead when the minimization is carried over the set of all single-qubit quantum channels.

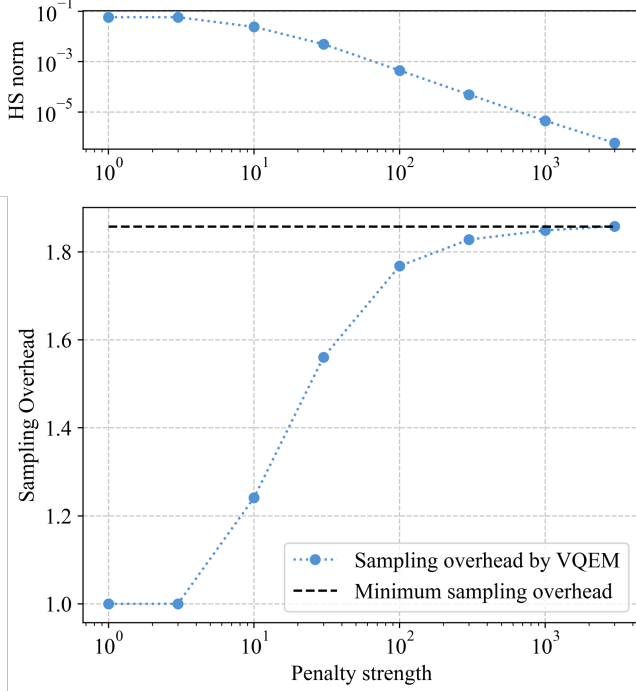


Fig 4. **Dependence of the approximation error and sampling overhead on the penalty strength.** *Top:* relation between the penalty strength and the approximation error, quantified by the Hilbert-Schmidt distance between the virtual state σ and target Bell state Φ^+ . *Bottom:* relation between the penalty strength and the sampling overhead (blue dotted line). The minimum sampling overhead over all possible quantum channels is shown by the black dashed line.

fixed sampling overhead selected in the range of $[1, 2.5]$. To test the performance of the algorithm we set the channel to be mitigated to a depolarizing channel with noise level $p = 0.2$.

The numerical results are shown in Fig. 5, where we compare the optimized HS norm with given sampling overhead budgets by noiseless circuits and noisy circuits. Both HS norms optimized by noisy circuits and noiseless circuits converge to zero with the increase of sampling overhead budget, indicating that the variational form of our circuits still permits a nearly perfect error mitigation.

It is worth observing that, naturally, the HS norm obtained by the noisy circuit is larger than that of the noiseless circuit. Remarkably, however, the noisy circuit can nearly reach the same error level of the noiseless one when we allow a larger sampling overhead. Overall, this result shows an important

feature of the VQNN model: random sampling and classical postprocessing can be used to reproduce the performance of an ideal noiseless circuit that is not directly accessible on a realistic quantum hardware.

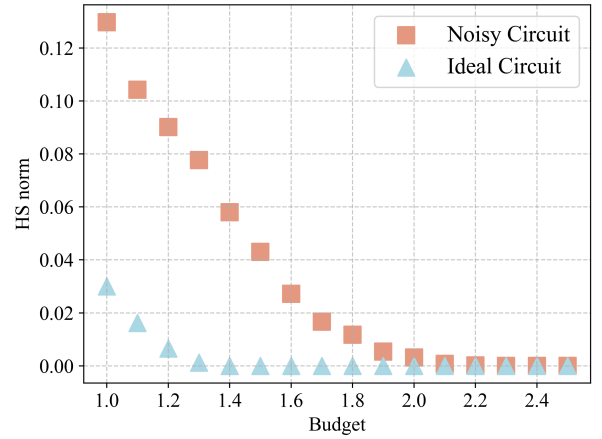


Fig 5. **Approximation error of VQEM on bounded-size noisy circuits.** The approximation error, quantified by the HS norm, is shown as a function of the sampling overhead budget in the range $[1, 2.5]$. The orange squares (light blue squares) show the results achieved by the VQEM algorithm run on bounded-size noisy (noiseless) circuits.

B. Quantum Classification

Quantum classifiers [25, 26] are a central model of supervised quantum machine learning. The input for the classification task is a data set consisting of labeled samples $D = \{\mathbf{x}_i, y_i\}_{i=1}^M$ where M is the data size, $\mathbf{x}_i$ is a vector, representing a *data point*, and y_i is the corresponding label. In the binary classification task, the labels are 0 and 1, i.e., $y_i \in \{0, 1\}$. The first step for the quantum classifier is to encode these input data into quantum states. The details on how the encoding is done are summarized in Appendix E. Once the encoding has been completed, the classification problem can be viewed as the task of assigning an appropriate label to an unseen quantum state, based on the information gained from previous training data.

Here, we apply the VQNN framework to the classification task, developing a new algorithm called the *virtual quantum*

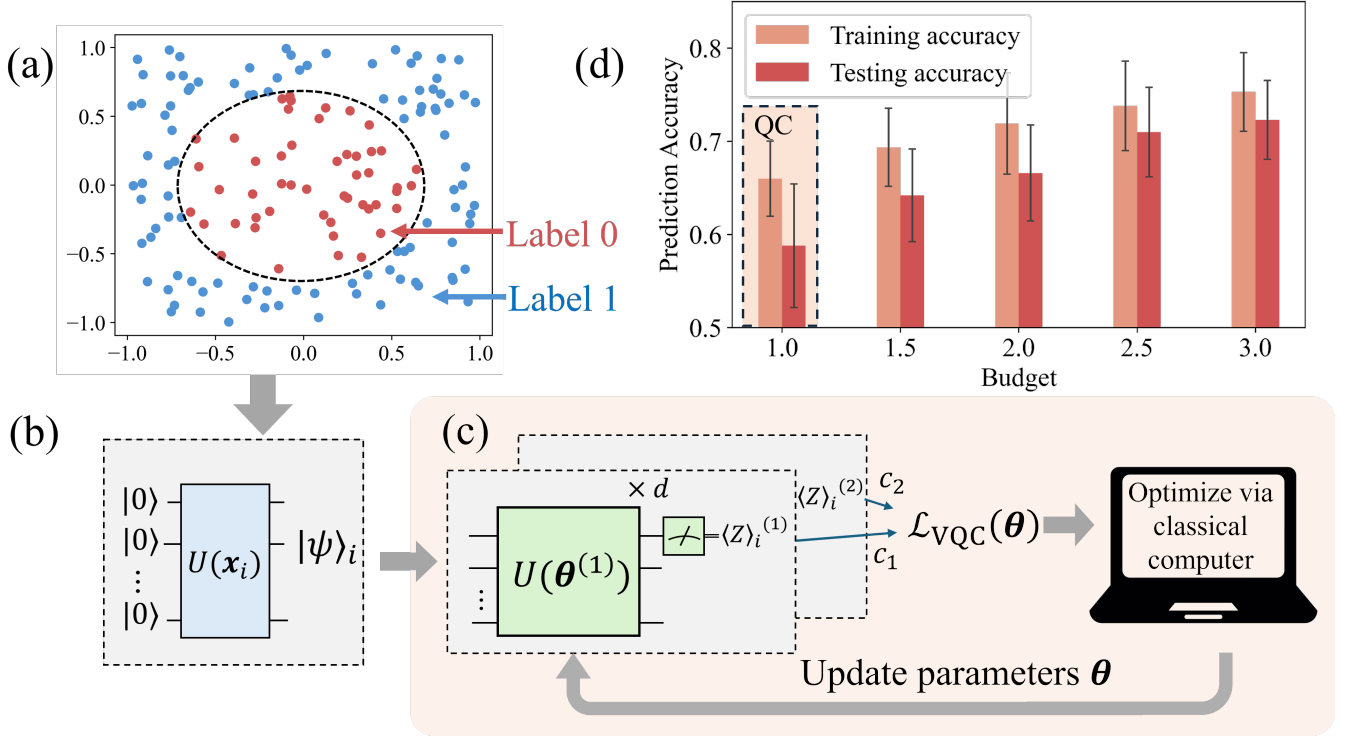


Fig 6. Virtual quantum classifier. (a) Training Data. In this example, the input vectors lie in the 2-dimensional plane, and are labeled with 0 and 1. (b) Data Encoding. The classical data are encoded into quantum state before training the virtual quantum classifier (c) Training virtual quantum classifier. Implement PQCs and make measurements to estimate the loss function on the quantum devices. Then optimize the loss function with classical computers by tuning the parameters in the PQCs. After iterations, the loss function converge to its minimum and out put the optimized virtual quantum classifier (d) Prediction accuracy by optimized VQC with different sampling overhead budgets. The prediction accuracy with the budget equal to 1.0 corresponds to the conventional quantum classifier.

classifier (VQC). Compared to conventional quantum classifiers, the VQC offers an advantage in the presence of unknown quantum noise affecting the encoding operations, as it can suitably adapt its classification policy.

Within the virtual quantum classifier paradigm, the classifier f_θ , which is a combination of QNNs with parameters θ and measurements, is utilized to predict the label $\tilde{y}_i$. A good quantum classifier f_θ should map data to their corresponding labels as accurately as possible $f_\theta(x_i) \rightarrow y_i$. Therefore, in the training procedure, we use the average distance between the actual label y_i and the predicted label $\tilde{y}_i$ as the loss function, which is

$$\mathcal{L}_{\text{VQC}}(\theta) = \frac{1}{M} \sum_{i=1}^M |\tilde{y}_i - y_i|^2. \quad (25)$$

The detailed procedure of training a virtual quantum classifier is shown in Appendix F.

A well trained classifier f_{θ^*} should also predict unseen data accurately. Thus, we need a testing data set to verify the performance of the trained classifier f_{θ^*} . To demonstrate the power of VQC on noisy quantum device, we conduct a numerical experiment. The task is binary classification, and the visualized training data set is shown in Fig. 6 (a). The training set contains 200 two-dimensional data points, and the red(blue) points represent the labels of 0(1). The quantum gates in the

ansatz of VQC are noisy, which are modeled as the ideal gates followed by noisy channels as shown in Fig. 3 (b), and the depth is set as 2. Specifically, the single-qubit rotation gates is followed by single-qubit depolarizing channels with noise level $p = 0.02$; the two-qubit rotation gates are followed by two-qubit depolarizing channels with noise level $p = 0.05$. After training, we randomly generate a testing data set with 100 data points, which have the same features as the training data set. Then we apply the trained classifier f_{θ^*} to the testing data to verify the performance of the classifier when encountering the unseen data. The numerical results are shown in Fig. 6 (d). The error bar refers to the standard deviation by repeating 10 times. Note that the results with a budget equal to 1.0 (red shaded area denoted as QC) correspond to the conventional quantum classifier. It can be seen that the prediction accuracy for both the training data set and the testing data set increases with the increase of budgets, demonstrating the advantage of VQNNs.

C. Estimating ground-state energy

Estimating the ground state energy of a Hamiltonian is a fundamental task in quantum physics [67], chemistry [68], and conductivity and magnetism [69], as it determines the

lowest possible energy and thus the stability and physical properties of a system. In quantum computing, the variational quantum eigensolver (VQE) can be used to estimate the ground state energy of a Hamiltonian H on near-term quantum devices, which is an unsupervised learning task. The loss function is $\mathcal{L}_{VQE}(\theta) = \text{Tr}[H|\psi(\theta)\rangle\langle\psi(\theta)|]$, where one aims to minimize the expectation value of the Hamiltonian H over a trial state $|\psi(\theta)\rangle = U(\theta)|0\rangle$ for some parametrized ansatz $U(\theta)$ and initial state $|0\rangle$. In practice, the Hamiltonian is typically expressed as a linear combination of tensor products of Pauli operators, $H = \sum_k h_k P_k$, where h_k are real coefficients. Consequently, the loss function $\mathcal{L}_{VQE}(\theta)$ can be evaluated as a weighted sum of the expectation values of the Pauli operators, which can be estimated on quantum devices.

When implementing VQE on quantum devices, however, noise is unavoidable, corrupting the computing and preventing us from accessing the true ground state energy. Here, we exploit the VQNN framework and propose a new algorithm called VVQE to solve this problem.

1. Loss function

We consider the scenario where two branches of PQCs are utilized, and the sampling overhead budget $\gamma_b = |c_1| + |c_2|$ is given, where $c_1 = (1 + \gamma_b)/2$ and $c_2 = (1 - \gamma_b)/2$ are real numbers. The output average state is $\sigma = c_1\rho_1 + c_2\rho_2$, and $\rho_i = U_i(\theta^{(i)})|0\rangle\langle 0|U_i^\dagger(\theta^{(i)})$, $i \in \{1, 2\}$. In this task, we aim to obtain the lowest energy of the Hamiltonian H . In the conventional VQE algorithm, it solves an optimization problem of $E^* = \min_\rho \text{Tr}[H\rho]$, where ρ is a quantum state. The constraint of the quantum state guarantees that the optimal value E^* is the lowest energy of the Hamiltonian H , and the corresponding state ρ^* is the ground state.

In our case, σ is the averaged state, which is a general Hermitian operator. If we simply set the loss function as $\text{Tr}[H\sigma]$, then the optimal value will be $c_2E_\infty + c_1E_0$ if $\gamma_b \geq 1$, where E_0 and E_∞ are the minimum and maximum energies of Hamiltonian H , respectively. Therefore, we should force the averaged state σ as a valid state, i.e., $\text{Tr}[\sigma] = 1$ and $\sigma \geq 0$. For the unit trace constraint, it is satisfied directly by setting $c_2 = 1 - c_1$. For the positive semi-definite constraint, we could take the negativity of the average state into the loss function $\text{Tr}[|\sigma| - \sigma]/2 = \sum_{\lambda_i \leq 0} |\lambda_i|$, which is the summation of all the absolute value of the negative eigenvalues λ_i of the average states σ . When minimizing the negativity, it will converge to 0, making all eigenvalues of the average state nonnegative, i.e., σ is positive semidefinite. Thus, the natural choice of the loss function is

$$\mathcal{L}_{VVQE}(\theta) = \text{Tr}[H\sigma] + \alpha \frac{\text{Tr}[|\sigma| - \sigma]}{2}, \quad (26)$$

where α is the penalty strength. However, estimating the negativity of a quantum state on a quantum device is generally hard, implying that such a loss function is not practical. Here, instead of utilizing the negativity directly, we approximate it with a K -truncated Chebyshev expansion. Recalling the expression of the normalized average state $\tilde{\sigma} = \sigma/|c_1|$ (so that

$\sigma = c_1 \tilde{\sigma}$, with $c_1 = |c_1| > 0$), the negativity reads

$$\frac{\text{Tr}[|\sigma| - \sigma]}{2} = c_1 \text{Tr}[f(\tilde{\sigma})], \quad f(\tilde{\sigma}) := \frac{|\tilde{\sigma}| - \tilde{\sigma}}{2}. \quad (27)$$

The details of the Chebyshev approximation of $f(\tilde{\sigma})$ by the truncated polynomial $p_K(\tilde{\sigma})$ are given in Appendix G. Approximating $\text{Tr}[f(\tilde{\sigma})] \simeq \text{Tr}[p_K(\tilde{\sigma})]$ then leads to the loss function

$$\begin{aligned} \mathcal{L}_{VVQE}(\theta) &= \text{Tr}[H\sigma] + \alpha c_1 \text{Tr}[p_K(\tilde{\sigma})] \\ &= c_1 \text{Tr}[H\rho_1] + c_2 \text{Tr}[H\rho_2] + \alpha c_1 \text{Tr}[p_K(\tilde{\sigma})]. \end{aligned} \quad (28)$$

With this loss function, we estimate the ground-state energy with the procedure shown in Fig. 1. The detailed processes are shown in Appendix H.

2. Estimating the ground-state energy of HeH^+

Here we conduct the numerical experiment to estimate the ground-state energy of HeH^+ on a noisy quantum device with the VVQE algorithm. Conduct Algorithm 4 by setting the noisy PQCs as shown in Fig. 3 (b) with depth $d = 5$. The truncation number K is set as 100. The PQCs noise is set as single-qubit depolarizing and two-qubit depolarizing noise with noise level $p = 0.01$ and $p = 0.02$, respectively.

The numerical results are shown in Fig. 7. The dark blue dashed line is the error-free ground state energy. The light blue squares correspond to the energy estimation via conventional VQE with noisy circuits. The orange stars and red triangles are obtained by the proposed virtual VQE with budgets $\gamma = 1.5$ and $\gamma = 2.5$, respectively. Obviously, an accurate ground state energy can be estimated with a reasonably large budget.

IV. DISCUSSION AND CONCLUSIONS

In this work we introduced the framework of virtual quantum neural networks, leveraging random sampling and classical data processing to enhance the expressivity and performance of quantum circuits in the NISQ regime. To illustrate the benefits of our framework, we applied it to three representative tasks, including both supervised and unsupervised learning. Compared with the conventional QNN, VQNN shows advantages in two aspects: enhanced expressivity and improved noise robustness.

The enhanced expressivity arises from an enlarged optimization space. Conventional QNNs restrict the optimization to a single parametric family of quantum circuits, represented by completely positive maps. VQNNs, instead, use sampling over multiple quantum circuits to achieve a larger optimization space, mathematically described by a family of Hermitian-preserving maps. Overall, one can view VQNNs as a way to overcome some of the limitations of the existing

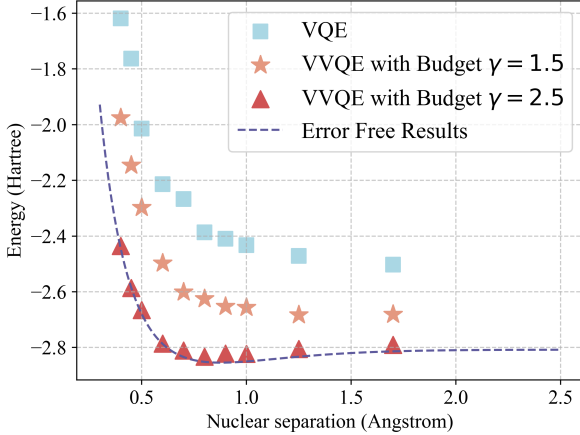


Fig 7. **Ground state energy of the molecule HeH^+ estimated via the VVQE algorithm.** The dashed line shows the exact values of the ground state energy as a function of the nuclear separation. The orange stars and red triangles show the energy values found by our VVQE algorithm for different values of the sampling budgets, set to $\gamma = 1.5$ and $\gamma = 2.5$, respectively. The light blue squares show the energies computed with the conventional VQE, which is equivalent to a VVQE with unit budget $\gamma = 1$.

quantum hardware by investing additional resources in sampling and classical computation at the training stage.

A promising direction is the application of the VQNN framework to scenarios that require long-range quantum entanglement. In a conventional QNN, establishing long-range entanglement requires quantum circuits of large depth [70, 71], which however tend to be affected by barren plateaus. Shallow circuits, instead, are gradient-healthy, but their expressive power is limited by their causal cones, which cannot support the generation of long-range entanglement. In contrast, linear combinations of shallow quantum circuits could be used to virtually simulate long-range entanglement [72–74]. This intuition is made rigorous by the observation that arbitrary non-local quantum operations can be simulated by linear combinations of local operations and classical commu-

nication, as shown in Appendix I.

Another key feature is the robustness of VQNNs against hardware noise. In conventional QNN, noise accumulates with circuit depth, leading to a loss of signal-to-noise ratio. In contrast, VQNN can partially cancel noise by creating linear combinations of multiple quantum circuits. Since the optimal coefficients in the linear combination are learned in the presence of noise, the optimization process naturally functions as a variational protocol for probabilistic error cancellation [33, 48, 51].

The above observations suggest that the sampling and classical postprocessing serve VQNNs with a dual purpose: on the one hand, they act as a resource to bypass the connectivity constraints of the available quantum hardware and, on the other hand, they act as a buffer that absorbs noise. In this way, the model’s expressivity is freed from some of the key limitations of the existing hardware, including limitations both at the logical level (restricted set of logical operations) and at the physical level (decoherence and hardware imperfections).

Acknowledgements.—B. Z would like to thank Mingrui Jing, Xiao Shi and Ya-dong Wu for valuable discussions. This work was supported by the Chinese Ministry of Science and Technology (MOST) through grant 2023ZD0300600, and by the Hong Kong Research Grant Council (RGC) through grants SRFS2021-7S02 and R7035-21F, and the State Key Laboratory of Quantum Information Technologies and Materials. Yingzhou Li is supported in part by the Scientific Research Innovation Capability Support Project for Young Faculty under grant SRICSPYF-ZY2025159, National Natural Science Foundation of China under grant 12526211, the Science and Technology Commission of Shanghai Municipality under grants 22TQ017. Yinan Li was supported by the National Key Research and Development Program of China under Grant 2024YFE0102500, the Fundamental Research Funds for the Central Universities under Grant 2042025kf0023, the National Nature Science Foundation of China under Grant 62302346 and the Hubei Provincial Nature Science Foundation of China under Grant 2024AFA045.

Data availability.—The data that support the findings of this article are openly available [75].

-
- [1] J. Preskill, Quantum computing in the nisq era and beyond, *Quantum* **2**, 79 (2018).
 - [2] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, *arXiv preprint arXiv:1411.4028* (2014).
 - [3] M. P. Harrigan, K. J. Sung, M. Neeley, K. J. Satzinger, F. Arute, K. Arya, J. Atalaya, J. C. Bardin, R. Barends, S. Boixo, *et al.*, Quantum approximate optimization of non-planar graph problems on a planar superconducting processor, *Nature Physics* **17**, 332 (2021).
 - [4] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, *Reviews of Modern Physics* **92**, 015003 (2020).
 - [5] Google AI Quantum and Collaborators, F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, S. Boixo, M. Broughton, B. B. Buckley, *et al.*, Hartree-fock on a superconducting qubit quantum computer, *Science* **369**, 1084 (2020).
 - [6] K. M. Nakanishi, K. Mitarai, and K. Fujii, Subspace-search variational quantum eigensolver for excited states, *Physical Review Research* **1**, 033062 (2019).
 - [7] C. H. Bennett and G. Brassard, Quantum cryptography: Public key distribution and coin tossing, *Theoretical computer science* **560**, 7 (2014).
 - [8] A. K. Ekert, Quantum cryptography based on Bell’s theorem, *Physical Review Letters* **67**, 661 (1991).
 - [9] K. Fujii, K. Mizuta, H. Ueda, K. Mitarai, W. Mizukami, and Y. O. Nakagawa, Deep variational quantum eigensolver: a divide-and-conquer method for solving a larger problem with

- smaller size quantum computers, *PRX Quantum* **3**, 010346 (2022).
- [10] X. Yuan, S. Endo, Q. Zhao, Y. Li, and S. C. Benjamin, Theory of variational quantum simulation, *Quantum* **3**, 191 (2019).
- [11] B. Zhao and K. Fujii, Variational quantum hamiltonian engineering, *Physical Review Research* **7**, 023123 (2025).
- [12] S. Endo, J. Sun, Y. Li, S. C. Benjamin, and X. Yuan, Variational quantum simulation of general processes, *Physical Review Letters* **125**, 010501 (2020).
- [13] X. Zhao, B. Zhao, Z. Wang, Z. Song, and X. Wang, Practical distributed quantum information processing with loccnet, *npj Quantum Information* **7**, 159 (2021).
- [14] R. Chen, B. Zhao, and X. Wang, Near-term efficient quantum algorithms for entanglement analysis, *Physical Review Applied* **20**, 024071 (2023).
- [15] R. Chen, Z. Song, X. Zhao, and X. Wang, Variational quantum algorithms for trace distance and fidelity estimation, *Quantum Science and Technology* **7**, 015019 (2021).
- [16] K. H. Wan, O. Dahlsten, H. Kristjánsson, R. Gardner, and M. Kim, Quantum generalisation of feedforward neural networks, *npj Quantum information* **3**, 36 (2017).
- [17] E. Farhi and H. Neven, Classification with quantum neural networks on near term processors, *arXiv preprint arXiv:1802.06002* (2018).
- [18] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, Quantum circuit learning, *Physical Review A* **98**, 032309 (2018).
- [19] M. Schuld, A. Bocharov, K. M. Svore, and N. Wiebe, Circuit-centric quantum classifiers, *Physical Review A* **101**, 032308 (2020).
- [20] I. Cong, S. Choi, and M. D. Lukin, Quantum convolutional neural networks, *Nature Physics* **15**, 1273 (2019).
- [21] Y. Du, X. Wang, N. Guo, Z. Yu, Y. Qian, K. Zhang, M.-H. Hsieh, P. Rebentrost, and D. Tao, Quantum machine learning: A hands-on tutorial for machine learning practitioners and researchers, *arXiv preprint arXiv:2502.01146* (2025).
- [22] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, Quantum machine learning, *Nature* **549**, 195 (2017).
- [23] P. Rebentrost, M. Mohseni, and S. Lloyd, Quantum support vector machine for big data classification, *Phys. Rev. Lett.* **113**, 130503 (2014).
- [24] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, *et al.*, Variational quantum algorithms, *Nature Reviews Physics* **3**, 625 (2021).
- [25] W. Li and D.-L. Deng, Recent advances for quantum classifiers, *Science China Physics, Mechanics & Astronomy* **65**, 220301 (2022).
- [26] E. Grant, M. Benedetti, S. Cao, A. Hallam, J. Lockhart, V. Stojevic, A. G. Green, and S. Severini, Hierarchical quantum classifiers, *npj Quantum Information* **4**, 65 (2018).
- [27] J. Romero, J. P. Olson, and A. Aspuru-Guzik, Quantum autoencoders for efficient compression of quantum data, *Quantum Science and Technology* **2**, 045001 (2017).
- [28] C. Cao and X. Wang, Noise-assisted quantum autoencoder, *Physical Review Applied* **15**, 054012 (2021).
- [29] D. Bondarenko and P. Feldmann, Quantum autoencoders to denoise quantum data, *Physical Review Letters* **124**, 130502 (2020).
- [30] X. Gao, Z.-Y. Zhang, and L.-M. Duan, A quantum machine learning algorithm based on generative models, *Science advances* **4**, eaat9004 (2018).
- [31] P.-L. Dallaire-Demers and N. Killoran, Quantum generative adversarial networks, *arXiv preprint arXiv:1804.08641* (2018).
- [32] W. Hoeffding, Probability inequalities for sums of bounded random variables, *Journal of the American statistical association* **58**, 13 (1963).
- [33] J. Jiang, K. Wang, and X. Wang, Physical implementability of linear maps and its application in error mitigation, *Quantum* **5**, 600 (2021).
- [34] B. Regula, R. Takagi, and M. Gu, Operational applications of the diamond norm and related measures in quantifying the non-physicality of quantum maps, *Quantum* **5**, 522 (2021).
- [35] X. Zhao, L. Zhang, B. Zhao, and X. Wang, Power of quantum measurement in simulating unphysical operations, *Physical Review Research* **7**, 013334 (2025).
- [36] J. Stokes, J. Izaac, N. Killoran, and G. Carleo, Quantum natural gradient, *Quantum* **4**, 269 (2020).
- [37] R. M. Parrish, J. T. Iosue, A. Ozaeta, and P. L. McMahon, A jacobi diagonalization and anderson acceleration algorithm for variational quantum algorithm parameter optimization, *arXiv preprint arXiv:1904.03206* (2019).
- [38] K. M. Nakanishi, K. Fujii, and S. Todo, Sequential minimal optimization for quantum-classical hybrid algorithms, *Physical Review Research* **2**, 043158 (2020).
- [39] M. Ostaszewski, E. Grant, and M. Benedetti, Structure optimization for parameterized quantum circuits, *Quantum* **5**, 391 (2021).
- [40] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* (2014).
- [41] M. Larocca, S. Thanasilp, S. Wang, K. Sharma, J. Biamonte, P. J. Coles, L. Cincio, J. R. McClean, Z. Holmes, and M. Cerezo, Barren plateaus in variational quantum computing, *Nature Reviews Physics* **1** (2025).
- [42] A. Peres, Separability criterion for density matrices, *Physical Review Letters* **77**, 1413 (1996).
- [43] M. Horodecki, P. Horodecki, and R. Horodecki, Separability of n-particle mixed states: necessary and sufficient conditions in terms of linear maps, *Physics Letters A* **283**, 1 (2001).
- [44] O. Gühne and G. Tóth, Entanglement detection, *Physics Reports* **474**, 1 (2009).
- [45] P. Pechukas, Reduced dynamics need not be completely positive, *Physical Review Letters* **73**, 1060 (1994).
- [46] K. Modi, Operational approach to open dynamics and quantifying initial correlations, *Scientific reports* **2**, 581 (2012).
- [47] D. Schmid, K. Ried, and R. W. Spekkens, Why initial system-environment correlations do not imply the failure of complete positivity: A causal perspective, *Physical Review A* **100**, 022112 (2019).
- [48] K. Temme, S. Bravyi, and J. M. Gambetta, Error mitigation for short-depth quantum circuits, *Physical Review Letters* **119**, 180509 (2017).
- [49] R. Takagi, Optimal resource cost for error mitigation, *Physical Review Research* **3**, 033178 (2021).
- [50] R. Takagi, S. Endo, S. Minagawa, and M. Gu, Fundamental limits of quantum error mitigation, *npj Quantum Information* **8**, 114 (2022).
- [51] X. Zhao, B. Zhao, Z. Xia, and X. Wang, Information recoverability of noisy quantum states, *Quantum* **7**, 978 (2023).
- [52] B. Zhao, M. Jing, L. Zhang, X. Zhao, Y.-A. Chen, K. Wang, and X. Wang, Retrieving nonlinear features from noisy quantum states, *PRX Quantum* **5**, 020357 (2024).
- [53] G. Bai and G. Chiribella, Test one to test many: a unified approach to quantum benchmarks, *Physical Review Letters* **120**, 150502 (2018).
- [54] QuAIR Team, QuAIRKit, <https://github.com/QuAIR/QuAIRKit> (2023).
- [55] M. A. Nielsen and I. L. Chuang, *Quantum computation and*

- quantum information* (Cambridge university press, 2010).
- [56] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O’Brien, A variational eigenvalue solver on a photonic quantum processor, *Nature Communications* **5**, 4213 (2014).
 - [57] A. J. Scott, Optimizing quantum process tomography with unitary 2-designs, *Journal of Physics A: Mathematical and Theoretical* **41**, 055308 (2008).
 - [58] A. Bisio, G. Chiribella, G. D’Ariano, S. Facchini, and P. Perinotti, Optimal quantum tomography of states, measurements, and transformations, *Physical Review Letters* **102**, 010404 (2009).
 - [59] A. Bisio, G. Chiribella, G. M. D’Ariano, S. Facchini, and P. Perinotti, Optimal quantum tomography, *IEEE Journal of Selected Topics in Quantum Electronics* **15**, 1646 (2009).
 - [60] A. Barenco, A. Berthiaume, D. Deutsch, A. Ekert, R. Jozsa, and C. Macchiavello, Stabilization of quantum computations by symmetrization, *SIAM Journal on Computing* **26**, 1541 (1997).
 - [61] H. Buhrman, R. Cleve, J. Watrous, and R. De Wolf, Quantum fingerprinting, *Physical Review Letters* **87**, 167902 (2001).
 - [62] A. Anshu, Z. Landau, and Y. Liu, Distributed quantum inner product estimation, in *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing* (2022) pp. 44–51.
 - [63] Y. Wang, G. Li, and X. Wang, Variational quantum gibbs state preparation with a truncated taylor series, *Phys. Rev. Appl.* **16**, 054035 (2021).
 - [64] G. Li, X. Zhao, and X. Wang, Quantum self-attention neural networks for text classification, *Science China Information Sciences* **67**, 142501 (2024).
 - [65] T. Scheiber, P. Haubenwallner, and M. Heller, Reducing pec overhead by pauli error propagation, *arXiv preprint arXiv:2412.01311* (2024).
 - [66] M. Liao, Y. Zhu, G. Chiribella, and Y. Yang, Noise-agnostic quantum error mitigation with data augmented neural models, *npj Quantum Information* **11**, 8 (2025).
 - [67] P. W. Anderson, An approximate quantum theory of the antiferromagnetic ground state, *Physical Review* **86**, 694 (1952).
 - [68] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, Elucidating reaction mechanisms on quantum computers, *Proceedings of the national academy of sciences* **114**, 7555 (2017).
 - [69] P. A. Lee, N. Nagaosa, and X.-G. Wen, Doping a mott insulator: Physics of high-temperature superconductivity, *Reviews of modern physics* **78**, 17 (2006).
 - [70] X.-G. Wen, Topological order: From long-range entangled quantum matter to a unified origin of light and electrons, *International Scholarly Research Notices* **2013**, 198710 (2013).
 - [71] X. Chen, Z.-C. Gu, and X.-G. Wen, Local unitary transformation, long-range quantum entanglement, wave function renormalization, and topological order, *Physical Review B—Condensed Matter and Materials Physics* **82**, 155138 (2010).
 - [72] K. Mitarai and K. Fujii, Constructing a virtual two-qubit gate by sampling single-qubit operations, *New Journal of Physics* **23**, 023021 (2021).
 - [73] K. Mitarai and K. Fujii, Overhead for simulating a non-local channel with local channels by quasiprobability sampling, *Quantum* **5**, 388 (2021).
 - [74] C. Piveteau and D. Sutter, Circuit knitting with classical communication, *IEEE Transactions on Information Theory* **70**, 2734 (2023).
 - [75] B. Zhao, Codes for Virtual Quantum Neural Network, <https://github.com/benchizhao/Virtual-Quantum-Neural-Network> (2026).
 - [76] M.-D. Choi, Completely positive linear maps on complex matrices, *Linear algebra and its applications* **10**, 285 (1975).
 - [77] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces, *Nature* **567**, 209 (2019).
 - [78] H.-P. Breuer and F. Petruccione, *The theory of open quantum systems* (OUP Oxford, 2002).
 - [79] R. LaRose and B. Coyle, Robust data encodings for quantum classifiers, *arXiv preprint arXiv:2003.01695* (2020).
 - [80] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven, and J. R. McClean, Power of data in quantum machine learning, *Nature Communications* **12**, 2631 (2021).
 - [81] K. Chen, Q. Wang, Z. Yu, and Z. Zhang, Simultaneous estimation of nonlinear functionals of a quantum state, *arXiv preprint arXiv:2505.16715* (2025).
 - [82] X. Shi, J. Jiang, X. Wu, J. Xie, H. Yao, and X. Wang, Near-optimal simultaneous estimation of quantum state moments, *arXiv preprint arXiv:2509.24842* (2025).
 - [83] B. Kraus and J. I. Cirac, Optimal creation of entanglement using a two-qubit gate, *Physical Review A* **63**, 062309 (2001).

Appendix for Virtual Quantum Neural Networks

Appendix A: Analysis of Barren Plateaus

The problem of barren plateaus (BPs) [41] arises when the gradients of the loss function $\mathcal{L}(\theta)$ vanish exponentially as the number of qubits n grows. Mathematically, a BP corresponds to the condition $\text{Var}[\partial_{\theta_\mu} \mathcal{L}(\theta)] \in \mathcal{O}(1/b^n)$, for some constant $b > 1$, and for some (not necessarily all) parameters $\theta_\mu \in \theta$. For many common parameterized quantum circuit architectures, the parameter space becomes uniformly explored across the exponentially large Hilbert space, making the cost landscape overwhelmingly flat and featureless. Consequently, optimizing the parameters becomes intractable as it requires an exponentially

large number of measurements to discern a reliable optimization direction, posing a fundamental challenge to training scalable QNNs.

We now analyze how the VQNN framework affects this phenomenon. For simplicity, we take the function f entering the loss function Eq. (8) to be the identity map, so that the VQNN loss becomes

$$\mathcal{L}_{\text{VQNN}}(\boldsymbol{\theta}, \mathbf{c}) = \sum_{i=1}^N c_i \text{Tr} \left[O U_i(\boldsymbol{\theta}^{(i)}) \rho_0 U_i^\dagger(\boldsymbol{\theta}^{(i)}) \right] =: \sum_{i=1}^N c_i \mathcal{L}_i(\boldsymbol{\theta}^{(i)}), \quad (\text{A1})$$

where $\mathcal{L}_i(\boldsymbol{\theta}^{(i)})$ is the loss function of the i -th branch. The gradient with respect to a parameter $\theta_\mu^{(i)}$ of the i -th branch is therefore $\partial_{\theta_\mu^{(i)}} \mathcal{L}_{\text{VQNN}}(\boldsymbol{\theta}, \mathbf{c}) = c_i \partial_{\theta_\mu} \mathcal{L}_i(\boldsymbol{\theta}^{(i)})$, and, since this gradient depends only on the i -th branch, its variance factorizes as

$$\text{Var} \left[\partial_{\theta_\mu^{(i)}} \mathcal{L}_{\text{VQNN}}(\boldsymbol{\theta}, \mathbf{c}) \right] = c_i^2 \text{Var} \left[\partial_{\theta_\mu} \mathcal{L}_i(\boldsymbol{\theta}^{(i)}) \right]. \quad (\text{A2})$$

Suppose now that each branch, taken on its own, exhibits a barren plateau, i.e., $\text{Var}[\partial_{\theta_\mu} \mathcal{L}_i] \in \mathcal{O}(1/b^n)$. The above relation then gives $\text{Var}[\partial_{\theta_\mu^{(i)}} \mathcal{L}_{\text{VQNN}}] = c_i^2 \mathcal{O}(1/b^n)$. As long as the coefficients are bounded, $|c_i| = \mathcal{O}(1)$, which is guaranteed by a bounded sampling overhead $\gamma = \sum_i |c_i|$, the variance remains exponentially small in n ; the coefficients only rescale it by a constant factor. In other words, a VQNN with bounded sampling overhead does not escape the barren plateau affecting its constituent branches, but it also does not aggravate it: the variance of each branch's gradient is of the same exponential order as for a standard QNN. Thus, the additional expressivity of the VQNN framework does not come at the price of a worse barren-plateau problem. Conversely, substantially mitigating a barren plateau would require $|c_i|$ to grow exponentially with n , which in turn entails an exponentially large sampling overhead and is therefore outside the regime of practical interest.

Appendix B: Derivation of Eq. (16)

In this Appendix we provide the derivation of Eq. (16) in the main text.

The starting point is to reformulate the optimization problem in Eq. (16) of the main text using the Choi correspondence between quantum processes and bipartite quantum states [76]. Mathematically, the Choi operator of a linear map $\mathcal{N} : L(\mathcal{H}_A) \rightarrow L(\mathcal{H}_B)$ is the operator $J^\mathcal{N} \in L(\mathcal{H}_A \otimes \mathcal{H}_B)$ defined as

$$J^\mathcal{N} = \sum_{i,j=0}^{d_A-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|), \quad (\text{B1})$$

where d_A is the dimension of the input system, and $\{|i\rangle\}_{i=0}^{d_A-1}$ is a fixed orthonormal basis for $\mathcal{H}_A \simeq \mathbb{C}^{d_A}$.

Eq. (B1) can be rewritten in a more compact way by introducing the following notations. For a quantum system A' with the same dimension as A , we denote by $\mathcal{H}_A \otimes \mathcal{H}_{A'}$ by $|\Phi^+\rangle_{AA'} := \sum_i |i\rangle_A \otimes |i\rangle_{A'} / \sqrt{d_A}$ the canonical maximally entangled state of AA' , and we will use the notation $\Phi_{AA'}^+ := |\Phi^+\rangle\langle\Phi^+|_{AA'}$ for the corresponding projector. We will also use subscripts to indicate the systems on which map act, e.g. writing $\mathcal{N}_{A \rightarrow B}$ to indicate that A and B are the input and output systems of the map $\mathcal{N}$, respectively. With this notation, the Choi matrix can be written as $J_{AB}^\mathcal{N} = d_A \mathcal{N}_{A' \rightarrow B}(\Phi_{AA'}^+)$, where we omitted the identity map $\mathcal{I}_{A \rightarrow A}$.

We are now ready to reformulate Eq. (15) in terms of the Choi operator. First, note that the error mitigation condition $\mathcal{N}^{-1} = \sum_i c_i \mathcal{C}_i$ is equivalent to $\mathcal{I} = \mathcal{N}^{-1} \circ \mathcal{N} = \sum_i c_i \mathcal{C}_i \circ \mathcal{N}$, which can be written as $\sum_i J_{\mathcal{C}_i \circ \mathcal{N}} = J_\mathcal{I}$. By Eq. (B1), this condition is equivalent to

$$\sum_i c_i \mathcal{C}_{i,B \rightarrow A''} \circ \mathcal{N}_{A' \rightarrow B}(\Phi_{AA'}^+) = \Phi_{AA''}^+, \quad (\text{B2})$$

where A'' is another quantum system with d_A -dimensional Hilbert space.

In general, each quantum channel $\mathcal{C}_i : L(\mathcal{H}_B) \rightarrow L(\mathcal{H}_A)$ can be realized by performing a unitary channel $\mathcal{U}_i : L(\mathcal{H}_B \otimes \mathcal{H}_R) \rightarrow L(\mathcal{H}_A \otimes \mathcal{H}_{R'})$ on a larger system including an auxiliary input system R and an auxiliary output system R' , of dimensions d_R and $d_{R'}$ satisfying the conditions $d_B d_R = d_A d_{R'}$ and $d_{R'} \leq d_A d_B$. In this unitary realization, the action of the channel $\mathcal{C}_i$ on a generic input state ρ is decomposed as

$$\mathcal{C}_i(\rho) = \text{Tr}_{R'}[\mathcal{U}_i(\rho \otimes |0\rangle\langle 0|)], \quad (\text{B3})$$

where $|0\rangle \in \mathcal{H}_R$ is a fixed initial state of system R .

Adopting the unitary realization (B3), the error mitigation condition (B2) becomes

$$\sum_i c_i \text{Tr}_{R'} [\mathcal{U}_{i, BR \rightarrow A'' R'} (\mathcal{N}_{A' \rightarrow B} (\Phi_{AA'}^+) \otimes |0\rangle\langle 0|_R)] = \Phi_{AA''}^+. \quad (\text{B4})$$

We now introduce the channel $\mathcal{D}_i : L(\mathcal{H}_{A'} \otimes \mathcal{H}_R) \rightarrow L(\mathcal{H}_{A''})$, defined as

$$\mathcal{D}_{i, A' R \rightarrow S''} := \text{Tr}_{R'} \circ \mathcal{U}_{i, BR \rightarrow A'' R'} \circ \mathcal{N}_{A' \rightarrow B}. \quad (\text{B5})$$

With this definition, Eq. (B4) becomes

$$\sum_i c_i \mathcal{D}_{i, A' R \rightarrow A''} (\Phi_{AA'}^+ \otimes |0\rangle\langle 0|_R) = \Phi_{AA''}^+, \quad (\text{B6})$$

and the optimization problem in Eq. (15) of the main text becomes

$$\min_{c_i, U_i} \left\{ \sum_i |c_i| \mid \sum_i c_i \mathcal{D}_{i, A' R \rightarrow A''} (\Phi_{AA'}^+ \otimes |0\rangle\langle 0|_R) = \Phi_{AA''}^+, c_i \in \mathbb{R}, U_i \in \text{Unitary} \right\}, \quad (\text{B7})$$

where Unitary denotes the set of all unitary CPTP maps with input $L(\mathcal{H}_B \otimes \mathcal{H}_{R'})$ and output $L(\mathcal{H}_{A''} \otimes \mathcal{H}_{R'})$. This concludes the proof of Eq. (16) in the main text.

Appendix C: Algorithms for variational quantum error mitigation

Algorithm 1: Variational quantum error mitigation (VQEM)

Input: Noise channel $\mathcal{N}$, real coefficients $\mathbf{c} = \{c_i\}_{i=1}^N$, penalty strength α , N -parameterized quantum channels $\{\mathcal{C}_i(\theta_i)\}_{i=1}^N$, number of iterations ITR.

Output: optimized sampling overhead γ' , quantum protocol to mitigate the noise $\mathcal{N}$.

- 1: Initialize the parameters θ and the real coefficient c .
 - 2: **for** itr = 1, ..., ITR **do**
 - 3: Prepare N -copies of maximally entangled state Φ_{AB}^+ and apply the quantum noise $\mathcal{N}$ on system B .
 - 4: Apply quantum channels $\mathcal{C}_i(\theta_i)$ in parallel on the quantum systems BR , and obtain states $\{\sigma_i\}_{i=1}^N$.
 - 5: Compute the sampling overhead $\gamma' = \|\mathbf{c}\|_1$.
 - 6: Measure $\text{Tr}[\sigma_i \sigma_j]$, $\forall i, j = 1, \dots, N$.
 - 7: Measure $\text{Tr}[\Phi^+ \sigma_i]$, $\forall i = 1, \dots, N$.
 - 8: Compute the loss function $\mathcal{L}_{\text{VQEM}}^{(1)}(\theta, \mathbf{c})$ as shown in Eq. (19).
 - 9: Update the parameters θ and coefficients $\mathbf{c}$ to minimize the loss function.
 - 10: **end for**
 - 11: Output the optimized sampling overhead γ' and the corresponding quantum mitigation protocol.
-

Algorithm 2: Variational quantum error mitigation (VQEM) with budget

Input: Noise channel $\mathcal{N}$, fixed coefficients $\mathbf{c} = \{c_i\}_{i=1}^N$ with sampling overhead budget γ_b , penalty strength α , N -parameterized quantum channels $\{\mathcal{C}_i(\theta_i)\}_{i=1}^N$, number of iterations ITR.

Output: Quantum protocol to mitigate the noise $\mathcal{N}$.

- 1: Initialize the parameters θ .
 - 2: **for** itr = 1, ..., ITR **do**
 - 3: Prepare N -copies of maximally entangled state Φ_{AB}^+ and apply the quantum noise $\mathcal{N}$ on system B .
 - 4: Apply quantum channels $\mathcal{C}_i(\theta_i)$ in parallel on the quantum systems BR , and obtain states $\{\sigma_i\}_{i=1}^N$.
 - 5: Measure $\text{Tr}[\sigma_i \sigma_j]$, $\forall i, j = 1, \dots, N$.
 - 6: Measure $\text{Tr}[\Phi^+ \sigma_i]$, $\forall i = 1, \dots, N$.
 - 7: Compute the loss function $\mathcal{L}_{\text{VQEM}}^{(2)}(\theta)$ as shown in Eq. (20).
 - 8: Update the parameters θ and coefficients $\mathbf{c}$ to minimize the loss function.
 - 9: **end for**
 - 10: Output the optimized quantum mitigation protocol.
-

Appendix D: SDP for channel inverse $\mathcal{N}^{-1}$

The inverse channel $\mathcal{N}^{-1}$ is generally not a CPTP map but an HPTP map, which cannot be implemented on a quantum device directly. Therefore, we have to decompose the inverse channel into a linear combination of CPTP maps,

$$\mathcal{N}^{-1} = \sum_i c_i \mathcal{C}_i, \quad (\text{D1})$$

where c_i are the real coefficients and $\sum_i c_i = 1$ and $\mathcal{C}_i$ are the CPTP maps. With the decomposition, one can simulate the unphysical map.

According to Ref. [33], any HPTP map can be represented with only two CPTP maps

Lemma 1 [33, 51] *Let $\mathcal{M}$ be an HPTP map, then there exist two real numbers $c_1 \geq 0, c_2 \geq 0$ and two CPTP maps $\mathcal{C}_1, \mathcal{C}_2$ such that*

$$\mathcal{M} = c_1 \mathcal{C}_1 - c_2 \mathcal{C}_2. \quad (\text{D2})$$

The decomposition can be simplified into $\mathcal{N}^{-1} = c_1 \mathcal{C}_1 - c_2 \mathcal{C}_2$.

Note that the decomposition of an unphysical map is not unique, and the optimal one is defined by the one with minimal sampling overhead. Then, the optimal decomposition problem can be written as a semidefinite problem as shown in the following. For a given noisy channel $\mathcal{N}$, the error mitigation protocol with minimum sampling overhead can be calculated by the following SDP

$$\min \quad c_1 + c_2 \quad (\text{D3a})$$

$$\text{subject to} \quad J_{\mathcal{N}^{-1}} = J_{\mathcal{C}_1} - J_{\mathcal{C}_2}, \quad (\text{D3b})$$

$$J_{\mathcal{C}_1} \geq 0, \text{Tr}_{out}[J_{\mathcal{C}_1}] = c_1 I_{in}, \quad (\text{D3c})$$

$$J_{\mathcal{C}_2} \geq 0, \text{Tr}_{out}[J_{\mathcal{C}_2}] = c_2 I_{in}, \quad (\text{D3d})$$

where $J_{\mathcal{N}^{-1}}$ refers to the Choi matrix of the noise inverse channel $\mathcal{N}^{-1}$.

Appendix E: Data encoding

In quantum classifier [77], the input data is classical, which cannot be utilized by quantum devices. Therefore, we have to transfer the classical data into quantum data before classifying the data. Such a process is called data encoding. There are various ways to encode classical data, such as angle encoding [78], basis encoding [78, 79], amplitude encoding [78], and Hamiltonian evolution ansatz encoding [80]. In this work we adopt angle encoding for its simplicity. Specifically, for the classical feature vector $\mathbf{x} = [x_1, \dots, x_i, \dots, x_M]^T$, the encoded state is

$$|\psi\rangle_{\mathbf{x}} = \bigotimes_{i=1}^M (\cos(x_i) |0\rangle + \sin(x_i) |1\rangle). \quad (\text{E1})$$

Such an encoded state can be realized by a fixed depth circuit with local rotation gates.

Appendix F: Algorithm for virtual quantum classifier

Appendix G: Chebyshev approximation

For a given budget $\gamma_b \geq 1$, the coefficients are $c_1 = \frac{1+\gamma_b}{2}$ and $c_2 = \frac{1-\gamma_b}{2}$. The spectrum of the averaged state $\sigma = c_1 \rho_1 + c_2 \rho_2$ is contained in $[-|c_2|, |c_1|]$, and the normalized averaged state is denoted as $\tilde{\sigma} = \frac{\sigma}{|c_1|}$. Our target is to approximate the function

$$f(\tilde{\sigma}) = \frac{|\tilde{\sigma}| - \tilde{\sigma}}{2} \quad (\text{G1})$$

Algorithm 3: Virtual quantum classifier (VQC)

Input: Training data set $D = \{\mathbf{x}_i, y_i\}_{i=1}^M$, sampling overhead budget γ_b , two parameterized quantum circuits (PQCs) $U(\boldsymbol{\theta}^{(1)})$, $U(\boldsymbol{\theta}^{(2)})$, number of iterations ITR.

Output: Optimized virtual quantum classifier $f_{\boldsymbol{\theta}^*}$.

- 1: Encode the classical data $\mathbf{x}_i$ into quantum states $|\psi_i\rangle$ for all i .
- 2: Initialize the parameters $\boldsymbol{\theta}^{(1)}$ and $\boldsymbol{\theta}^{(2)}$.
- 3: **for** itr = 1, ..., ITR **do**
- 4: **for** i = 1, ..., M **do**
- 5: Apply $U(\boldsymbol{\theta}^{(1)})$ on the encoded quantum states $|\psi_i\rangle$.
- 6: Make measurement on the first qubit and obtain $\langle Z \rangle_i^{(1)}$.
- 7: Similarly, apply $U(\boldsymbol{\theta}^{(2)})$ on the state $|\psi_i\rangle$, measure the first qubit and obtain $\langle Z \rangle_i^{(2)}$.
- 8: Calculate the prediction $\tilde{y}_i = \frac{1}{2}(1 + c_1 \langle Z \rangle_i^{(1)} + c_2 \langle Z \rangle_i^{(2)})$, where $c_1 = \frac{1+\gamma_b}{2}$ and $c_2 = \frac{1-\gamma_b}{2}$.
- 9: **end for**
- 10: Compute the loss function $\mathcal{L}_{\text{VQC}}(\boldsymbol{\theta}) = \frac{1}{M} \sum_{i=1}^M |\tilde{y}_i - y_i|^2$.
- 11: Minimize the loss function and update the parameters $\boldsymbol{\theta}$.
- 12: The cost converges to its minimum and the corresponding parameters are $\boldsymbol{\theta}^*$.
- 13: **end for**
- 14: Output the optimized virtual quantum classifier $f_{\boldsymbol{\theta}^*}$.

with Chebyshev approximation. The operator absolute value admits the Chebyshev expansion

$$|\tilde{\sigma}| = \frac{2}{\pi} \mathbb{I} + \sum_{k=1}^{\infty} \frac{4(-1)^{k+1}}{\pi(4k^2 - 1)} T_{2k}(\tilde{\sigma}), \quad (\text{G2})$$

where $T_n(\sigma)$ denotes the Chebyshev polynomial of degree n evaluated at σ via functional calculus.

$$T_0(\tilde{\sigma}) = 1, \quad T_1(\tilde{\sigma}) = \tilde{\sigma}, \quad T_{n+1}(\tilde{\sigma}) = 2\tilde{\sigma}T_n(\tilde{\sigma}) - T_{n-1}(\tilde{\sigma}). \quad (\text{G3})$$

Then we obtain

$$f(\tilde{\sigma}) = \frac{|\tilde{\sigma}| - \tilde{\sigma}}{2} = \frac{1}{\pi} \mathbb{I} - \frac{1}{2} \tilde{\sigma} + \sum_{k=1}^{\infty} \frac{2(-1)^{k+1}}{\pi(4k^2 - 1)} T_{2k}(\tilde{\sigma}) \quad (\text{G4})$$

Define the truncated expansion

$$p_K(\tilde{\sigma}) = \sum_{n=0}^K a_n T_n(\tilde{\sigma}), \quad (\text{G5})$$

with coefficients a_n inherited from the above series. By the spectral theorem,

$$f(\tilde{\sigma}) - p_K(\tilde{\sigma}) = \sum_i (f(\lambda_i) - p_K(\lambda_i)) |i\rangle\langle i| \quad (\text{G6})$$

where $\{\lambda_i\}$ are the eigenvalues of $\tilde{\sigma}$. Using the uniform bound $\|T_n(\tilde{\sigma})\|_{\infty} \leq 1$ valid whenever $\text{spec}(\tilde{\sigma}) \subset [-1, 1]$, the Chebyshev tail satisfies

$$\|f(\tilde{\sigma}) - p_K(\tilde{\sigma})\|_{\infty} \leq \sum_{n>K} |a_n| = \frac{2}{\pi(K+1)}. \quad (\text{G7})$$

Let $r = \text{rank}(\tilde{\sigma})$, Then

$$|\text{Tr}[f(\tilde{\sigma})] - \text{Tr}[p_K(\tilde{\sigma})]| = |\text{Tr}[f(\tilde{\sigma}) - p_K(\tilde{\sigma})]| \quad (\text{G8})$$

$$\leq \sum_{i=1}^r |f(\lambda_i) - p_K(\lambda_i)| \quad (\text{G9})$$

$$\leq r \|f(\tilde{\sigma}) - p_K(\tilde{\sigma})\|_{\infty}. \quad (\text{G10})$$

Combine it with Eq. (G7), we have

$$|\text{Tr}[f(\tilde{\sigma})] - \text{Tr}[p_K(\tilde{\sigma})]| \leq \frac{2r}{\pi(K+1)}. \quad (\text{G11})$$

Note that each Chebyshev Polynomial $T_n(\sigma)$ is a degree- n polynomial in $\tilde{\sigma}$,

$$T_n(\tilde{\sigma}) = \sum_{k=0}^{\lfloor n/2 \rfloor} c_{n,k} \tilde{\sigma}^{n-2k}. \quad (\text{G12})$$

Therefore,

$$\text{Tr}[T_n(\tilde{\sigma})] = \sum_{k=0}^{\lfloor n/2 \rfloor} c_{n,k} \text{Tr}[\tilde{\sigma}^{n-2k}], \quad (\text{G13})$$

the Chebyshev polynomial is a linear combination of high-order moments. According to Ref. [81, 82], such a quantity can be efficiently estimated with sampling complexity $\mathcal{O}(k \log(k)/\varepsilon^2)$, where k is the truncation number, and ε refers to the additive number. Even on noisy quantum devices, the high-order moments can be retrieved with error mitigation techniques [48, 52].

From Eq. (G11), the error is upper bounded by $\mathcal{O}(\frac{r}{K})$. For the average state σ with large rank r , the truncation number K should increase correspondingly to guarantee the precision of approximation. However, in real case, the increase of the truncation number K is mitigated for three reasons: ⁽¹⁾ The upper bound is calculated by the largest error $\|f(\tilde{\sigma}) - p_K(\tilde{\sigma})\|_\infty$ times the rank r , which is very loose. In general case, such a bound cannot be saturated; ⁽²⁾ In optimization, the absolute value of the objective function need not be precise; it suffices for the gradient direction to be correct. ⁽³⁾ Errors arising from high rank are distributed across numerous eigenvalues. As these errors arise from polynomial oscillations, they tend to partially cancel during tracing (summation) and do not deviate entirely in one direction.

Appendix H: Algorithm for virtual variational quantum eigensolver

Algorithm 4: Virtual variational quantum eigensolver (VVQE)

Input: Truncation number K , real coefficients c_1, c_2 , penalty strength α , two parameterized quantum circuits (PQCs) $U(\theta^{(1)})$, $U(\theta^{(2)})$, number of iterations ITR, Hamiltonian H .

Output: The ground-state energy of Hamiltonian H .

- 1: Initialize the quantum system to zero states, and randomly prepare the parameters $\theta^{(1,2)}$.
 - 2: **for** itr = 1, ..., ITR **do**
 - 3: Apply $U(\theta^{(1)})$ to zero state and achieve state ρ_1 .
 - 4: Estimate the expectation value $\text{Tr}[H\rho_1]$.
 - 5: Similarly, apply $U(\theta^{(2)})$ to achieve state ρ_2 , and estimate the expectation value $\text{Tr}[H\rho_2]$.
 - 6: Estimate the truncated Chebyshev approximation p_K .
 - 7: Compute the loss function $\mathcal{L}_{\text{VVQE}}(\theta)$ as shown in Eq. (28).
 - 8: Minimize the loss function and update parameter θ .
 - 9: **end for**
 - 10: When the loss function is minimized, we obtain the optimized parameterized quantum circuits $U(\theta^{*(1)})$, $U(\theta^{*(2)})$.
 - 11: Apply the optimal circuits and the ground-state energy is estimated by $E_0 = c_1 \text{Tr}[H\rho_1] + c_2 \text{Tr}[H\rho_2]$
 - 12: Output the ground-state energy E_0 .
-

Appendix I: Simulating long-range entanglement operations with local operations

It was proposed in Ref. [72, 73] that the non-local channel could be simulated by a quasiprobability-based method. Specifically, for a non-local operation Φ , decompose it into the linear combination of local operations and classical communication (LOCC)

$$\Phi = \sum_i c_i L_i \quad (\text{I1})$$

where c_i are the real coefficients, and $\sum_i c_i = 1$, and $L_i \in LOCC$. With the decomposition, Φ can be simulated in a Monte-Carlo manner by sampling L_i with probability proportional $|c_i|$. More concretely, let us define a probabilistic map $\hat{\Phi}$ such that it becomes $\frac{c_i}{|c_i|} L_i$ with probability $p_i = \frac{|c_i|}{W(\Phi)}$ where $W(\Phi) = \sum_i |c_i|$, which is known as simulation overhead. Then,

$$\mathbb{E}[W(\Phi)\hat{\Phi}] = W(\Phi) \cdot \sum_i \frac{|c_i|}{W(\Phi)} \frac{c_i}{|c_i|} L_i = \Phi, \quad (12)$$

which shows that $W(\Phi)\hat{\Phi}$ becomes equal to Φ when executed for many times. The non-local gate simulation procedure is the same as the simulation of an unphysical map.

For a two-qubit gate, its non-local part can always be written as Ref. [83]

$$U = \exp \left[i \left(\sum_{\alpha=1}^3 \theta_{\alpha} \sigma_{\alpha} \otimes \sigma_{\alpha} \right) \right] = \sum_{\alpha=0}^3 u_{\alpha} \sigma_{\alpha} \otimes \sigma_{\alpha}, \quad (13)$$

where σ_0 is the identity with dimension equal to 2, and σ_1, σ_2 and σ_3 are Pauli x, y and z operators, respectively. θ_{α} is a real parameter, and u_{α} is a coefficient that is determined from $\{\theta_{\alpha}\}$. These lead to the following decomposition of channel

$$U = \sum_{\alpha} |u_{\alpha}|^2 \sigma_{\alpha}^{\otimes 2} + \sum_{\alpha < \alpha'} (u_{\alpha} u_{\alpha'}^* + u_{\alpha'} u_{\alpha}^*) (A_{\alpha\alpha'}^{\otimes 2} - B_{\alpha\alpha'}^{\otimes 2}) + \sum_{\alpha < \alpha'} i(u_{\alpha} u_{\alpha'}^* - u_{\alpha'} u_{\alpha}^*) (A_{\alpha\alpha'} B_{\alpha\alpha'} + B_{\alpha\alpha'} A_{\alpha\alpha'}) \quad (14)$$

where σ_{α} refers to the Pauli channels, and $A_{\alpha\alpha'}$ and $B_{\alpha\alpha'}$ are the channels that

$$A_{\alpha\alpha'}(\rho) = \frac{1}{2}(\sigma_{\alpha} \rho \sigma_{\alpha'} + \sigma_{\alpha'} \rho \sigma_{\alpha}), \quad B_{\alpha\alpha'}(\rho) = \frac{1}{2i}(\sigma_{\alpha} \rho \sigma_{\alpha'} - \sigma_{\alpha'} \rho \sigma_{\alpha}) \quad (15)$$

The overhead to simulate the channel U is given in the following $W(U)$

$$W(U) = 1 + \sum_{\alpha \neq \alpha'} (|u_{\alpha} u_{\alpha'}^* + u_{\alpha'} u_{\alpha}^*| + |u_{\alpha} u_{\alpha'}^* - u_{\alpha'} u_{\alpha}^*|). \quad (16)$$