Stochastic noise can be helpful for variational quantum algorithms

Junyu Liu,[1,2,3,4] Frederik Wilde,[5] Antonio Anna Mele,[5] Xin Jin,[2,6] Liang Jiang,[1,3] and Jens Eisert[5,7,8]

[1]*Pritzker School of Molecular Engineering, The University of Chicago, Chicago, Illinois 60637, USA*
[2]*School of Computing and Information, University of Pittsburgh, Pittsburgh, Pennsylvania 15260, USA*
[3]*Chicago Quantum Exchange, Chicago, Illinois 60637, USA*
[4]*Kadanoff Center for Theoretical Physics, The University of Chicago, Chicago, Illinois 60637, USA*
[5]*Dahlem Center for Complex Quantum Systems, Freie Universität Berlin, Berlin 14195, Germany*
[6]*Electrical Engineering and Computer Science, Technische Universität Berlin, Berlin 10587, Germany*
[7]*Helmholtz-Zentrum Berlin für Materialien und Energie, Berlin 14109, Germany*
[8]*Fraunhofer Heinrich Hertz Institute, Berlin 10587, Germany*

(Received 30 May 2024; accepted 6 May 2025; published 22 May 2025)

Saddle points constitute a crucial challenge for first-order gradient descent algorithms. In notions of classical machine learning, they are avoided, for example, by means of stochastic gradient descent methods. In this work, we provide evidence that the saddle-points problem can be naturally avoided in variational quantum algorithms by exploiting the presence of stochasticity. We prove convergence guarantees and present practical examples in numerical simulations and on quantum hardware. We argue that the natural stochasticity of variational algorithms can be beneficial for avoiding strict saddle points, i.e., those saddle points with at least one negative Hessian eigenvalue. This insight that some levels of shot noise could help is expected to add a new perspective to notions of near-term variational quantum algorithms.

DOI: 10.1103/PhysRevA.111.052441

I. INTRODUCTION

Quantum computing has, for many years, been a hugely inspiring theoretical idea. Already in the 1980s, it was suggested that quantum devices could possibly have superior computational capabilities over computers operating based on classical laws [1,2]. It is a relatively recent development that devices have been devised that may indeed have computational capabilities beyond classical means [3–6]. These devices go substantially beyond what was possible not long ago. And still, they are unavoidably noisy and imperfect, likely for many years to come. The quantum devices that are available today and presumably will be in the near future are often conceived as hybrid quantum devices running variational quantum algorithms [7], where a quantum circuit is addressed by a substantially larger surrounding classical circuit. This classical circuit takes measurements from the quantum device and appropriately varies variational parameters of the quantum device in an update. Large classes of *variational quantum eigensolvers* (VQEs), the *quantum approximate optimisation algorithm* (QAOA), and models for quantum-assisted machine learning are thought to operate along those lines, based on suitable *loss functions* to be minimized [8–14]. In fact, many near-term quantum algorithms in the era of *noisy intermediate-scale quantum* (NISQ) computing [15] belong to the class of variational quantum algorithms. While this is an exciting development, it puts a lot of burden on understanding how reasonable and practical classical control can be conceived.

Generally, when the optimization space is high dimensional, updates of the variational parameters are done via *gradient evaluations* [16–19], while zeroth-order and second-order methods are, in principle, also applicable, but typically only up to a limited number of parameters. This makes a lot of sense, as one may think that going downhill in a variational quantum algorithm is a good idea. That said, the concomitant classical optimization problems are generally not convex optimization problems and the variational landscapes are marred by globally suboptimal local optima and saddle points. This becomes particularly prominent when the the search space dimension is high, which often leads to most stationary points—points where the gradient vanishes—being saddle points [20]. This is, however, precisely the overparametrized regime where one expects variational quantum algorithms to perform well. In fact, it is known that the problems of optimizing variational parameters of quantum circuits are computationally hard in worst-case complexity [21]. While this is not of too much concern in practical considerations (since it is often sufficient to find a "good" local minimum instead of the *global* minimum) and resembles an analogous situation in classical machine learning, it does point to the fact that one should expect a rugged optimization landscape, featuring different local minima as well as saddle points. Although, in the infinite-time limit, the first-order algorithm might eventually avoid saddle points with high probability [22], it is shown that in the practical timescale, saddle points matter significantly in the general settings of first-order optimization algorithms [23]. Such saddle points can indeed be a burden to feasible and practical classical optimization of variational quantum algorithms.

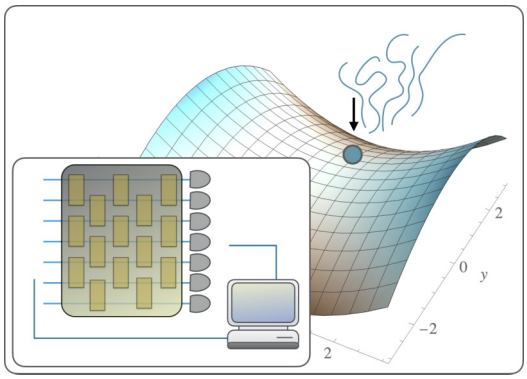

FIG. 1. Stochasticity in variational quantum algorithms can help in avoiding (strict) saddle points.

In this work, we establish the notion that in such situations, *small noise levels* can actually be of substantial help (see Fig. 1). More precisely, we show that some levels of *statistical noise* (specifically, the kind of noise that naturally arises from a finite number of measurements to estimate quantum expectation values) can even be beneficial. We get inspiration from and build on a powerful mathematical theory in *classical machine learning*: there, theorems have been established that say that "noise" can help gradient-descent optimization not get stuck at saddle points [24,25]. Building on such ideas, we show that they can be adapted and developed to be applicable to the variational quantum algorithms setting. Then we argue that the "natural" statistical noise of a quantum experiment can play the role of the artificial noise that is inserted by hand in classical machine learning algorithms to avoid saddle points. We maintain the precise and rigorous mindset of Ref. [24], but show that the findings have practical importance and can be made concrete use of when running variational quantum algorithms on near-term quantum devices. In previous studies, it has been anecdotally observed that small levels of noise can indeed be helpful for improving the optimization procedure [19,26–30]. What is more, variational algorithms have been seen as being noise robust in a sense [31]. That said, while in the past rigorous convergence guarantees have been formulated for convex loss functions of variational quantum algorithms (VQAs) [19,30], in this work we focus on the nonconvex scenario, where saddle points and local minima are present. Such a systematic and rigorous analysis of the type we have conducted that explains the origin of the phenomenon of noise-facilitating optimization has been lacking.

It is important to stress that the noise we refer to in our theorems is the type of noise that adds stochasticity to the gradient estimations, such as the use of a finite number of measurements or the zero-average fluctuations that are involved in real experiments. Also, instances of global depolarizing noise are covered as discussed in Appendix 2. Thus, in this case, noise does not mean the generic quantum noise that results from the interaction with the environment characterized by *completely positive and trace-preserving* (CPTP) maps, which can be substantially detrimental to the performance of the algorithm [32,33]. In addition, it has been shown that noisy CPTP maps in the circuit may significantly worsen the problem of *barren plateaus* [34,35], which is one of the main obstacles to the scalability of *variational quantum algorithms* (VQAs).

We perform numerical experiments, and we show examples where optimizations with gradient descent without noise get stuck at saddle points, whereas if we add some noise, we can escape this problem and get to the minimum—convincingly demonstrating the functioning of the approach. We verify the latter not only in a numerical simulation, but also making use of the data of a real IBM quantum machine.

II. PRELIMINARIES

In our work, we will show how a class of saddle points, the so-called strict saddle points, can be avoided in noisy gradient descent. In developing our machinery, we build strongly on the rigorous results laid out in Ref. [24] and uplift them to the quantum setting at hand. For this, we do method development in its own right. First, we introduce some useful definitions and theorems (see Ref. [24] for a more in-depth discussion).

Throughout this work, we consider the problem of minimizing a function $\mathcal{L} : \mathbb{R}^p \to \mathbb{R}$. We indicate its gradient at θ as $\partial\mathcal{L}(\theta)$ and its Hessian matrix at point θ as $\partial^2\mathcal{L}(\theta)$. We denote as $\|\cdot\|_2$ the l_2 norm of a vector. $\|\cdot\|_{\mathrm{HS}}$ and $\|\cdot\|_\infty$ denote, respectively, the Hilbert-Schmidt norm and the largest eigenvalue norm of a matrix. We denote as $\lambda_{\min}(\cdot)$ as the minimum eigenvalue of a matrix.

Definition 1. L-Lipschitz function. A function $g : \mathbb{R}^p \to \mathbb{R}^d$ is L-Lipschitz if and only if

$$\|g(\theta) - g(\phi)\|_2 \leqslant L\|\theta - \phi\|_2, \tag{1}$$

for every θ and ϕ.

Definition 2. β-strong smoothness. A differentiable function $\mathcal{L} : \mathbb{R}^p \to \mathbb{R}$ is called β-strongly smooth if and only if its gradient is a β-Lipschitz function, i.e.,

$$\|\partial\mathcal{L}(\theta) - \partial\mathcal{L}(\phi)\|_2 \leqslant \beta\|\theta - \phi\|_2, \tag{2}$$

for every θ and ϕ.

Definition 3. Stationary point. If $\mathcal{L}$ is differentiable, θ^* is defined as a stationary point if

$$\|\partial\mathcal{L}(\theta^*)\|_2 = 0. \tag{3}$$

Definition 4. ϵ-approximate stationary point. If $\mathcal{L}$ is differentiable, θ^* is defined as an ϵ-approximate stationary point if

$$\|\partial\mathcal{L}(\theta^*)\|_2 \leqslant \epsilon. \tag{4}$$

Definition 5. Local minimum, local maximum, and saddle point. If $\mathcal{L}$ is differentiable, a stationary point θ^* is a

(i) *local minimum*, if there exists $\delta > 0$ such that $\mathcal{L}(\theta^*) \leqslant \mathcal{L}(\theta)$ for any θ with $\|\theta - \theta^*\|_2 \leqslant \delta$.

(ii) *local maximum*, if there exists $\delta > 0$ such that $\mathcal{L}(\theta^*) \geqslant \mathcal{L}(\theta)$ for any θ with $\|\theta - \theta^*\|_2 \leqslant \delta$.

(iii) *saddle point*, otherwise.

Definition 6. ρ-Lipschitz Hessian. A twice differentiable function $\mathcal{L}$ has ρ-Lipschitz Hessian matrix $\partial^2\mathcal{L}$ if and only if

$$\|\partial^2\mathcal{L}(\theta) - \partial^2\mathcal{L}(\phi)\|_{\mathrm{HS}} \leqslant \rho\|\theta - \phi\|_2, \tag{5}$$

for every θ and ϕ (where $\|\cdot\|_{\mathrm{HS}}$ is the Hilbert-Schmidt norm).

Definition 7. Gradient descent. Given a differentiable function $\mathcal{L}$, the gradient descent algorithm is defined by the update

rule

$$\theta_i^{t+1} = \theta_i^t - \eta \partial_i \mathcal{L}(\theta^t), \tag{6}$$

where $\eta > 0$ is called *learning rate*.

The convergence time of the gradient descent algorithm is given by the following theorem [24].

Theorem 1. Gradient descent complexity. Given a β-strongly smooth function $\mathcal{L}(\cdot)$, for any $\epsilon > 0$, if we set the learning rate as $\eta = 1/\beta$, then the number of iterations required by the gradient descent algorithm such that it will visit an ϵ-approximate stationary point is

$$\mathcal{O}\left(\frac{\beta[\mathcal{L}(\theta_0) - \mathcal{L}^\star]}{\epsilon^2}\right),$$

where θ_0 is the initial point and $\mathcal{L}^\star$ is the value of $\mathcal{L}$ computed in the global minimum.

It is important to note that this result does not depend on the number of free parameters. Also, the stationary point at which the algorithm will converge is not necessarily a local minimum, but can also be a saddle point. Note that a generic saddle point satisfies $\lambda_{\min}[\partial^2 \mathcal{L}(\theta_s)] \leqslant 0$, where $\lambda_{\min}(\cdot)$ is the minimum eigenvalue. Now we define a subclass of saddle points.

Definition 8. Strict saddle point. θ_s is a *strict saddle* point for a twice differentiable function $\mathcal{L}$ if and only if θ_s is a stationary point and if the minimum eigenvalue of the Hessian is $\lambda_{\min}[\partial^2 \mathcal{L}(\theta_s)] < 0$.

Adding the *strict* condition, we remove the case in which a saddle point satisfies $\lambda_{\min}[\partial^2 \mathcal{L}(\theta_s)] = 0$. Moreover, note that a local maximum respects our definition of strict saddle point. Analogously to Ref. [24], in this work, we focus on avoiding strict saddle points. Hence, it is useful to introduce the following definition.

Definition 9. Second-order stationary point. Given a twice differentiable function $\mathcal{L}(\cdot)$, θ^* is a second-order stationary point if and only if

$$\partial \mathcal{L}(\theta^*) = \mathbf{0} \quad \text{and} \quad \lambda_{\min}[\partial^2 \mathcal{L}(\theta^*)] \geqslant 0. \tag{7}$$

Definition 10. ϵ-second-order stationary point. For a ρ-Hessian Lipschitz function $\mathcal{L}(\cdot)$, θ^* is an ϵ-second-order stationary point if

$$\|\partial \mathcal{L}(\theta^*)\|_2 \leqslant \epsilon \quad \text{and} \quad \lambda_{\min}[\partial^2 \mathcal{L}(\theta^*)] \geqslant -\sqrt{\rho\epsilon}. \tag{8}$$

Gradient descent (GD) makes a nonzero step only when the gradient is nonzero, and thus in the nonconvex setting it will be stuck at saddle points. A simple variant of GD is the *perturbed gradient descent* (PGD) method [24], which adds randomness to the iterates at each step.

Definition 11. Perturbed gradient descent. Given a differentiable function $\mathcal{L} : \mathbb{R}^p \to \mathbb{R}$, the perturbed gradient descent algorithm is defined by the update rule,

$$\theta_i^{t+1} = \theta_i^t - \eta[\partial_i \mathcal{L}(\theta^t) + \zeta^t], \tag{9}$$

where $\eta > 0$ is the learning rate and ζ^t is a normally distributed random variable with mean $\mu = 0$ and variance $\sigma^2 = r^2/p$ with $r \in \mathbb{R}$.

In Ref. [24], the authors show that if we pick $r = \tilde{\Theta}(\epsilon)$, PGD will find an ϵ-second-order stationary point in a number of iterations that has only a polylogarithmic dependence on the number of free parameters, i.e., it has the same complexity of (standard) gradient descent up to polylogarithmic dependence.

Theorem 2. [24]. Let the function $\mathcal{L} : \mathbb{R}^d \to \mathbb{R}$ be β-strongly smooth and such that it has a ρ Lipschitz-Hessian. Then, for any $\epsilon, \delta > 0$, the PGD algorithm starting at the point θ_0, with parameters $\eta = \tilde{\Theta}(1/\beta)$ and $r = \tilde{\Theta}(\epsilon)$, will visit an ϵ-second-order stationary point at least once in the following number of iterations, with probability at least $1 - \delta$,

$$\tilde{\mathcal{O}}\left(\frac{\beta[\mathcal{L}(\theta_0) - \mathcal{L}^\star]}{\epsilon^2}\right),$$

where $\tilde{\mathcal{O}}$ and $\tilde{\Theta}$ hide polylogarithmic factors in $p, \beta, \rho, 1/\epsilon, 1/\delta$, and $\Delta_{\mathcal{L}} := \mathcal{L}(\theta_0) - \mathcal{L}^\star$. Here, θ_0 is the initial point and $\mathcal{L}^\star$ is the value of $\mathcal{L}$ computed in the global minimum.

This theorem has been proven in Ref. [24] for Gaussian distributions, but the authors have pointed out that this is not strictly necessary and that it can be generalized to other types of probability distributions in which appropriate concentration inequalities can be applied (for a more in-depth discussion, see Ref. [24]).

In Ref. [36], it has been shown that although the standard GD (without perturbations) almost always escapes the saddle points asymptotically [37], there are (nonpathological) cases in which the optimization requires exponential time to escape. This highlights the importance of using gradient descent with perturbations.

III. STATISTICAL NOISE IN VARIATIONAL QUANTUM ALGORITHMS

Our analysis focuses on variational quantum algorithms in which the loss function to be minimized has the following form:

$$\mathcal{L}(\theta) := \langle 0|U^\dagger(\theta) O U(\theta)|0\rangle, \tag{10}$$

where O is a Hermitian operator and $U(\theta)$ is a parameterized unitary of the form

$$U(\theta) := \prod_{\ell=1}^{p} W_\ell \exp(i\theta_\ell X_\ell), \tag{11}$$

where W_ℓ and X_ℓ are, respectively, fixed unitaries and Hermitian operators. Theorem 2 above assumes that the loss function to minimize is β-strongly smooth and has a ρ-Lipschitz Hessian. To guarantee that these conditions are met for loss functions of parametrized quantum circuits, we provide the following theorem.

Theorem 3. Conditions for loss functions of parametrized quantum circuits. The loss function given in Eq. (10) with θ being a p-dimensional vector is β-strongly smooth and has a ρ-Lipschitz Hessian. In particular, we have

$$\beta \leqslant 2^2 p \|O\|_\infty \max_{j=1,\ldots,p} \|X_j\|_\infty^2, \tag{12}$$

$$\rho \leqslant 2^3 p^{\frac{3}{2}} \|O\|_\infty \max_{j=1,\ldots,p} \|X_j\|_\infty^3. \tag{13}$$

We provide a detailed proof in Appendix 1. It is important to observe that for typical VQAs, the observable O associated to the loss function and the components X_i have an operator

norm that grows, at most, polynomially with the number of qubits, so β and ρ will also grow, at most, polynomially. This is because the circuit depth p must be chosen to be, at most, $\mathcal{O}[\text{poly}(n)]$ for n qubits, X_l as well as O are usually chosen to be Pauli strings in which case their operator norms are 1 or, to be linear combinations of $\mathcal{O}[\text{poly}(n)]$, many Pauli strings with $\mathcal{O}[\text{poly}(n)]$ coefficients (as in QAOA), therefore, by the triangle inequality, their operator norm is bounded by $\mathcal{O}[\text{poly}(n)]$. Sometimes, O is also chosen to be a quantum state [7], therefore with the operator norm bounded by 1. Hence, the number of iterations in Theorem 2 does not grow exponentially in the number of qubits.

The previous results can be easily generalized for the case of differentiable and bounded loss functions, which are functions of expectation values, i.e.,

$$\mathcal{L}(\theta) = f(\langle 0|U^\dagger(\theta)OU(\theta)|0\rangle). \tag{14}$$

In fact, we observe that if $\mathcal{L}$ and g are Lipschitz functions, then

$$\begin{aligned} |\mathcal{L}[g(\theta)] - \mathcal{L}[g(\theta')]| &\leqslant L_{\mathcal{L}}\|g(\theta) - g(\theta')\|_2 \\ &\leqslant L_{\mathcal{L}}L_g\|\theta - \theta'\|_2. \end{aligned} \tag{15}$$

In addition, if $\mathcal{L}$ is a differentiable function with bounded derivatives on a convex set, then (because of the mean value theorem) $\mathcal{L}$ is Lipschitz on this set. From this follows that if $\mathcal{L}$ is a differentiable function with bounded derivatives of a quantum expectation value (whose image defines a bounded $\mathbb{R}$ interval), then it is Lipschitz. Moreover, the sum of Lipschitz functions is a Lipschitz function. Therefore, functions of expectation values commonly used in machine learning tasks, such as the *mean-squared error*, satisfy the Lipschitz condition.

Moreover, Theorem 2 assumes that at each step of the gradient descent, a normally distributed random variable is added to the gradient, namely, $\theta_i^{t+1} = \theta_i^t - \eta[\partial_i\mathcal{L}(\theta^t) + \zeta^t]$. In VQAs, the partial derivatives are commonly estimated using a finite number of measurements, such as by the *parameter shift rule* [16]. Here, the update rule for the gradient descent is

$$\theta_i^{t+1} = \theta_i^t - \eta\hat{g}_i(\theta^t), \tag{16}$$

where $\hat{g}_i(\theta^t)$ is an estimator of the partial derivative $\partial_i\mathcal{L}(\theta^t)$ obtained by a finite number of measurements, N_{shots}, from the quantum device. Moreover, we define

$$\hat{\zeta}^t_{N_{\text{shots}}} := \partial_i\mathcal{L}(\theta^t) - \hat{g}_i(\theta^t). \tag{17}$$

Note that $\hat{\zeta}^t_{N_{\text{shots}}}$ is a random variable with zero expectation value. Therefore, we have

$$\theta_i^{t+1} = \theta_i^t - \eta\big[\partial_i\mathcal{L}(\theta^t) + \hat{\zeta}^t_{N_{\text{shots}}}\big]. \tag{18}$$

The "noise" $\hat{\zeta}^t_{N_{\text{shots}}}$ will play the role of the noise that is added by hand in the perturbed-gradient descent of the algorithm given in Definition 11. However, we cannot exactly control the distribution of such random variable, nor the variance. However, it is to be expected that in the limit of many measurement shots, by the central limit theorem, the noise encountered in practice will be close to the noise considered here, i.e., a Gaussian distribution.

IV. NUMERICAL AND QUANTUM EXPERIMENTS

In this section, we discuss the results of numerical and quantum experiments we have performed to show that stochasticity can help escape saddle points. Our results suggest that statistical noise leads to a nonvanishing probability of not getting stuck in a saddle point and thereby reaching a lower value of the loss function. These numerical experiments also complement the rigorous results that are proven to be valid under very precisely defined conditions, while the intuition developed here is expected to be more broadly applicable, so that the rigorous results can be seen as proxies for a more general mindset. We have also observed this phenomena in a real IBM quantum device. We have done so to convincingly stress the significance of our results in practice.

Let us first consider the Hamiltonian $O = \sum_{i=1}^{N=4} Z_i$. The loss function we consider is defined as the expectation value of such a Hamiltonian over the parametrized quantum circuit qml.StronglyEntanglingLayers implemented in PENNYLANE [17], where two layers of the circuit are used.

In all our experiments, we first initialize the parameters in multiple randomly chosen values. Next, we select the initial points for which the optimization process gets stuck at a suboptimal loss-function value, thereby focusing on cases in which saddle points constitute a significant problem for the (noiseless) optimizer. This selection can be justified by the fact that the loss function defined by O is trivial to begin with. The relevant aspect of this experiment is to study situations in which the optimizer encounters saddle points. As such, we exclusively investigate these specifically selected initial points by subsequently initializing the noisy optimizer with them.

As a proof of principle, we first show the results of an exact simulation (i.e., the expectation values are not estimated using a finite number of shots, but are calculated exactly) in which noise is added manually at each step of the gradient descent. The probability distribution associated to the noise is chosen to be a Gaussian distribution with mean $\mu = 0$ and variance $\sigma^2 = r^2$. Figure 2 shows the difference between the noiseless and noisy calculations with the same initial conditions of the gradient descent, when the noise is from random Gaussian perturbations that are added manually. Figure 3 shows the performance of the experiment, defined as $1/(\mathcal{L} - \mathcal{L}_{\text{opt}})$ as a function of the noise parameter. Here, we can find a critical value of noise, leading to saddle-point avoidance. Figure 4 specifically addresses quantum noise levels, with simulated results about purely statistical noise levels (shot noise) and device noise (simulated by making use of the noise model of actual quantum hardware IBM QISKIT).

It should be noted that including device noise generally also means dealing with completely positive trace-preserving maps that can lead to a different loss function, with new local minima, new saddle points, and a flatter landscape [34]. However, even in this case, we observe an improvement in performance using the same initial parameters leading to the saddle point in the noiseless case. This is perfectly in line with the intuition developed here, as long as the effective emerging noise can be seen as a small perturbation of the reference circuit featuring a given loss landscape that is then in effect perturbed by stochastic noise.

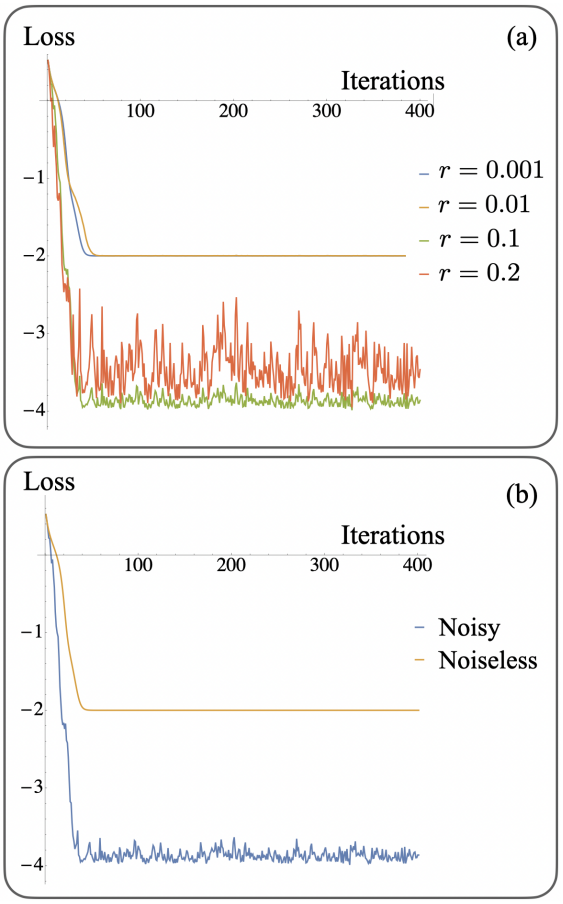

FIG. 2. Comparison of the loss evolution with or without noise. The noise levels are manually added Gaussian distributions, and we keep the same initial conditions. (a) Four different values of the standard deviation r. (b) Noiseless case and the noisy case with the standard deviation of the noise $r = 0.1$.

Aside from the quantum machine learning example, we also provide another instance in *variational quantum eigensolvers* (VQEs). Here, we use the Hamiltonian associated to the hydrogen molecule H_2, which is a four-qubit Hamiltonian obtained by the fermionic one performing a Jordan-Wigner transformation. We specifically use the same circuit from h2.xyz, the Hydrogen VQE example in PENNYLANE [18]. Also

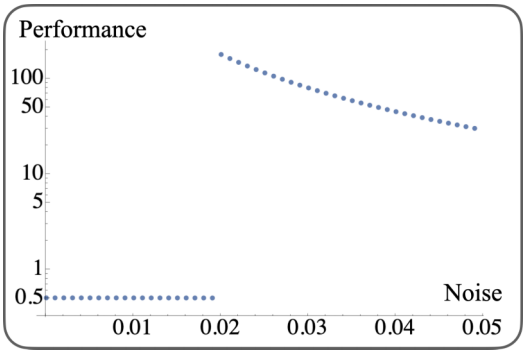

FIG. 3. We quantify the performance against the size of the noise r (classical Gaussian noise) by $1/(\mathcal{L} - \mathcal{L}_{\text{opt}})$.

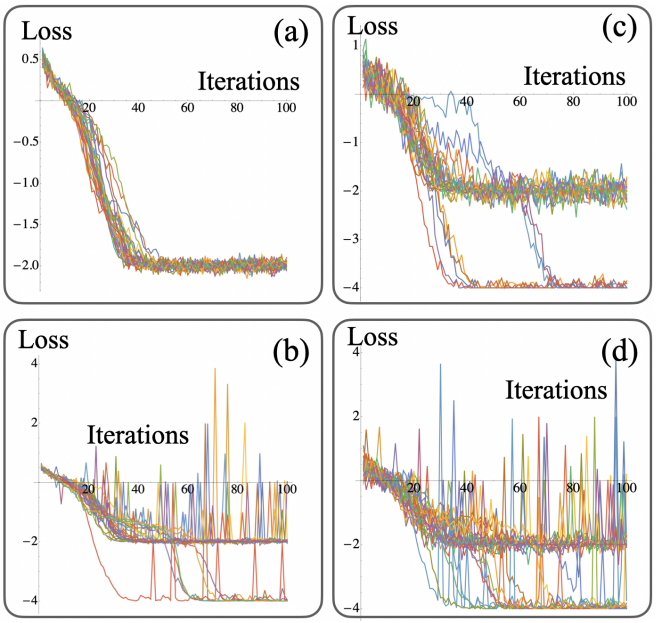

FIG. 4. Saddle-point avoidance from quantum noise. We prepare 30 instances starting from the same initial condition. When (a) noise levels are small, with (b) purely measurement noise, including device noise, and shot number is 1000, most trajectories cannot jump out of the saddle points. When (c) noise levels are larger, with (d) purely measurement noise, including device noise, and shot number is 70, we have a probability to jump towards the global minimum.

here, given the initial parameters that led to saddle points in the noiseless case, we find that starting by the same parameters and adding noise can lead to saddle-point avoidance. Results are shown in Fig. 5 where we compare the noiseless and noisy simulation.

To further provide evidence of the functioning of our suggested approach and the rigorous established insights, we put the findings into contact with the results of a real experiment example in the IBM QISKIT environment. We use the Hamiltonian $O = \sum_{i=1}^{N=4} Z_i$ that we used in our first numerical

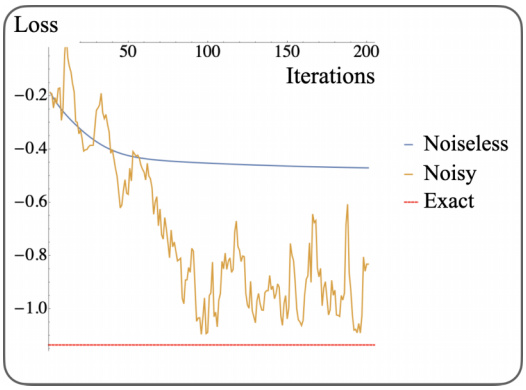

FIG. 5. Comparison of the loss evolution with or without noise with Hydrogen VQE. The noise is manually drawn from Gaussian distributions with the standard deviation 0.2, and we keep the same initial conditions. We compare the noiseless case, noisy case, and the exact solution.

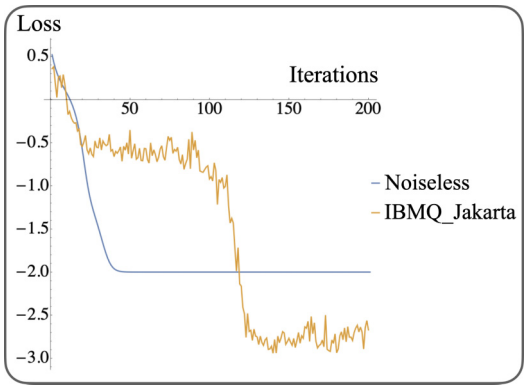

FIG. 6. A real quantum experiment. We use the IBMQ Jakarta device with 10 000 shots.

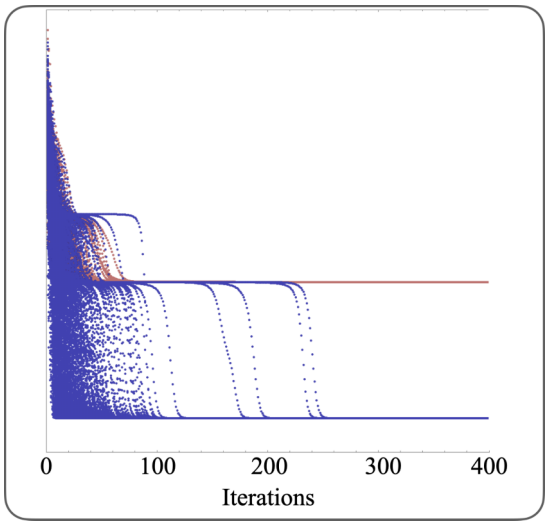

FIG. 7. More initial points under the same gradient-descent dynamics. We have studied the same gradient descent dynamics as in Fig. 3 with four qubits, where 1000 initial points have been randomly sampled. We have found that in 305 cases (labeled in red), the gradient descent gets trapped by saddle points.

simulation with four qubits and two layers. We run the experiment using as the initial condition the one that leads to a saddle point in the noiseless case. We use the IBMQ Jakarta device with 10 000 shots. The result in Fig. 6 shows that it is possible to obtain a lower value of the cost function than that of the simulation without noise that has been stuck in a saddle point.

One may also ask at this point what the "sweet spot" of the appropriate stochastic noise might possibly be. It is known that for most local quantum circuits being subject to constant noise levels under general local qubit noise, one can expect the maximum attainable circuit depth to scale like $\mathcal{O}[\ln(n)]$ [38]. Also, it is known that Pauli expectation values between two different input states are exponentially suppressed in the circuit depth d [38]. This implies that the logarithm of the noise levels must be chosen as $\mathcal{O}[1/\mathrm{poly}(d)]$ to be nondetrimental for read-out, but still make sure that one avoids saddle points in variational optimization. Too high noise levels, in the form of contractive device noise, will eventually lead to noise-induced barren plateaus [34] and an eventual disappearance of any distinguishability of outputs.

To investigate the significance of saddle points in the case of variational quantum algorithms, we examine the generality of initial points that could be trapped by saddle points. In Fig. 7, we have studied 1000 initial variational angles randomly sampled between $[0, 2\pi)$ in the same setup of Fig. 3 with four qubits. Under identical gradient-descent conditions, we have observed that 305 initial points lead to saddle points rather than local minima in the absence of noise. Consequently, the estimated probability of getting stuck near saddle points in our study is approximately 30.5%. Classical nonconvex optimization theory demonstrates that saddle points are not anomalies but rather common features in loss function landscapes, making stochastic gradient descents crucial for most traditional machine learning applications. While a 30.5% failure rate in optimization is manageable by simply repeating the algorithm multiple times to recover the global minimum with high probability, we anticipate that in quantum machine learning, getting trapped by saddle points will also be a general occurrence. Moreover, the likelihood of encountering saddle points is expected to grow significantly in higher-dimensional parameter spaces [20]. Here is a straightforward explanation: Saddle points are determined by the signs of Hessian eigenvalues. In models with p parameters, there are p Hessian eigenvalues. Assuming equal chances of positive and negative eigenvalues, the probability of obtaining a positive semidefinite Hessian becomes exponentially small (2^{-p}). Therefore, as the model size increases, saddle points become increasingly prevalent.

V. CONCLUSION AND OUTLOOK

In this work, we have proposed small stochastic noise levels as an instrument to facilitate variational quantum algorithms. This noise can be substantial, but should not be too large: The way a noise level can strike the balance in overcoming getting stuck in saddle points and being detrimental is in some ways reminiscent of the phenomenon of *stochastic resonance* in statistical physics [39]. This is a phenomenon in which suitable small increases in levels of noise can increase in a metric of the quality of the signal transmission, resonance, or detection performance, rather than a decrease. Here, also, fine-tuned noise levels can facilitate resonance behavior and avoid getting trapped.

It is worth stressing that our results focus specifically on shot noise, which can help in overcoming saddle points. This is fundamentally distinct from contractive noise maps, such as depolarizing noise, which may induce barren plateaus [34] in variational quantum algorithms and does not help in avoiding saddle points. It has been shown that shot noise leads to barren plateaus *only* in the case of global observables [34], which is not the setting considered in this work. At the end of the day, one should expect specifics of the noise map, as for overly large noise levels of a certain type, the performance of variational approaches will also worsen [40].

We also emphasize that our analysis is focused on intrinsic quantum noise in quantum computing (shot noise), not merely the setting where one adds extra classical Gaussian noise to the gradient. While we use Gaussian noise to approximate

the shot noise under certain conditions (when the number of measurement shots is large), our primary objective is to show that quantum noise—an unavoidable feature of quantum systems—can enhance optimization within limited noise level ranges. We do not claim that noise is universally beneficial nor advocate for intentionally adding extra classical noise. Instead, we emphasize the importance of identifying an optimal level of inherent quantum noise that balances performance and practicality. This perspective suggests a more tolerant approach to quantum noise in gradients, reducing the reliance on perfectly noiseless quantum systems while mitigating saddle-point entrapment.

On a higher level, our work invites one to think more deeply about the use of classical stochasic noise in variational quantum algorithms as well as ways to prove performance guarantees about such approaches. For example, *Metropolis sampling*–inspired classical algorithms in which a stochastic process satisfying detailed balance is set up over variational quantum circuits may assist in avoiding getting stuck in rugged energy landscapes.

It is also interesting to note that the technical results obtained here provide further insights into an alternative interpretation of the setting discussed here. Instead of regarding the noise as stochastic noise that facilitates the optimization in the proven fashion established here, one may argue that the noise channels associated with the noise alter the variational landscapes in the first place [41,42]. For example, such quantum channels are known to be able to break parameter symmetries in over-parameterized variational algorithms. It is plausible to assume that these altered variational landscapes may be easier to optimize over. It is an interesting observation in its own right that the technical results obtained here also have implications to this alternative viewpoint, as the convergence guarantees are independent of the interpretation. It is the hope that the present work puts the role of stochasticity in variational quantum computing into a new perspective, and contributes to a line of thought exploring the use of suitable noise and sampling for enhancing quantum computing schemes.

ACKNOWLEDGMENTS

J.L. is supported in part by the International Business Machines (IBM) Quantum through the Chicago Quantum Exchange, and the Pritzker School of Molecular Engineering at the University of Chicago through AFOSR MURI (Grant No. FA9550-21-1-0209). J.L. and X.J. are supported in part by the University of Pittsburgh, School of Computing and Information, Department of Computer Science, Pitt Cyber, PQI Community Collaboration Awards and NASA under Award No. 80NSSC25M7057. F.W., A.A.M., and J.E. thank the ERC (DebuQC), the BMBF (Hybrid, MuniQC-Atoms, DAQC, Hybrid++, QuSol), the BMWK (EniQmA, PlanQK), the MATH+ Cluster of Excellence, the Quantum Flagship (Millenion, PasQuans2), the Einstein Foundation (Einstein Unit on Quantum Devices), Berlin Quantum, the QuantERA (HQCC), the Munich Quantum Valley (K8), the DFG (CRC 183), and the European Research Council (DebuQC) for support. L.J. acknowledges support from the ARO (Grants No. W911NF-18-1-0020 and No. W911NF-18-1-0212), ARO MURI (Grant No. W911NF-16-1-0349), AFOSR MURI (Grants No. FA9550-19-1-0399 and No. FA9550-21-1-0209), DoE Q-NEXT, NSF (Grants No. EFMA-1640959, No. OMA-1936118, and No. EEC-1941583), NTT Research, and the Packard Foundation (Grant No. 2013-39273). This research used resources of the Oak Ridge Leadership Computing Facility, which is a U. S. Department of Energy Office of Science User Facility supported under Contract No. DE-AC05-00OR22725.

J.E. and J.L. suggested the exploitation of classical stochastic noise in variational quantum algorithms, and to prove convergence guarantees for the performance of the resulting algorithms. J.L., A.A.M., F.W., and J.E. proved the theorems of convergence. J.L. and F.W. devised and conducted the numerical simulations. J.L. performed the quantum device experiments under the guidance of L.J. X.J. attended the discussions and contributed in the scientific updates of the draft. All authors discussed the results and wrote the manuscript.

DATA AVAILABILITY

The data and code used for the experiments are available at [43].

APPENDIX

1. Strong smoothness and Lipschitz-Hessian property

In this Appendix, we provide a proof of Theorem 3 of the main text. As stated in the main text, we focus our analysis on ansatz circuits of the form

$$U(\theta) := \prod_{\ell=1}^{p} W_\ell \exp(i\theta_\ell X_\ell), \tag{A1}$$

where W_ℓ and X_ℓ are, respectively, fixed unitaries and Hermitian operators. As a reminder, we want to show that the loss function

$$\mathcal{L}(\theta) = \langle 0|U^\dagger(\theta) O U(\theta)|0\rangle \tag{A2}$$

is β-strongly smooth and has a ρ-Lipschitz Hessian, with

$$\beta \leqslant 2^2 p \|O\|_\infty \max_{j=1,\ldots,p} \|X_j\|_\infty^2, \tag{A3}$$

$$\rho \leqslant 2^3 p^{\frac{3}{2}} \|O\|_\infty \max_{j=1,\ldots,p} \|X_j\|_\infty^3. \tag{A4}$$

To begin, we state three important facts about the Lipschitz constants of multivariate functions.

Lemma A 1. If $\mathcal{L} : \mathbb{R}^p \to \mathbb{R}$ is differentiable with bounded partial derivatives, then

$$L = \sqrt{p} \max_j \left(\sup_\theta \left| \frac{\partial \mathcal{L}(\theta)}{\partial \theta_j} \right| \right) \tag{A5}$$

is the Lipschitz constant for $\mathcal{L}$.

The proof is given in Ref. [44] (Lemma 7).

Lemma A 2. If $\mathcal{L} : \mathbb{R}^p \to \mathbb{R}^M$ is a function with all its M-component Lipschitz functions with Lipschitz constant L_i, then $\mathcal{L}$ has Lipschitz constant $L = \sqrt{\sum_{i=1}^M L_i^2}$.

The proof is given in Ref. [44] (Lemma 8).

Equipped with these facts, we can proceed to derive an upper bound for the Lipschitz constant of functions from $\mathbb{R}^p$ to $\mathbb{R}^M$.

Lemma A 3. If $g:\mathbb{R}^p \to \mathbb{R}^M$ is a differentiable function with bounded gradient, then its Lipschitz constant L satisfies

$$L \leqslant \sqrt{pM}\max_{i,j}\left(\sup_{\theta}\left|\frac{\partial g_i(\theta)}{\partial\theta_j}\right|\right), \tag{A6}$$

where $g_i(\theta)$ the ith component of $\theta \mapsto g(\theta)$.

Proof. Using Lemmas A 1 and A 2, we have

$$\begin{aligned} L &= \left(\sum_{i=1}^{M} L_i^2\right)^{1/2} \leqslant \sqrt{M}\max_i(L_i) \\ &= \sqrt{pM}\max_{i,j}\left(\sup_{\theta}\left|\frac{\partial g_i(\theta)}{\partial\theta_j}\right|\right), \end{aligned} \tag{A7}$$

where L_i is the Lipschitz constant of the ith component of $\mathcal{L}$ as defined in Lemma A 2. ∎

Next, we focus on loss functions of the type in Eq. (A2).

Lemma A 4. The loss function as defined in Eq. (A2) (with $\theta \in \mathbb{R}^p$) satisfies

$$\max_{i_1,i_2\ldots i_k}\left(\sup_{\theta}\left|\frac{\partial^k\mathcal{L}(\theta)}{\partial\theta_{i_k}\cdots\partial\theta_{i_2}\partial\theta_{i_1}}\right|\right) \leqslant 2^k\|O\|_\infty \max_{j=1,\ldots,p}\|X_j\|_\infty^k, \tag{A8}$$

where $U(\theta)$ is given by Eq. (A1).

Proof. We introduce the standard *multi-index* notation. For this, we have

$$\partial^\alpha\mathcal{L}(\theta) := \frac{\partial^k\mathcal{L}(\theta)}{\partial\theta_{i_k}\cdots\partial\theta_{i_2}\partial\theta_{i_1}}. \tag{A9}$$

With this, the multiderivative of the loss reads

$$\begin{aligned} \partial^\alpha\mathcal{L}(\theta) &= \langle 0|\partial^\alpha[U^\dagger(\theta)OU(\theta)]|0\rangle \\ &= \sum_{\beta:\beta\leqslant\alpha}\binom{\alpha}{\beta}\langle 0|[\partial^\beta U^\dagger(\theta)]O[\partial^{\alpha-\beta}U(\theta)]|0\rangle, \end{aligned} \tag{A10}$$

where we have exploited the generalized Leibniz formula,

$$\partial^\alpha(fg) = \sum_{\beta:\beta\leqslant\alpha}\binom{\alpha}{\beta}(\partial^\beta f)\partial^{\alpha-\beta}g. \tag{A11}$$

We have

$$\begin{aligned} |\partial^\alpha\mathcal{L}| &\leqslant \sum_{\beta:\beta\leqslant\alpha}\binom{\alpha}{\beta}|\langle 0|[\partial^\beta U^\dagger(\theta)]O[\partial^{\alpha-\beta}U(\theta)]|0\rangle| \\ &= 2^{|\alpha|}\max_{\gamma:\gamma\leqslant\alpha}|\langle 0|[\partial^\gamma U^\dagger(\theta)]O\,\partial^{\alpha-\gamma}U(\theta)|0\rangle| \\ &\leqslant 2^{|\alpha|}\max_{\gamma:\gamma\leqslant\alpha}\|[\partial^\gamma U^\dagger(\theta)]O\,\partial^{\alpha-\gamma}U(\theta)\|_\infty \\ &\leqslant 2^{|\alpha|}\max_{\gamma:\gamma\leqslant\alpha}\|\partial^\gamma U^\dagger(\theta)\|_\infty\|O\|_\infty\|\partial^{\alpha-\gamma}U(\theta)\|_\infty, \end{aligned} \tag{A12}$$

where we have used the triangle inequality and the multibinomial theorem formula to write

$$\sum_{\beta:\beta\leqslant\alpha}\binom{\alpha}{\beta} = 2^{|\alpha|}, \tag{A13}$$

the fact that

$$|\langle 0|A|0\rangle| \leqslant \|A|0\rangle\|_2 \leqslant \|A\|_\infty, \tag{A14}$$

which follows immediately by Cauchy-Schwarz, and the subadditivity of the $\|\cdot\|_\infty$ norm. Using the form of the parameterized unitary in Eq. (A1), we can also observe that

$$\begin{aligned} \|\partial^\gamma U^\dagger(\theta)\|_\infty &= \left\|\frac{\partial^{\gamma_p}}{\partial\theta_p^{\gamma_p}}\cdots\frac{\partial^{\gamma_1}}{\partial\theta_1^{\gamma_1}}U^\dagger(\theta)\right\|_\infty \\ &\leqslant \|X_1\|_\infty^{\gamma_1}\cdots\|X_p\|_\infty^{\gamma_p} \\ &\leqslant \left(\max_{j=1,\ldots,p}\|X_j\|_\infty\right)^{\gamma_1+\cdots+\gamma_p} \\ &\leqslant \max_{j=1,\ldots,p}\|X_j\|_\infty^{|\gamma|}, \end{aligned} \tag{A15}$$

where we have used that the subadditivity of the infinity norm and the fact that the spectral norm of a unitary matrix is given by the unity. Similarly, we have

$$\|\partial^{\alpha-\gamma}U(\theta)\|_\infty \leqslant \max_{j=1,\ldots,p}\|X_j\|_\infty^{|\alpha|-|\gamma|}. \tag{A16}$$

Therefore, combining the previous two inequalities with Eq. (A12), we have

$$\begin{aligned} |\partial^\alpha\mathcal{L}| &\leqslant 2^{|\alpha|}\|O\|_\infty\max_{j=1,\ldots,p}\|X_j\|_\infty^{|\alpha|} \\ &= 2^k\|O\|_\infty\max_{j=1,\ldots,p}\|X_j\|_\infty^k, \end{aligned} \tag{A17}$$

where we have used $|\alpha| = k$. ∎

We are now ready to provide the proof of Theorem 3 of the main text. Since the loss function is a combination of sine and cosine functions, its derivatives exist and are bounded, and from this it follows that the loss function is strongly smooth and its Hessian is Lipschitz. However, it is worth explicitly calculating β and ρ and bounding them to verify, for example, the scaling with the number of qubits.

Proof. We have the β-smooth constant defined by the smallest β with

$$\|\partial\mathcal{L}(\theta) - \partial\mathcal{L}(\theta')\| \leqslant \beta\|\theta - \theta'\|, \tag{A18}$$

which means we need to consider the Lipschitz constant for the p-dimensional function $\partial\mathcal{L}(\theta)$. Using Lemma A 3, where $g(\theta) = \partial\mathcal{L}$ and $M = p$, we have

$$\beta \leqslant p\max_{i,j}\left(\sup_{\theta}\left|\frac{\partial^2\mathcal{L}(\theta)}{\partial\theta_i\partial\theta_j}\right|\right). \tag{A19}$$

Applying Lemma A 4, we find

$$\beta \leqslant 2^2 p\|O\|_\infty\max_i(\|X_i\|_\infty)^2, \tag{A20}$$

where we have used the matrix spectral (operator) norm.

The ρ-Hessian constant is defined as

$$\|\partial^2\mathcal{L}(\theta) - \partial^2\mathcal{L}(\theta')\|_{\mathrm{HS}} \leqslant \rho\|\theta - \theta'\|_2, \tag{A21}$$

where $\partial^2\mathcal{L}$ is the Hessian matrix and we have used the Hilbert-Schmidt matrix norm. Note that the Hilbert-Schmidt norm of a matrix is the 2-norm of the matrix *vectorization* $\mathrm{vec}(\cdot)$. We can now apply Lemma A 3, where $M = p^2$ since the Hessian

is a map from $\mathbb{R}^p$ to $\mathbb{R}^{p\times p}$. Defining $g(\theta) = \mathrm{vec}[\partial^2\mathcal{L}(\theta)]$, we find

$$\rho \leqslant p^{\frac{3}{2}} \max_{i,j,k}\left(\sup_{\theta}\left|\frac{\partial^3\mathcal{L}(\theta)}{\partial\theta_k\partial\theta_i\partial\theta_j}\right|\right). \tag{A22}$$

Thus, applying Lemma A 4, we arrive at

$$\rho \leqslant 2^3 p^{\frac{3}{2}} \|O\|_\infty \max_i(\|X_i\|_\infty)^3. \tag{A23}$$

■

2. Discussion on more general noise

In this Appendix, we discuss the impact of more general noise. We assume that we have device noise that is constant in time, i.e., for each state preparation, effectively, we always encounter the same CPTP maps acting on the initial state. This will change the state $\rho(\theta)$ at the end of the circuit to a noisy instance $\rho_{\text{noisy}}(\theta)$ which can be modeled as the applications of parametrized unitary layers interspersed with suitable CPTP maps to the initial state ρ_0 as

$$\rho_{\text{noisy}}(\theta) = \mathcal{N}_p \circ \mathcal{U}_p \circ \cdots \circ \mathcal{N}_1 \circ \mathcal{U}_1(\rho_0), \tag{A24}$$

where $\mathcal{N}_i$ and $\mathcal{U}_i$ are, respectively, noisy CPTP maps and the parametrized unitary channels. As a result, the loss function changes to

$$\mathcal{L}_{\text{noisy}}(\theta) = \mathrm{Tr}[H\rho_{\text{noisy}}(\theta)], \tag{A25}$$

i.e., we now have to optimize an inherently different loss function. Still, to estimate the noisy loss function $\mathcal{L}_{\text{noisy}}$, also here we will have to deal with statistical noise derived by the finite number of measurements. In particular, for the case of global depolarising noise with depolarising noise parameter $q \in [0, 1]$,

$$\mathcal{N}_i(\cdot) = (1-q)(\cdot) + q\mathrm{Tr}(\cdot)\frac{\mathbb{1}}{2^n}, \tag{A26}$$

the cost function will be

$$\mathcal{L}_{\text{noisy}}(\theta) = (1-q)^p\mathcal{L}(\theta) + [1-(1-q)^p]\frac{\mathrm{Tr}(H)}{2^n}. \tag{A27}$$

Therefore, in this case, the landscape of the cost function will be rescaled and shifted, but will preserve features of the noiseless landscape like the position of saddle points.

Proof. We have

$$\begin{aligned}
\rho_{\text{noisy}}(\theta) &= (1-q)\mathcal{N}_p \circ \mathcal{U}_p \circ \cdots \circ \mathcal{N}_2 \circ \mathcal{U}_2[\mathcal{U}_1(\rho_0)] + q\frac{\mathbb{1}}{2^n} \\
&= (1-q)^2\mathcal{N}_p \circ \mathcal{U}_p \circ \cdots \circ \mathcal{N}_3 \circ \mathcal{U}_3[\mathcal{U}_2 \circ \mathcal{U}_1(\rho_0)] \\
&\quad + q[(1-q)+1]\left(\frac{\mathbb{1}}{2^n}\right) \\
&= (1-q)^p\rho(\theta) + q\left[\sum_{k=0}^{p-1}(1-q)^k\right]\frac{\mathbb{1}}{2^n} \\
&= (1-q)^p\rho(\theta) + q\frac{1-(1-q)^p}{q}\frac{\mathbb{1}}{2^n}.
\end{aligned}$$

Plugging such a state into the definition of the noisy cost function (A25), we have the result. ■

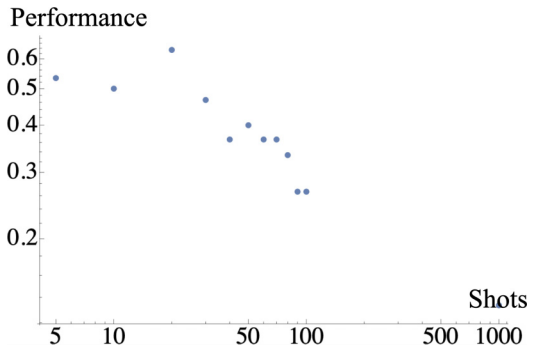

FIG. 8. We quantify the performance against the number of shots of the quantum noise by the probability of saddle-point avoidance for 100 different independent instances with the same initial conditions.

3. Additional numerical results

In this Appendix, we show additional numerical results to those reported in the main text. In Fig. 8, we show the estimated probability of avoiding the saddle point as a function of the number of shots, for the loss function given by the expectation value of the local Pauli Hamiltonian $H = \sum_{i=1}^{N=4} Z_i$ over the circuit qml.StronglyEntanglingLayers (see Fig. 9) in PENNYLANE [17] where two layers have been used.

One could directly extend our results towards the description of situations involving more qubits. Here, we consider eight qubits and two layers of quantum gates. Now the loss landscape is richer and we can converge at more integers. For instance, we find convergence at -3, -4, -5, and -6 for different initial conditions. Figure 10 illustrates saddle-point avoidance with different noise levels when noise is selected from Gaussian distributions. Figure 11 illustrates the performances among different sizes of noise levels and one can again find a critical value of the noise which leads to the saddle-point avoidance. In Fig. 12, we depict the performances as a function of the noise level obtained for the H_2 molecule experiment.

Furthermore, we try to find the relation between the convergence time T and the noise size $r \sim \epsilon$. With the same setup, we plot the dependence between the convergence time (the time where we approximately get the true minimum) and the size of the noise in the Gaussian distribution case, in Fig. 13. We find that the convergence time indeed decays when we add more noise, and we fit the scaling and find where $T \sim \#/\epsilon^{0.6}$, which is consistent with the bound $T \sim \#/\epsilon^2$ in theory. In Appendix A 4, we provide a heuristic derivation

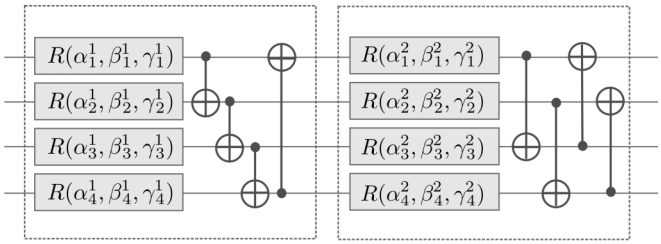

FIG. 9. A four-qubit example of one strongly entangling layer as given in qml.StronglyEntanglingLayers in PENNYLANE. The figure has been adopted from the documentation of PENNYLANE [17].

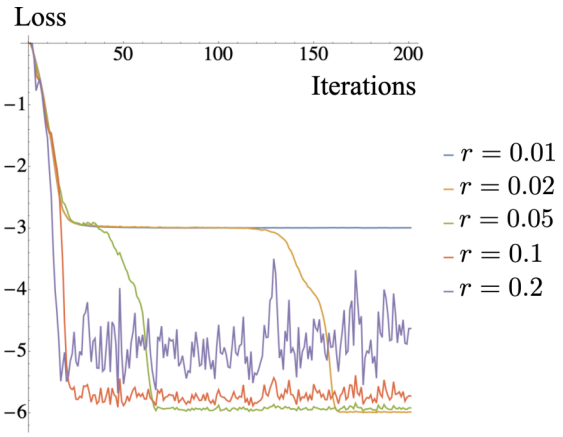

FIG. 10. Comparison of the loss evolution with or without noise with eight qubits. The noise has been drawn manually from Gaussian distributions, and we keep the same initial conditions. We use four different values of the noise norms.

on the scaling $T \sim 1/\epsilon^2$ by dimensional analysis and other analytic heuristics.

4. Analytic heuristics

In this Appendix, we provide a set of analytic heuristics about predicting the noisy convergence and the critical noise with significant improvements in performance. Our derivation is physical and heuristic, but we expect that they will be helpful to understand the nature of the noisy dynamics during gradient descent in the quantum devices. The developed results corroborate the idea that a balance between too little and too much noise will have to be struck.

a. Brownian motion and the Polya's constant

One of the simplest heuristics about noisy gradient descent is the theory of Brownian motion. Define $p(d)$, also known as Polya's constant, as the likelihood that a random walk on a d-dimensional lattice has the capability to return to its starting point. It has been proven that [45]

$$p(1) = p(2) = 1\,, \tag{A28}$$

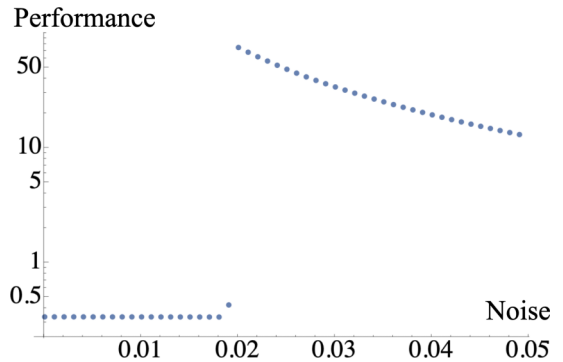

FIG. 11. We quantify the performance against the size of the noise r (classical Gaussian noise) by $1/(\mathcal{L} - \mathcal{L}_{\text{opt}})$. We again have eight qubits.

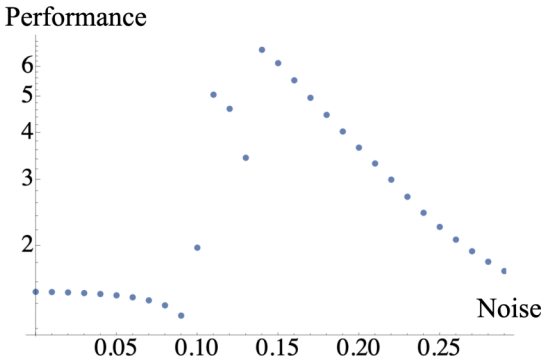

FIG. 12. In the Hydrogen VQE example, we quantify the performance against the size of the noise r (classical Gaussian noise) by $1/(\mathcal{L} - \mathcal{L}_{\text{opt}})$.

but

$$p(d \geqslant 3) < 1\,. \tag{A29}$$

In fact, $d \mapsto p(d)$ has the closed formula [46]

$$p(d) = 1 - \left\{ \int_0^\infty \left[I_0\left(\frac{t}{d}\right) \right]^d e^{-t} dt \right\}^{-1}, \tag{A30}$$

where $d > 3$ is the number of training parameters in our case, and I is the modified Bessel function of the first kind. One could compute numerical values of the probability $p(d)$ for increasing d. From $d = 4$ to $d = 8$, it changes monotonically from 0.19 to 0.07. It is hard to accurately compute the integral because of damping, but it is clear that it is decaying and will vanish for large d. In our problem, we could regard the process of noisy gradient descent as random walks in the space of variational angles. One could regard the returning probability roughly as the probability of coming back to the saddle point from the minimum. Thus, the statement about lattice random walk gives us intuition that it is less likely to return back when we have a large number of variational angles.

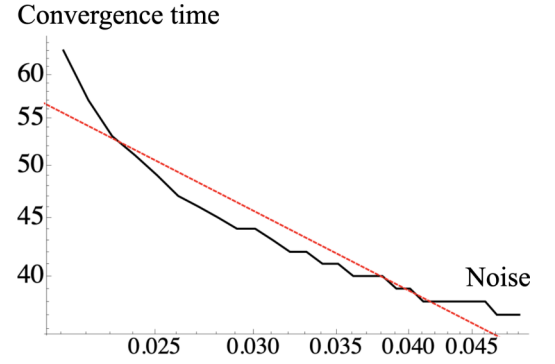

FIG. 13. The relationship between the convergence time T and the size of the noise, $r \sim \epsilon$. The data are plotted in black, and we fit the data using $\#/\epsilon^{\Delta}$ in red, and get $\Delta \approx 0.6$. The setup is the same as before: We use four qubits and two layers for our first example Hamiltonian and we use Gaussian noise simulation.

b. Guessing $1/\epsilon^2$ by dimensional analysis

One primary progress of the technical result presented in Ref. [36] is the $1/\epsilon^2$ dependence on the convergence time T with the size of the noise $\epsilon > 0$. Here, we show that one could guess such a result in the small-η limit (where η is the learning rate) simply by dimensional analysis. Starting from the definition of the gradient-descent algorithm,

$$\delta\theta_i = \theta_i(t+1) - \theta_i(t) = -\eta \frac{\partial \mathcal{L}}{\partial \theta_i}, \tag{A31}$$

we can instead study the variation of the loss function,

$$\delta\mathcal{L} = \mathcal{L}(t+1) - \mathcal{L}(t) \approx \sum_i \frac{\partial \mathcal{L}}{\partial \theta_i}\delta\theta_i = -\eta \sum_i \frac{\partial \mathcal{L}}{\partial \theta_i}\frac{\partial \mathcal{L}}{\partial \theta_i}$$

$$= -4\eta \sum_i \frac{\partial \sqrt{\mathcal{L}}}{\partial \theta_i}\frac{\partial \sqrt{\mathcal{L}}}{\partial \theta_i}\mathcal{L}. \tag{A32}$$

Here, we use the assumption where η is small, such that we could expand the loss-function change $\delta\mathcal{L}$ by the first-order Taylor expansion. Now we define

$$K_{\mathcal{L}} := 4\sum_i \frac{\partial \sqrt{\mathcal{L}}}{\partial \theta_i}\frac{\partial \sqrt{\mathcal{L}}}{\partial \theta_i}, \tag{A33}$$

and we have

$$\delta\mathcal{L} = -\eta K_{\mathcal{L}}\mathcal{L}. \tag{A34}$$

If $K_{\mathcal{L}}$ is a constant (and we could assume this is true since we are doing dimensional analysis), we get

$$\mathcal{L}(t) = (1-\eta K_{\mathcal{L}})^t \approx e^{-\eta K_{\mathcal{L}} t}. \tag{A35}$$

In general, we can assume a time-dependent solution as

$$\mathcal{L}(t) = [1-\eta K_{\mathcal{L}}(t)]^t \approx e^{-\eta K_{\mathcal{L}}(t) t}. \tag{A36}$$

Now let us think about how the scaling of convergence time will be with noise. First, in the $\eta \to 0$ limit, for small η the convergence time would get smaller, not larger (since it is immediately dominated by noise). So it is not possible that $T \sim 1/\eta$ to some powers in η. For this reason, the only possibility left is

$$T = \mathcal{O}(1) + \mathcal{O}(\eta) + \mathcal{O}(\eta^2)\cdots, \tag{A37}$$

in the scaling of η. We will thus focus on the first $\mathcal{O}(1)$ term in the small-η limit. Furthermore, from the form $e^{-\eta K_{\mathcal{L}} t}$, we know that $T \sim 1/K_{\mathcal{L}}$.

Now let us count the dimension, assuming θ_i has the θ-dimension 1 and L has the θ-dimension 0. From the gradient-descent formula, η has the θ-dimension 2, $K_{\mathcal{L}}$ has the θ-dimension -2, and ϵ has the θ-dimension 1. The time T is dimensionless since $\eta K_{\mathcal{L}} T$ is dimensionless and appears in the exponent. Thus, since we know that $T \sim 1/K_{\mathcal{L}}$, there must be an extra factor balancing the θ-dimension of $K_{\mathcal{L}}$. The only choice is ϵ^2, and we cannot use η because we are studying the term with the η-scaling $\mathcal{O}(1)$. Thus, we immediately get $T \sim 1/(K_{\mathcal{L}}\epsilon^2)$. That is how we get the dependence $T \sim 1/\epsilon^2$ by dimensional analysis. Note that the estimation only works in the small-η limit. More generally, we have

$$T = \sum_{m,n>2m} \mathcal{O}\left(\frac{\eta^{n-2m}}{K_{\mathcal{L}}^m \epsilon^n}\right) \tag{A38}$$

if we assume that the expression of T is analytic.

c. Large-width limit

The dependence $T \sim 1/\epsilon^2$ can also be made plausible using the *quantum neural tangent kernel* (QNTK) theory. The QNTK theory has been established [47–49] in the limit where we have a large number of trainable angles d and a small learning rate η, with the quadratic loss function. According to Ref. [49], we use the loss function

$$\mathcal{L}(\theta) = \tfrac{1}{2}[\langle\Psi_0|U^\dagger(\theta)OU(\theta)|\Psi_0\rangle - O_0]^2 =: \tfrac{1}{2}\varepsilon^2. \tag{A39}$$

Here, we make predictions on the eigenvalue of the operator O towards O_0. And we use $U(\theta)$ as the variational ansatz. The gradient-descent algorithm is

$$\theta_i(t+1) - \theta_i(t) =: \delta\theta_i = -\eta\frac{\partial \mathcal{L}}{\partial \theta_i} \tag{A40}$$

when there is no noise. Furthermore, we hereby model the noise by adding Gaussian random variables in each step of the update. Those random fluctuations are independently distributed through $\Delta\theta_i \sim \mathcal{N}(0,\epsilon^2)$. Now, in the limit where d is large, we have an analytic solution of the convergence time, given by

$$T \approx \frac{\ln\left(\frac{\epsilon}{\sqrt{2\varepsilon^2(0)\eta - \varepsilon^2(0)\eta^2 K + \epsilon^2}}\right)}{\ln(1-\eta K)}, \tag{A41}$$

where $K := K_{\mathcal{L}}/2$. In the small-η limit, we have

$$T \approx \frac{\varepsilon^2(0)}{\epsilon^2 K}. \tag{A42}$$

This gives substance to the claim in the dimensional analysis.

d. Critical noise from random walks

Moreover, using the result from Ref. [49], we can also estimate the critical noise ϵ_{cri}, namely, the critical value of phase transition of the noise size that leads to better performance and avoids the saddle points.

In particular, here we will be interested in the case where the saddle-point avoidance is triggered purely by random walks without any extra potential. The assumption, although it may not be real in the practical loss-function landscape, might still provide some useful guidance. According to Ref. [49], we have

$$\overline{\varepsilon^2}(t) = (1-\eta K)^{2t}\left(\varepsilon^2(0) - \frac{\epsilon^2}{\eta(2-\eta K)}\right) + \frac{\epsilon^2}{\eta(2-\eta K)}. \tag{A43}$$

Here, $\overline{\varepsilon^2}$ is the variance of the residual training error ε after averaging over the realizations of the noise. Imagine that now the gradient-descent process is running from the saddle point to the exact local minimum; we have

$$\frac{1}{2}(|\varepsilon_{\text{saddle}}|^2 - |\varepsilon_{\min}|^2) = \Delta_{\mathcal{L}} \sim \frac{\epsilon^2}{2\eta(2-\eta K)}, \tag{A44}$$

where $\Delta_{\mathcal{L}}$ is the distance of the loss function from the saddle point to the local minimum (defined also in the main text),

$\Delta_{\mathcal{L}} = \mathcal{L}_{\text{saddle}} - \mathcal{L}_{\text{mininum}} = \frac{1}{2}(|\varepsilon_{\text{saddle}}|^2 - |\varepsilon_{\text{min}}|^2)$. So we get an estimate of the critical noise,

$$\epsilon_{\text{cri}}^2 \sim \Delta_{\mathcal{L}}[2\eta(2-\eta K)] \sim 4\eta\Delta_{\mathcal{L}}\,. \tag{A45}$$

Here, on the most right-hand side of the formula, we use the approximation where η is small enough. This formula might be more generic beyond QNTK since one could regard it as an analog of Einstein's formula of *Brownian motion*,

$$\overline{x^2}(t) = 2Dt, \tag{A46}$$

with the averaging moving distance square $\overline{x^2}$, mass diffusivity D, and time t in the Brownian motion.

One can also show such a scaling in the linear model. Say that we have a linear loss function

$$\mathcal{L} = \sum_{\mu} c_\mu \theta_\mu + b, \tag{A47}$$

with constants c_μ and b. For simplicity, we assume that the initialization $\theta(0)$ makes $\mathcal{L}[\theta(0)] = \mathcal{L}(0) > 0$. The gradient-descent relation is

$$\delta\theta_\mu = \theta_\mu(t+1) - \theta_\mu(t) = -\eta\frac{\partial\mathcal{L}}{\partial\theta_\mu} = -\eta c_\mu. \tag{A48}$$

One can find the closed-form solution,

$$\theta_\mu(t) = \theta_\mu(0) - \eta t c_\mu. \tag{A49}$$

It is also possible to identify the change of the loss function to be

$$\mathcal{L}(t) = \sum_{\mu} c_\mu\theta_\mu(0) + b - \eta t\sum_{\mu} c_\mu^2 = \mathcal{L}(0) - \eta t\sum_{\mu} c_\mu^2. \tag{A50}$$

The convergence time can be estimated as

$$T = \frac{\mathcal{L}(0)}{\eta\sum_\mu c_\mu^2}. \tag{A51}$$

Now, instead, we add a random $\xi_\mu(t)$ in the gradient-descent dynamics, which is following the normal distribution $\xi_\mu(t) \sim \mathcal{N}(0,\sigma_\mu^2)$. Now, the stochastic gradient-descent equation is

$$\delta\theta_\mu = \theta_\mu(t+1) - \theta_\mu(t) = -\eta\frac{\partial\mathcal{L}}{\partial\theta_\mu} + \xi_\mu = -\eta c_\mu + \xi_\mu, \tag{A52}$$

which gives the solution

$$\theta_\mu(t) = \theta_\mu(0) - \eta t c_\mu + \sum_{i=0}^{t-1}\xi_\mu(i). \tag{A53}$$

Thus, we get the loss function

$$\begin{aligned}\mathcal{L}(t) &= \sum_{\mu} c_\mu\theta_\mu(0) + b - \eta t\sum_{\mu} c_\mu^2 + \sum_{\mu,i=0}^{t-1} c_\mu\xi_\mu(i)\\ &= \mathcal{L}(0) - \eta t\sum_{\mu} c_\mu^2 + \sum_{\mu,i=0}^{t-1} c_\mu\xi_\mu(i).\end{aligned} \tag{A54}$$

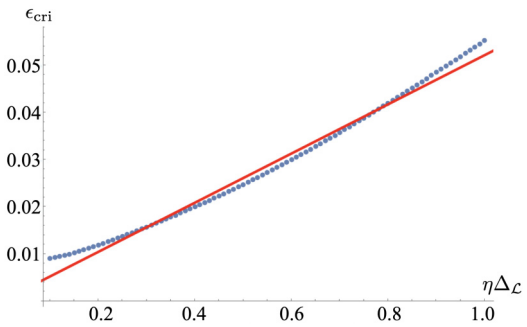

FIG. 14. The dependence of the critical noise ϵ_{cri} on $\eta\Delta_{\mathcal{L}}$ in the example of four qubits. Here, we fit the dependence by the linear relation $\epsilon_{\text{cri}} = c_\epsilon\eta\Delta_{\mathcal{L}}$, where $c_\epsilon = 0.052\,140\,4$.

The critical point $\sigma_\mu = \varepsilon_{\text{cri}}$ can be identified as

$$\eta t\sum_{\mu} c_\mu^2 \sim \epsilon_{\text{cri}}\sqrt{t}\left(\sum_{\mu} c_\mu^2\right)^{1/2}, \tag{A55}$$

where the standard deviation of the noise term will compensate the decay. Thus, we get

$$\epsilon_{\text{cri}} \sim \eta\sqrt{t}\left(\sum_{\mu} c_\mu^2\right)^{1/2}. \tag{A56}$$

In the limit where the noise levels are small, we can study the behavior in the late-time limit,

$$t = T = \frac{\mathcal{L}(0)}{\eta\sum_\mu c_\mu^2}. \tag{A57}$$

So we get

$$\epsilon_{\text{cri}} \sim \sqrt{\eta\mathcal{L}(0)} \sim \sqrt{\eta\Delta_{\mathcal{L}}}, \tag{A58}$$

which is

$$\epsilon_{\text{cri}}^2 \sim \eta\Delta_{\mathcal{L}}. \tag{A59}$$

Thus, the linear model result is consistent with the derivation using QNTK with the quadratic loss.

e. Phenomenological critical noise

In practice, random walks may not be the only source triggering the saddle-point avoidance, leading to the $\sqrt{t}$ scaling in loss functions. Since saddle points have negative Hessian eigenvalues, those directions will provide driven forces with linear contributions $\propto t$ in the loss function. In the linear model, we can estimate the critical noise as

$$\eta t\Delta_{\mathcal{L}} \sim \epsilon_{\text{cri}} t, \tag{A60}$$

which leads to the linear relation

$$\epsilon_{\text{cri}} \sim \eta\Delta_{\mathcal{L}}. \tag{A61}$$

In Fig. 14, we show the dependence of the critical noise ϵ_{cri} on $\eta\Delta_{\mathcal{L}}$ in our numerical example with four qubits from Fig. 3 of the main text and its linear fitting. We find that our theory is justified for a decent range of learning rate. In our numerical data, in smaller learning rates, the increase of the critical noise might be smoother, while for larger critical noise, the growth

is closer to linear scaling. If we fit for the exponent of $\epsilon_{\rm cri} \sim (\eta\Delta_L)^{\Delta_{\rm cri}}$, we get $\Delta_{\rm cri} \approx 0.8722$.

One could also transform critical learning rates towards the number of shots if the noises are dominated from quantum measurements. One can assume the scaling

$$\epsilon_{\rm cri} \sim \frac{\eta}{\sqrt{N_{\rm cri}}}, \tag{A62}$$

and we can obtain the optimal number of shots,

$$N_{\rm cri} = N_{\rm cri} = c_N \eta^{2-2\Delta_{\rm cri}} \Delta_{\mathcal{L}}^{-2\Delta_{\rm cri}}. \tag{A63}$$

One can then take $\Delta_{\rm cri} = 1$ as a good approximation, assuming that the saddle-point avoidance is dominated by negative saddle-point eigenvalues. $c_{\rm cri}$ is a constant depending on the circuit architecture and the loss-function landscapes. This formalism could be useful to estimate the optimal number of shots used in variational quantum algorithms. For instance, we take $\Delta_{\rm cri} = 1$ and we get

$$N_{\rm cri} = c_N \Delta_{\mathcal{L}}^{-2}. \tag{A64}$$

In the situation of Fig. 3 of the main text, we obtain

$$\epsilon = c_\eta \frac{\eta}{\sqrt{N}}, \tag{A65}$$

with c_η estimated as $c_\eta \approx 1.197\,33$ from QISKIT. So we get

$$N_{\rm cri} = \frac{c_\eta^2}{c_\epsilon^2} \frac{1}{\Delta_{\mathcal{L}}^2} = 131.8, \tag{A66}$$

which is the optimal numbers of shots in this experiment with pure measurement noises. Here, $c_N = c_\eta^2/c_\epsilon^2$.

[1] R. P. Feynman, Quantum mechanical computers, Found. Phys. **16**, 507 (1986).

[2] D. Deutsch, Quantum theory, the Church-Turing principle and the universal quantum computer, Proc. Roy. Soc. A **400**, 97 (1985).

[3] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. S. L. Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi, B. Foxen, A. Fowler *et al.*, Quantum supremacy using a programmable superconducting processor, Nature (London) **574**, 505 (2019).

[4] S. Boixo, S. V. Isakov, V. N. Smelyanskiy, R. Babbush, N. Ding, Z. Jiang, M. J. Bremner, J. M. Martinis, and H. Neven, Characterizing quantum supremacy in near-term devices, Nat. Phys. **14**, 595 (2018).

[5] D. Hangleiter and J. Eisert, Computational advantage of quantum random sampling, Rev. Mod. Phys. **95**, 035001 (2023).

[6] P. Jurcevic, A. Javadi-Abhari, L. S. Bishop, I. Lauer, D. F. Bogorin, M. Brink, L. Capelluto, O. Günlük, T. Itoko, N. Kanazawa, A. Kandala, G. A. Keefe, K. Krsulich, W. Landers, E. P. Lewandowski, D. T. McClure, G. Nannicini, A. Narasgond, H. M. Nayfeh, E. Pritchett *et al.*, Demonstration of quantum volume 64 on a superconducting quantum computing system, Quantum Sci. Technol. **6**, 025020 (2021).

[7] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, Variational quantum algorithms, Nat. Rev. Phys. **3**, 625 (2021).

[8] 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, Nat. Commun. **5**, 4213 (2014).

[9] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. W. Chow, and J. M. Gambetta, Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets, Nature (London) **549**, 242 (2017).

[10] J. R McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, The theory of variational hybrid quantum-classical algorithms, New J. Phys. **18**, 023023 (2016).

[11] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, arXiv:1411.4028.

[12] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices, Phys. Rev. X **10**, 021067 (2020).

[13] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W.-K. Mok, S. Sim, L.-C. Kwek, and A. Aspuru-Guzik, Noisy intermediate-scale quantum (NISQ) algorithms, Rev. Mod. Phys. **94**, 015004 (2022).

[14] J. Tilly, H. Chen, S. Cao, D. Picozzi, K. Setia, Y. Li, E. Grant, L. Wossnig, I. Rungger, G. H. Booth, and J. Tennyson, The variational quantum eigensolver: A review of methods and best practices, Phys. Rep. **986**, 1 (2022).

[15] J. Preskill, Quantum computing in the NISQ era and beyond, Quantum **2**, 79 (2018).

[16] M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, and N. Killoran, Evaluating analytic gradients on quantum hardware, Phys. Rev. A **99**, 032331 (2019).

[17] V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, and N. Killoran, On a problem of probability theory concerning the random walk in a street network, arXiv:1811.04968.

[18] https://pennylane.ai/qml/demos/tutorial_vqe_qng.html#stokes2019.

[19] R. Sweke, F. Wilde, J. Meyer, M. Schuld, P. K. Fährmann, B. Meynard-Piganeau, and J. Eisert, Stochastic gradient descent for hybrid quantum-classical optimization, Quantum **4**, 314 (2020).

[20] A. J. Bray and D. S. Dean, Statistics of critical points of Gaussian fields on large-dimensional spaces, Phys. Rev. Lett. **98**, 150201 (2007).

[21] L. Bittel and M. Kliesch, Training variational quantum algorithms is NP-hard, Phys. Rev. Lett. **127**, 120502 (2021).

[22] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht, Gradient descent converges to minimizers, arXiv:1602.04915.

[23] S. S. Du, C. Jin, J. D. Lee, M. I. Jordan, A. Singh, and B. Poczos, Gradient descent can take exponential time to escape saddle points, Adv. Neur. Inf. Proc. Sys. **30**, 1067 (2017).

[24] C. Jin, P. Netrapalli, R. Ge, S. M. Kakade, and M. I. Jordan, On nonconvex optimization for machine learning: Gradients, stochasticity, and saddle points, arXiv:1902.04811.

[25] P. Jain and P. Kar, Non-convex optimization for machine learning, FNT Mach. Learn. **10**, 142 (2017).

[26] S. Duffield, M. Benedetti, and M. Rosenkranz, Bayesian learning of parameterised quantum circuits, Mach. Learn.: Sci. Technol. **4**, 025007 (2023).

[27] K. Borras, S. Y. Chang, L. Funcke, M. Grossi, T. Hartung, K. Jansen, D. Kruecker, S. Kühn, F. Rehm, C. Tüysüz, and S. Vallecorsa, Impact of quantum noise on the training of quantum generative adversarial networks, J. Phys.: Conf. Ser. **2438**, 012093 (2023).

[28] M. Oliv, A. Matic, T. Messerer, and J. M. Lorenz, Evaluating the impact of noise on the performance of the variational quantum eigensolver, arXiv:2209.12803.

[29] T. L. Patti, K. Najafi, X. Gao, and S. F. Yelin, Entanglement devised barren plateau mitigation, Phys. Rev. Res. **3**, 033090 (2021).

[30] A. Gu, A. Lowe, P. A. Dub, P. J. Coles, and A. Arrasmith, Adaptive shot allocation for fast convergence in variational quantum algorithms, arXiv:2108.10434.

[31] L. Gentini, A. Cuccoli, S. Pirandola, P. Verrucchi, and L. Banchi, Noise-resilient variational hybrid quantum-classical optimization, Phys. Rev. A **102**, 052414 (2020).

[32] G. De Palma, M. Marvian, C. Rouzé, and D. S. França, Limitations of variational quantum algorithms: A quantum optimal transport approach, PRX Quantum **4**, 010309 (2023).

[33] D. S. França and R. García-Patrón, Limitations of optimization algorithms on noisy quantum devices, Nat. Phys. **17**, 1221 (2021).

[34] S. Wang, E. Fontana, M. Cerezo, K. Sharma, A. Sone, L. Cincio, and P. J. Coles, Noise-induced barren plateaus in variational quantum algorithms, Nat. Commun. **12**, 6961 (2021).

[35] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, Barren plateaus in quantum neural network training landscapes, Nat. Commun. **9**, 4812 (2018).

[36] S. S. Du, C. Jin, J. D. Lee, M. I. Jordan, B. Poczos, and A. Singh, Gradient descent can take exponential time to escape saddle points, arXiv:1705.10412.

[37] J. D. Lee, I. Panageas, G. Piliouras, M. Simchowitz, M. I. Jordan, and B. Recht, First-order methods almost always avoid saddle points, arXiv:1710.07406.

[38] A. A. Mele, A. Angrisani, S. Ghosh, S. Khatri, J. Eisert, D. S. França, and Y. Quek, Noise-induced shallow circuits and absence of barren plateaus, arXiv:2403.13927.

[39] R. Benzi, A. Sutera, and A. Vulpiani, The mechanism of stochastic resonance, J. Phys. A: Math. Gen. **14**, L453 (1981).

[40] D. Lykov, J. Wurtz, C. Poole, M. Saffman, T. Noel, and Y. Alexeev, Sampling frequency thresholds for the quantum advantage of the quantum approximate optimization algorithm, npj Quantum Inf **9**, 73 (2023).

[41] E. Fontana, N. Fitzpatrick, D. M. Ramo, R. Duncan, and I. Rungger, Evaluating the noise resilience of variational quantum algorithms, Phys. Rev. A **104**, 022403 (2021).

[42] T. Haug, K. Bharti, and M. S. Kim, Capacity and quantum geometry of parametrized quantum circuits, PRX Quantum **2**, 040309 (2021).

[43] https://github.com/junyuphybies/saddlepoints/.

[44] D. Patel, P. J. Coles, and M. M. Wilde, Variational quantum algorithms for semidefinite programming, Quantum **8**, 1374 (2024).

[45] G. Pólya, Über eine aufgabe der wahrscheinlichkeitsrechnung betreffend die irrfahrt im straßennetz, Math. Ann. **84**, 149 (1921).

[46] E. W. Montroll, Random walks in multidimensional spaces, especially on periodic lattices, J. Soc. Industr. Appl. Math. **4**, 241 (1956).

[47] J. Liu, F. Tacchino, J. R. Glick, L. Jiang, and A. Mezzacapo, Representation learning via quantum neural tangent kernels, PRX Quantum **3**, 030323 (2022).

[48] J. Liu, K. Najafi, K. Sharma, F. Tacchino, L. Jiang, and A. Mezzacapo, An analytic theory for the dynamics of wide quantum neural networks, Phys. Rev. Lett. **130**, 150601 (2023).

[49] J. Liu, Z. Lin, and L. Jiang, Laziness, barren plateau, and noise in machine learning, Mach. Learn.: Sci. Technol. **5**, 015058 (2024).