Error mitigation in variational quantum eigensolvers using tailored probabilistic machine learning

Tao Jiang,[1] John Rogers,[2] Marius S. Frank,[3] Ove Christiansen ●,[3] Yong-Xin Yao ●,[1,4,*] and Nicola Lanatà ●[5,6,†]

[1]*Ames National Laboratory, Ames, Iowa 50011, USA*
[2]*Department of Physics and Astronomy, Texas A&M University, College Station, Texas 77845, USA*
[3]*Department of Chemistry, Aarhus University, 8000 Aarhus C, Denmark*
[4]*Department of Physics and Astronomy, Iowa State University, Ames, Iowa 50011, USA*
[5]*School of Physics and Astronomy, Rochester Institute of Technology, 84 Lomb Memorial Drive, Rochester, New York 14623, USA*
[6]*Center for Computational Quantum Physics, Flatiron Institute, New York, New York 10010, USA*

Quantum computing technology has the potential to revolutionize the simulation of materials and molecules in the near future. A primary challenge in achieving near-term quantum advantage is effectively mitigating the noise effects inherent in current quantum processing units (QPUs). This challenge is also decisive in the context of quantum-classical hybrid schemes employing variational quantum eigensolvers (VQEs) that have attracted significant interest in recent years. In this paper, we present a method that employs parametric Gaussian process regression (GPR) within an active learning framework to mitigate noise in quantum computations, focusing on VQEs. Our approach, grounded in probabilistic machine learning, exploits a custom prior based on the VQE ansatz to capture the underlying correlations between VQE outputs for different variational parameters, thereby enhancing both accuracy and efficiency. We demonstrate the effectiveness of our method on a two-site Anderson impurity model and a eight-site Heisenberg model, using the IBM open-source quantum computing framework, Qiskit, showcasing substantial improvements in the accuracy of VQE outputs while reducing the number of direct QPU energy evaluations. This paper contributes to the ongoing efforts in quantum-error mitigation and optimization, bringing us a step closer to realizing the potential of quantum computing in quantum matter simulations.

I. INTRODUCTION

Quantum computers hold the promise of dramatically enhancing our ability to simulate quantum-mechanical systems. Computational chemistry is expected to be one of the most promising fields to benefit significantly from quantum technologies within the next few years [1–6], as the number of qubits required to represent the active degrees of freedom for small molecules is relatively low compared to that of larger systems. Furthermore, quantum embedding (QE) methods [7,8] could allow us to benefit from quantum devices for simulating larger systems (including molecules and periodic materials) by employing QPUs to handle only the most important degrees of freedom, while treating others (at the mean-field level) on classical devices [9–12].

A major obstacle to realizing this ambitious program is the insufficient reliability of data produced by quantum devices, even though several effective error-mitigation methodologies

have already been developed and implemented [13–15]. In this paper, we propose a method to address this problem that is complementary to existing approaches and has the potential to significantly improve the accuracy achieved in VQE frameworks [16–19].

Existing error-mitigation techniques focus on enhancing the computation of expectation values with respect to a parametrized quantum circuit at a single point in parameter space [13–15,20–27]. For instance, the zero-noise extrapolation technique measures an observable at a single parameter point for a set of equivalent circuits with different noise strengths, followed by fitting with analytical functions such as polynomials and extrapolating the expectation value to the zero-noise limit [13,14,20,22,23]. Probabilistic error cancellation and randomized compiling approaches convert the expectation value with respect to a parametrized circuit at a single point to a sum of estimations with equivalent random circuits, effectively transforming the coherent noise into stochastic error [15,20,21,24].

In contrast to mitigating the error of VQE measurements for each individual variational state, as in the methods mentioned above, we propose a complementary approach that aims to mitigate the error for the entire variational landscape simultaneously by exploiting the underlying correlations between VQE outputs for different variational parameters. Specifically, we employ a custom probabilistic machine learning method based on GPR (rooted in Bayesian statistics)

---

[28], tailored to incorporate the specific structure of any given parametrized quantum circuit. This method allows us to include within the computation our prior knowledge about general mathematical structure of the variational ansatz. This approach can be combined with any of the error-mitigation techniques mentioned above, substantially improving the accuracy of VQE calculations.

The paper is structured as follows. In Secs. II, III, and IV we present our error-mitigation formalism from a general perspective; in Secs. V and VI we show benchmark calculations of a Fermionic impurity model and a Heisenberg model using our method within the IBM quantum-computing framework Qiskit [29].

II. HAMILTONIAN-INDEPENDENT FEATURES OF THE VARIATIONAL LANDSCAPE

Let us consider a generic Hamiltonian $\hat{H}$ and a variational space $\{|\Psi(\boldsymbol{\theta})\rangle\}$, where the components of $\boldsymbol{\theta} = (\theta^1, .., \theta^d)$ are real numbers parametrizing the trial quantum states. Our goal is to determine

$$\bar{\boldsymbol{\theta}} = \mathrm{argmin}_{\boldsymbol{\theta}} \, \mathcal{E}(\boldsymbol{\theta}), \tag{1}$$

$$\mathcal{E}(\boldsymbol{\theta}) = \langle \phi(\boldsymbol{\theta})| \, \hat{H} \, |\phi(\boldsymbol{\theta})\rangle. \tag{2}$$

Within VQE frameworks, we prepare the state $|\phi(\boldsymbol{\theta})\rangle$ using parametrized quantum gates, and estimate $\mathcal{E}(\boldsymbol{\theta})$ from a series of quantum measurements for each $\boldsymbol{\theta}$. However, outputs from currently available intermediate-scale quantum devices are affected by spurious effects such as decoherence and hardware imperfections, resulting in both random and systematic noise [30].

The key idea at the basis of our method for mitigating the noise in VQE frameworks is to exploit the fact that the variational landscape satisfies exact properties that are known beforehand. Therefore, corrections to the VQE measurements, which may be affected by spurious noise effects that potentially violate such exact properties, can be rationally enforced for consistency with them. As we are going to show, this approach can effectively mitigate noise by aligning the VQE output with the inherent structure of the variational landscape. Here we illustrate this point focusing on a generic ansatz represented as follows:

$$|\phi(\boldsymbol{\theta})\rangle = \hat{U}(\boldsymbol{\theta})|\phi_0\rangle. \tag{3}$$

Here, $|\phi_0\rangle$ is a single-particle state (e.g., the Hartree-Fock solution of $\hat{H}$), and $\hat{U}(\boldsymbol{\theta})$ is a unitary transformation given by

$$\hat{U}(\boldsymbol{\theta}) = \prod_{l=1}^{d} \prod_{m=1}^{M_l} e^{i\hat{G}_{lm}\frac{\theta^l}{2}}, \tag{4}$$

where θ^l are variational parameters, the generators $\hat{G}_{lm}$ are Pauli strings expressed in the Hartree-Fock basis, d is the number of variational parameters, and M_l is the number of Pauli strings for each variational parameter. Note that a Pauli string is defined as a generic tensor product of Pauli operators $\hat{P} = \otimes_k \hat{P}^k$, where $\hat{P}^k \in \{\mathbb{1}, X^k, Y^k, Z^k\}$, k is a generic site label, X^k, Y^k, and Z^k are the corresponding local Pauli operators and $\mathbb{1}$ is the identity.

A key observation is that, as noted in Refs. [31,32], since $\hat{G}_{lm}^2 = \hat{I} \, \forall \, l, m$, where $\hat{I}$ is the identity, we have

$$e^{i\hat{G}_{lm}\frac{\theta^l}{2}} = \cos\left(\frac{\theta^l}{2}\right)\hat{I} + i \sin\left(\frac{\theta^l}{2}\right)\hat{G}_{lm}. \tag{5}$$

Therefore, the variational energy [Eq. (2)] can be expanded in the following form:

$$\mathcal{E}(\boldsymbol{\theta}) = \sum_{i_1=0}^{2M_1} ... \sum_{i_d=0}^{2M_d} \xi_{i_1,..,i_d} \prod_{l=1}^{d} \cos\left(\frac{\theta^l}{2}\right)^{i_l} \sin\left(\frac{\theta^l}{2}\right)^{2M_l-i_l}$$

$$= \sum_{\mathbf{s}=1}^{S} \xi_{\mathbf{s}} T_{\mathbf{s}}(\boldsymbol{\theta}), \tag{6}$$

where we use for convenience a composite index $\mathbf{s} = (i_1, .., i_d)$ to label the coefficients $\xi_{\mathbf{s}}$, $T_{\mathbf{s}}(\boldsymbol{\theta})$ are the corresponding trigonometric functions, and $S = \prod_{l=1}^{d}(2M_l + 1)$. Note that the coefficients $\xi_{\mathbf{s}}$ are not fixed by the variational ansatz, but they depend on the specific Hamiltonian operator $\hat{H}$.

It is useful to note that the functions $T_{\mathbf{s}}(\boldsymbol{\theta})$ have periodicity 2π, and the condition expressed by Eq. (6) can be conveniently reformulated in terms of the corresponding plane-waves orthonormal basis as follows:

$$\mathcal{E}(\boldsymbol{\theta}) = \sum_{k_1=-M_1}^{M_1} ... \sum_{k_d=-M_d}^{M_d} \epsilon_{k_1,..,k_d} \prod_{l=1}^{d} \frac{e^{ik_l\theta_l}}{(2\pi)^{1/2}}$$

$$= \sum_{\mathbf{k}} \epsilon_{\mathbf{k}} \frac{e^{i\mathbf{k}\boldsymbol{\theta}}}{(2\pi)^{d/2}}, \tag{7}$$

where $\mathbf{k} = (k_1, \ldots, k_d)$ and each k_l is an integer running from $-M_l$ to M_l. Note that this reformulation in terms of the plane-wave basis does not alter the functional space spanned by the $T_{\mathbf{s}}(\boldsymbol{\theta})$ functions, thereby ensuring the exact representation of the variational energy landscape is preserved. This step shows that Eq. (6) encodes precise information about the smoothness of the energy landscape, guaranteeing that all Fourier components with $|k_l| > M_l$ are 0. The proposed approach involves utilizing the GPR framework to exploit such prior knowledge about global properties of the variational landscape. As we are going to show, since Eq. (7) correlates the data with each other, it can mitigate the error arising from individual QPU measurements, and reduce substantially the computational cost of the variational-energy minimization.

III. GPR-BASED ERROR MITIGATION

In this section, we demonstrate how to use the GPR framework in conjunction with Eq. (6) and the variational-energy data D measured on QPUs to compute a "posterior probability distribution" $P[\mathcal{E}|D]$ within the space of energy landscapes.

In the following sections, we show that the resulting posterior probability distribution can be efficiently employed for minimizing the variational energy in VQE frameworks, mitigating errors and significantly reducing the required number of quantum measurements.

A. Prior probability distribution associated with a variational ansatz

Suppose the Hamiltonian $\hat{H}$ is unspecified, but the variational ansatz [Eq. (3)] has been chosen. Our aim is to "learn" the energy landscape $\mathcal{E}(\boldsymbol{\theta})$ from data D obtained through quantum measurements, and express our prediction in the form of a "posterior probability distribution" $P(\mathcal{E}|D)$. To construct such a predictive model, we need to incorporate our prior knowledge about the function to be learned (i.e., the information available before any measurement) into the calculation.

In the GPR framework, this prior knowledge is encapsulated in a "prior probability distribution" $P[\mathcal{E}]$, which is conveniently assumed to be Gaussian. A significant benefit of assuming that $P[\mathcal{E}]$ is Gaussian is that all Gaussian distributions are entirely characterized by their corresponding "propagator" (or Kernel function), which is defined as follows:

$$K(\boldsymbol{\theta}, \boldsymbol{\theta}') = \int \mathcal{D}[\mathcal{E}] P[\mathcal{E}] \, \mathcal{E}(\boldsymbol{\theta}) \, \mathcal{E}(\boldsymbol{\theta}'), \qquad (8)$$

where $\mathcal{D}[\mathcal{E}]$ is the standard functional-integral measure, (see Appendix A for a more detailed introduction to GPR). A key feature of the proposed approach is that, unlike in standard GPR where the kernel function is typically chosen empirically from a list (including, e.g., the widely used "square exponential" kernel), here we *calculate* it from Eq. (6), as follows:

$$P[\mathcal{E}] \propto \int \prod_r d\xi_r \exp\left\{ -\frac{1}{2\eta^2} \int d\boldsymbol{\theta} \left[\mathcal{E}(\boldsymbol{\theta}) - \sum_{\mathbf{s}} \xi_{\mathbf{s}} T_{\mathbf{s}}(\boldsymbol{\theta}) \right]^2 \right\}$$
$$\times \exp\left\{ -\frac{1}{2} \sum_{\mathbf{s},\mathbf{s}'} \mathcal{A}_{\mathbf{s}\mathbf{s}'} \xi_{\mathbf{s}} \xi_{\mathbf{s}'} \right\}, \qquad (9)$$

where

$$d\boldsymbol{\theta} = \prod_{l=1}^{d} d\theta^l, \qquad (10)$$

where the integrals over $d\theta^l$ are all taken between $-\pi$ and π, η is considered in the limit $\eta \to 0$, and we have introduced a (strictly) positive-definite matrix $\mathcal{A}$, whose role is to encode a Gaussian probability distribution for the coefficients $\xi_{\mathbf{s}}$, represented as

$$p(\xi) \propto \exp\left\{ -\frac{1}{2} \sum_{\mathbf{s}\mathbf{s}'} \mathcal{A}_{ss'} \xi_{\mathbf{s}} \xi_{\mathbf{s}'} \right\}. \qquad (11)$$

The meaning of Eq. (9) is that, within our scenario, we know a priori that $\mathcal{E}(\boldsymbol{\theta})$ can be expressed as in Eq. (6), where $T_{\mathbf{s}}(\boldsymbol{\theta})$ are known analytical functions. However, additional information about $p(\xi)$ for the coefficients $\xi_{\mathbf{s}}$ is necessary to fully specify the prior probability distribution $P[\mathcal{E}]$.

The functional integral in Eq. (8) can be calculated explicitly, leading to

$$K(\boldsymbol{\theta}, \boldsymbol{\theta}') = \sum_{\mathbf{s},\mathbf{s}'=1}^{S} \Delta_{ss'} \, T_{\mathbf{s}}(\boldsymbol{\theta}) T_{\mathbf{s}'}(\boldsymbol{\theta}'), \qquad (12)$$

where the matrix

$$\Delta_{\mathbf{s}\mathbf{s}'} = \mathcal{A}_{\mathbf{s}\mathbf{s}'}^{-1} = \langle \xi_{\mathbf{s}} \xi_{\mathbf{s}'} \rangle_p \qquad (13)$$

fully encodes the Gaussian prior probability distribution $p(\xi)$.

Below we derive a few examples of ansatzes for the GPR prior, and calculate the corresponding kernel functions.

1. Kernel design I

A possible ansatz is to encode within the prior only the condition that the energy landscape must be bounded, which is true for all physical Hamiltonians,

$$p(\xi) \propto \exp\left\{ -\frac{t}{2} \int d\boldsymbol{\theta} \, \mathcal{E}^2(\boldsymbol{\theta}) \right\}$$
$$= \exp\left\{ -\frac{t}{2} \int d\boldsymbol{\theta} \left[\sum_{\mathbf{s}} \xi_{\mathbf{s}} T_{\mathbf{s}}(\boldsymbol{\theta}) \right]^2 \right\}, \qquad (14)$$

corresponding to

$$\mathcal{A}_{\mathbf{s}\mathbf{s}'} = t \int d\boldsymbol{\theta} \, T_{\mathbf{s}}(\boldsymbol{\theta}) T_{\mathbf{s}'}(\boldsymbol{\theta}). \qquad (15)$$

The parameter t encodes our prior information concerning the range of the variational energy, as the resulting probability distribution vanishes exponentially for $|\mathcal{E}(\boldsymbol{\theta})| \gg t^{-1/2}$.

A direct calculation of Eq. (12) shows that the corresponding Kernel function is the following:

$$K(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \frac{t^{-1}}{(2\pi)^d} \sum_{k_1=-M_1}^{M_1} \cdots \sum_{k_d=-M_d}^{M_d} e^{i\mathbf{k}(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2)}. \qquad (16)$$

It should be noted that the framework above has the flexibility of encoding in the prior also additional information that may be available to us for specific VQEs. As an example, in Sec. VI A we consider the example of a Heisenberg model such that, because of the specific generators $\hat{G}_{lm}$ used, the unitary transformation $\hat{U}(\boldsymbol{\theta})$ has periodicity π instead of 2π. Since, under such hypothesis, only the coefficients $\epsilon_{\mathbf{k}}$ of Eq. (7) with even k_l are nonzero, we have that

$$K(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \frac{t^{-1}}{(2\pi)^d} \sum_{k_1=-M_1/2}^{M_1/2} \cdots \sum_{k_d=-M_d/2}^{M_d/2} e^{i2\mathbf{k}(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2)}. \qquad (17)$$

2. Kernel design II

While the derivation of the Kernel function in Eqs. (16) and (17) is conceptually correct, its applicability is limited by exponential term growth with increasing variational parameters d. This poses a scalability challenge. Here we derive an alternative kernel function that also encodes the same information about the mathematical structure of the VQE landscape, but fully resolves scalability for VQEs with many variational parameters.

Let us consider Eq. (6), representing the VQE energy landscape. We note that of the prefactor $\xi_{i_1,\dots,i_d}$ of each function

$$T_{i_1,\dots,i_d}(\boldsymbol{\theta}) = \prod_{l=1}^{d} \cos\left(\frac{\theta_l}{2} \right)^{i_l} \sin\left(\frac{\theta_l}{2} \right)^{2M_l - i_l} \qquad (18)$$

is the sum of $\prod_{l=1}^{d} \binom{2M_l}{i_l}$ expectation values with respect to $|\Psi_0\rangle$ of different complex operators. Based on this

observation, we propose the following ansatz:

$$\langle \xi_{\mathbf{s}} \xi_{\mathbf{s}'} \rangle_p = \langle \xi_{i_1,...,i_d} \xi_{j_1,...,j_d} \rangle_p = C \prod_{l=1}^{d} \delta_{i_l j_l} \binom{2M_l}{i_l}, \qquad (19)$$

where C is a positive constant and δ_{ij} is the Kroneker delta. The rationale underlying Eq. (19) is that it corresponds to consider the $\prod_{l=1}^{d} \binom{2M_l}{i_l}$ terms in $\xi_{i_1,..,i_d}$ as independently sampled from a zero-mean Gaussian.

A direct calculation, provided in Appendix C for completeness, gives the following result:

$$K(\boldsymbol{\theta}, \boldsymbol{\theta}') = C \prod_{l=1}^{d} \left[\cos\left(\frac{\theta_l - \theta_l'}{2} \right) \right]^{2M_l}. \qquad (20)$$

It is important to note that, while the kernel function in Eq. (20) enforces our prior knowledge about the mathematical structure of the VQE energy landscape, similarly to Eq. (16), it resolves the scalability issues for a high number of variational parameters.

3. Kernel design III

To enforce the symmetry condition that the unitary transformation $\hat{U}(\boldsymbol{\theta})$ has periodicity π instead of 2π, as in the examples presented in Sec. VI, let us introduce the translation of π on the lth dimension, defined as follows:

$$\mathcal{T}_l g(\theta_1, .., \theta_d) = g(\theta_1, .., \theta_l - \pi, ..., \theta_d), \qquad (21)$$

and the define the following "symmetrization operator":

$$\mathcal{S} = 2^{-\frac{d}{2}} \prod_{l=1}^{d} [1 + \mathcal{T}_l], \qquad (22)$$

which consists of a summation over all group elements with respect to which $\mathcal{E}(\boldsymbol{\theta})$ is invariant. We consider the following symmetrized form of the variational landscape:

$$\mathcal{E}(\boldsymbol{\theta}) = \mathcal{S} \sum_{i_1=0}^{2M_1} ... \sum_{i_d=0}^{2M_d} \xi_{i_1,...,i_d} \prod_{l=1}^{d} \cos\left(\frac{\theta_l}{2} \right)^{i_l} \sin\left(\frac{\theta_l}{2} \right)^{2M_l - i_l}, \qquad (23)$$

assuming that the Gaussian probability distribution encoded in Eq. (19) for the coefficients $\xi_{i_1,..,i_d}$.

A direct calculation, provided in Appendix D for completeness, gives the following result:

$$K(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \langle \mathcal{E}(\boldsymbol{\theta}) \mathcal{E}(\boldsymbol{\theta}') \rangle$$

$$= C \prod_{l=1}^{d} \left(\left[\cos\left(\frac{\theta_l - \theta_l'}{2} \right) \right]^{2M_l} + \left[\sin\left(\frac{\theta_l - \theta_l'}{2} \right) \right]^{2M_l} \right), \qquad (24)$$

which is also free of the scalability issues of the Kernel function in Eq. (17). In Sec. VI B we illustrate this applying the kernel function in Eq. (24) to a Heisenberg model, for a VQE with 27 variational parameters.

B. QPUs data

Let us consider any fixed Hamiltonian $\hat{H}$ whose energy landscape is a function $\mathcal{E}(\boldsymbol{\theta})$ sampled from our prior probability distribution $P[\mathcal{E}]$.

For each variational parameter $\boldsymbol{\theta}_\alpha$ we consider the random variable,

$$\mathcal{E}_\alpha = \frac{1}{N} \sum_{r=1}^{N} \mathcal{E}_{\alpha r}, \qquad (25)$$

describing the average of a series of N_{sh} quantum measurements (shots), given the variational state $|\phi(\boldsymbol{\theta}_\alpha)\rangle$ prepared on the quantum circuit. We assume that, for large N, the conditional probability of obtaining $\mathcal{E}_\alpha$ can be represented as follows:

$$P(\boldsymbol{\theta}_\alpha | \mathcal{E}_\alpha) \propto \exp\left\{ -\frac{1}{2\sigma_\alpha^2} (\mathcal{E}(\boldsymbol{\theta}_\alpha) - \mathcal{E}_\alpha)^2 \right\}, \qquad (26)$$

i.e., that $\mathcal{E}_\alpha$ is approximately gaussian distributed around the real underlying variational energy

$$\mathcal{E}(\boldsymbol{\theta}_\alpha) = \langle \phi(\boldsymbol{\theta}_\alpha) | \hat{H} | \phi(\boldsymbol{\theta}_\alpha) \rangle, \qquad (27)$$

with a variance σ_α.

We note that Eq. (26) is rigorously applicable primarily to ideal (fault-free) quantum machines. In fact, in such scenarios the outcome of quantum measurements is inherently probabilistic, and the variance σ_α^2 can be precisely defined as $\langle \phi(\boldsymbol{\theta}_\alpha) | \hat{H}^2 - \mathcal{E}^2(\boldsymbol{\theta}_\alpha) | \phi(\boldsymbol{\theta}_\alpha) \rangle / N$. In contrast, on real quantum devices σ_α must be regarded as an aggregate measure of the "error bar" for quantum measurements. This measure encompasses not only the intrinsic uncertainties of quantum projective measurements but also includes spurious effects like decoherence and hardware imperfections. Consequently, σ_α is effectively a characteristic of the specific quantum device and is thus treated as a "hyperparameter" within our GPR framework. This hyperparameter plays a crucial role in guiding the GPR model in determining how closely the posterior probability distribution for the variational landscape should align with the energy values measured by the quantum device.

It is important to highlight that this setting is typical in the practical application of GPR, where scenarios featuring an error bar that is intrinsically probabilistic and Gaussian are rarely, if ever, encountered. In real-world applications, GPR is commonly utilized in contexts where the error bar reflects a combination of inherent uncertainties and additional, device-specific noise factors.

In summary, we assume that the probability of obtaining a data set

$$D = \{(\boldsymbol{\theta}_\alpha, \mathcal{E}_\alpha, \sigma_\alpha) \, | \, \alpha = 1, .., n \}, \qquad (28)$$

given an underlying Hamiltonian $\hat{H}$ and a variational state $|\phi(\boldsymbol{\theta}_\alpha)\rangle$, can be estimated by the following equation:

$$P[D|\mathcal{E}] \propto \prod_{\alpha=1}^{n} P(\boldsymbol{\theta}_\alpha | \mathcal{E}_\alpha)$$

$$\propto \exp\left\{ -\sum_{\alpha=1}^{n} \frac{1}{2\sigma_\alpha^2} (\mathcal{E}(\boldsymbol{\theta}_\alpha) - \mathcal{E}_\alpha)^2 \right\}, \qquad (29)$$

where σ_α is a hyperparameter (that depends on the "quality" of the quantum device) for all $\boldsymbol{\theta}_\alpha$ such that the energy is evaluated from projective measurements.

C. The posterior probability distribution

Applying Bayes' rule with the given data D [refer to Eq. (28)] and a prior probability $P[\mathcal{E}]$, we obtain the following "posterior" conditional probability for the function $\mathcal{E}$:

$$P[\mathcal{E}|D] \propto P[\mathcal{E}] \exp\left\{-\sum_{\alpha=1}^{n} \frac{1}{2\sigma_\alpha^2}(\mathcal{E}(\boldsymbol{\theta}_\alpha) - \mathcal{E}_\alpha)^2\right\}, \quad (30)$$

which represents the probability distribution for the function $\mathcal{E}$, given a data set D, see Eq. (28).

The interpretation of Eq. (30) is that the probability of a specific energy landscape is influenced by the prior distribution and "anchored" to the data, i.e., it is exponentially suppressed for configurations deviating from the measured data $\mathcal{E}_\alpha$ by more than σ_α.

Since $P[\mathcal{E}|D]$ is Gaussian, we can calculate the following quantities exactly:

$$\langle\mathcal{E}(\boldsymbol{\theta})\rangle = \int \mathcal{D}[\mathcal{E}]P[\mathcal{E}|D]\,\mathcal{E}(\boldsymbol{\theta}), \quad (31)$$

$$\Sigma^2(\boldsymbol{\theta}) = \int \mathcal{D}[\mathcal{E}]P[\mathcal{E}|D]\left(\mathcal{E}^2(\boldsymbol{\theta}) - \langle\mathcal{E}(\boldsymbol{\theta})\rangle^2\right), \quad (32)$$

where Eq. (31) represents our prediction for $\mathcal{E}(\boldsymbol{\theta})$ at any test point $\boldsymbol{\theta}$ and Eq. (32) represents the uncertainty of our prediction.

Let us express explicitly $\langle\mathcal{E}(\boldsymbol{\theta})\rangle$ and $\Sigma(\boldsymbol{\theta})$ as a function of the data D [refer to Eq. (28)]. We define the matrix

$$\bar{K}_{\alpha\beta} = K(\boldsymbol{\theta}_\alpha, \boldsymbol{\theta}_\beta) + \sigma_\alpha^2\delta_{\alpha\beta} \quad \forall\,\alpha,\beta \in 1,..,n\,, \quad (33)$$

where K is the kernel function defined in Eq. (8). Following the procedure detailed in Appendices A and B for completeness, we have

$$\langle\mathcal{E}(\boldsymbol{\theta})\rangle = \sum_{\alpha,\beta=1}^{n} K(\boldsymbol{\theta}, \boldsymbol{\theta}_\alpha)[\bar{K}^{-1}]_{\alpha\beta}\mathcal{E}_\beta, \quad (34)$$

$$\Sigma^2(\boldsymbol{\theta}) = K(\boldsymbol{\theta}, \boldsymbol{\theta}) - \sum_{\alpha,\beta=1}^{n} K(\boldsymbol{\theta}, \boldsymbol{\theta}_\alpha)[\bar{K}^{-1}]_{\alpha\beta}K(\boldsymbol{\theta}_\beta, \boldsymbol{\theta})\,. \quad (35)$$

Note that evaluating Eqs. (34) and (35) requires computing the inverse of $\bar{K}$, whose size equals the number n of training data points. Therefore, for our approach to be practically applicable, it is important that n does not become prohibitively large. On the other hand, minimizing the variational energy does *not* necessitate learning the entire variational landscape [which is encoded in S parameters, see Eq. (6)]. In fact, as we are going to show in our benchmark calculations, the number n of training data point necessary for this task is generally smaller than S.

In the next section we describe a general method for calculating the VQE energy minima from QPU measurements, by combining the GPR approach framework discussed above with standard minimization algorithms.
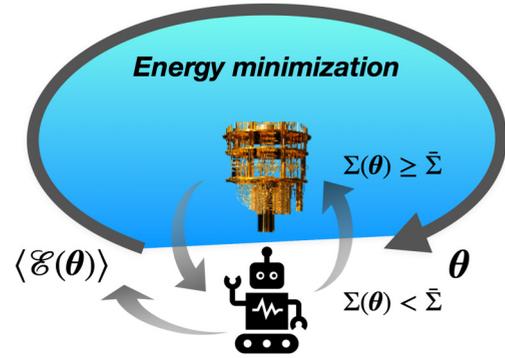

FIG. 1. Schematic illustration of the proposed energy minimization algorithm based on GPR. The method incorporates a GPR machine with a customized prior based on the specific VQE ansatz. The main steps of the algorithm are shown, including evaluation of uncertainty, utilization of GPR estimates when the uncertainty is below a threshold, and updating the GPR database when the uncertainty is above the threshold.

IV. ENERGY MINIMIZATION WITH ACTIVE LEARNING FRAMEWORK

The standard approach for computing the minimum of the variational energy $\mathcal{E}(\boldsymbol{\theta})$ involves evaluating the energy for a series of different variational parameters by averaging over a series of quantum measurements at each iteration. This method does not take advantage of any prior knowledge about the variational ansatz.

In contrast, we propose a modified procedure, as illustrated in Fig. 1, which incorporates the GPR method and a customized prior based on the specific variational ansatz implemented on the parametrized quantum circuit. This new algorithm can be summarized as follows. After initializing the posterior probability distribution starting from an empty dataset D, we perform the following steps:

(i) Whenever the minimization algorithm requests to evaluate the variational energy for a given set of variational parameters $\boldsymbol{\theta}$, compute the uncertainty $\Sigma(\boldsymbol{\theta})$ using GPR, based only on the information already available in D.

(ii) If $\Sigma(\boldsymbol{\theta}) \leqslant \bar{\Sigma}$, where $\bar{\Sigma}$ is a predefined accuracy threshold, proceed using the GPR estimate for the variational energy $\langle\mathcal{E}(\boldsymbol{\theta})\rangle$, without performing any additional quantum measurement.

(iii) If $\Sigma(\boldsymbol{\theta}) > \bar{\Sigma}$, estimate the variational energy using quantum measurements, and load this information into the GPR database D. Then, provide the obtained GPR estimate $\langle\mathcal{E}(\boldsymbol{\theta})\rangle$ to the minimization algorithm.

Note that our procedure, as outlined above, departs from the standard Bayesian optimization, which typically employs GPR uncertainty quantification to guide exploration in the variational space (a strategy generally feasible only for low-dimensional problems). Our approach, in contrast, relies on an independent energy minimization procedure, and uses GPR only to bypass QPU evaluations when possible. In this general sense, our algorithm's structure bears more resemblance to the adaptive scheme employed in building potential energy surfaces (PESs) using GPR, as seen in Ref. [33].

A crucial element of our framework is the custom kernel function, crafted to match the structure of the VQE circuit, thus improving GPR predictions by incorporating the specific form of the variational ansatz in the prior. Additionally, it is essential to point out that our approach does not aim to learn the full VQE energy landscape. Instead, it targets the trajectory of the energy-minimization process, which is essentially a one-dimensional path in the parameter space, from the start to the energy minimum. Therefore, our method effectively avoids the computational issues often associated with high-dimensional spaces, like the curse of dimensionality. In particular, as the minimization process advances, the explored points become increasingly close to each other, making our prior knowledge about the landscape's smoothness increasingly relevant, regardless of the number of variational parameters in the VQE ansatz.

In the next section, we illustrate in detail the methodology proposed above for calculations of a series of 2-qubit impurity-model Hamiltonians and present benchmark calculations performed on real quantum devices.

V. BENCHMARK CALCULATIONS OF A 2-QUBIT IMPURITY MODEL

We consider a Fermionic Hamiltonian represented as follows:

$$\hat{H} = \frac{U}{2}(\hat{n}_c - 1)^2 + \mathcal{D} \sum_{\sigma=\uparrow,\downarrow} (c_\sigma^\dagger d_\sigma + d_\sigma^\dagger c_\sigma) + \lambda^c \sum_{\sigma=\uparrow,\downarrow} d_\sigma d_\sigma^\dagger, \tag{36}$$

where $c_\sigma^\dagger$ and c_σ are the creation and annihilation operators of the so-called "impurity degrees of freedom", $d_\sigma^\dagger$ and d_σ are the creation and annihilation operators of the so-called "bath degrees of freedom", $\sigma \in \{\uparrow, \downarrow\}$ is the spin index, $\hat{n}_c = \sum_\sigma c_\sigma^\dagger c_\sigma$ is the impurity number operator, U is the Hubbard-repulsion parameter for the impurity degrees of freedom, while λ^c and $\mathcal{D}$ are the coupling constants characterizing the bath of the impurity model and its coupling to the impurity, respectively. From now on we set $\mathcal{D} = -|\mathcal{D}|$, with $|\mathcal{D}|$ serving as the unit of energy.

We emphasize that Eq. (36) represents the simplest possible "impurity model" or "embedding Hamiltonian" (EH), a fundamental building block of QE methods. Given that solving the EH in QE calculations is one of the most promising potential applications of VQE, our choice of this model for benchmark calculations aims to showcase the potential of our method in addressing more complex quantum systems typically encountered in practical QE calculations.

Specifically, we illustrate and benchmark our method for calculating the ground state $|\Phi(\mathcal{D}, \lambda^c, U)\rangle$ of $\hat{H}$, within the subspace generated by states with two electrons (i.e., half of the maximum possible occupation). The QPU data are generated using the IBM open-source framework for quantum computing Qiskit [29], which provides methods for manipulating quantum programs on real quantum computers, as well as on classical QPU simulators.

A. Qubit representation of the EH

Following the approach in Ref. [11], we transform the EH [see Eq. (36)] using the so-called "parity mapping" [34], as implemented in Qiskit [29], leading to the following 2-qubit representation:

$$\hat{H} = \zeta_0 \mathbb{1} + \zeta_1 (\sigma_1^z - \sigma_2^z) + \zeta_2 (\sigma_1^x + \sigma_2^x) + \zeta_3 \sigma_1^z \sigma_2^z$$
$$+ \zeta_4 (\sigma_1^x \sigma_2^z - \sigma_1^z \sigma_2^x) + \zeta_5 \sigma_1^x \sigma_2^x, \tag{37}$$

where the symbols σ_k^x, σ_k^y, and σ_k^z (with $k = 1, 2$) represent the Pauli matrices acting on the kth qubit, and the coefficients ζ_j are determined from the EH parameters U, $\mathcal{D}$, and λ^c through the parity mapping.

B. The VQE ansatz

To calculate the spin-singlet ground state of the EH we use an ansatz inspired by unitary coupled cluster ansatz with single and double excitations (UCCSD) [16,17,19,35]. We here express the wave function directly in qubit basis similar to previous studies [11,17] and expressed as products of unitary single and double rotations,

$$|\phi(\theta)\rangle = e^{\frac{i}{2}\theta^1 \sigma_2^y} e^{-\frac{i}{2}\theta^1 \sigma_1^y} e^{-\frac{i}{2}\theta^2 \sigma_1^y \sigma_2^x} e^{\frac{i}{2}\theta^2 \sigma_1^x \sigma_2^y} |\phi_0\rangle. \tag{38}$$

Here $\theta = (\theta^1, \theta^2)$, with each angle in the range of $[-\pi, \pi]$ and $|\phi_0\rangle$ is the spin-restricted Hartree-Fock ground-state solution of $\hat{H}$. The angles for the single-qubit rotations are set to be equal because of spin rotational symmetry.

Note that Eq. (38) is a special case of Eqs. (3) and (4), with $d = 2$ and $M_1 = M_2 = 2$. Therefore, from the framework described in Sec. II it follows that the variational energy $\mathcal{E}(\theta)$ can be expressed as a linear combination of 25 trigonometric functions of the angles,

$$\mathcal{E}(\boldsymbol{\theta}) = \sum_{k_1,k_2=-2}^{2} \epsilon_{k_1,k_2} \frac{e^{i\mathbf{k}\cdot\boldsymbol{\theta}}}{2\pi}, \tag{39}$$

and the GPR framework can be applied with the Kernel function,

$$K(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \frac{t^{-1}}{(2\pi)^2} \sum_{k_1,k_2=-2}^{2} e^{i\mathbf{k}\cdot(\boldsymbol{\theta}_1-\boldsymbol{\theta}_2)}. \tag{40}$$

The variational energy estimation $\langle \mathcal{E}(\boldsymbol{\theta}) \rangle$ and uncertainty quantification $\Sigma(\boldsymbol{\theta})$ are given by Eqs. (34) and (35), respectively.

C. Numerical tests on QPU simulators and real quantum devices

In this subsection, we evaluate the performance of our parametric GPR method for the impurity model described by Eqs. (36) and (37). We consider bath parameters $\lambda^c = 0, 1, 2$ and Hubbard interaction strengths $U = 1, 4, 8$, covering a wide range of interaction scenarios, from weakly interacting to strongly correlated regimes.

To demonstrate the effectiveness of our method, we used the simultaneous perturbation stochastic approximation (SPSA) [36], a standard optimization approach used for problems involving noisy function evaluations, comparing the results obtained from the bare method with the results augmented with an active learning strategy, here referred to as
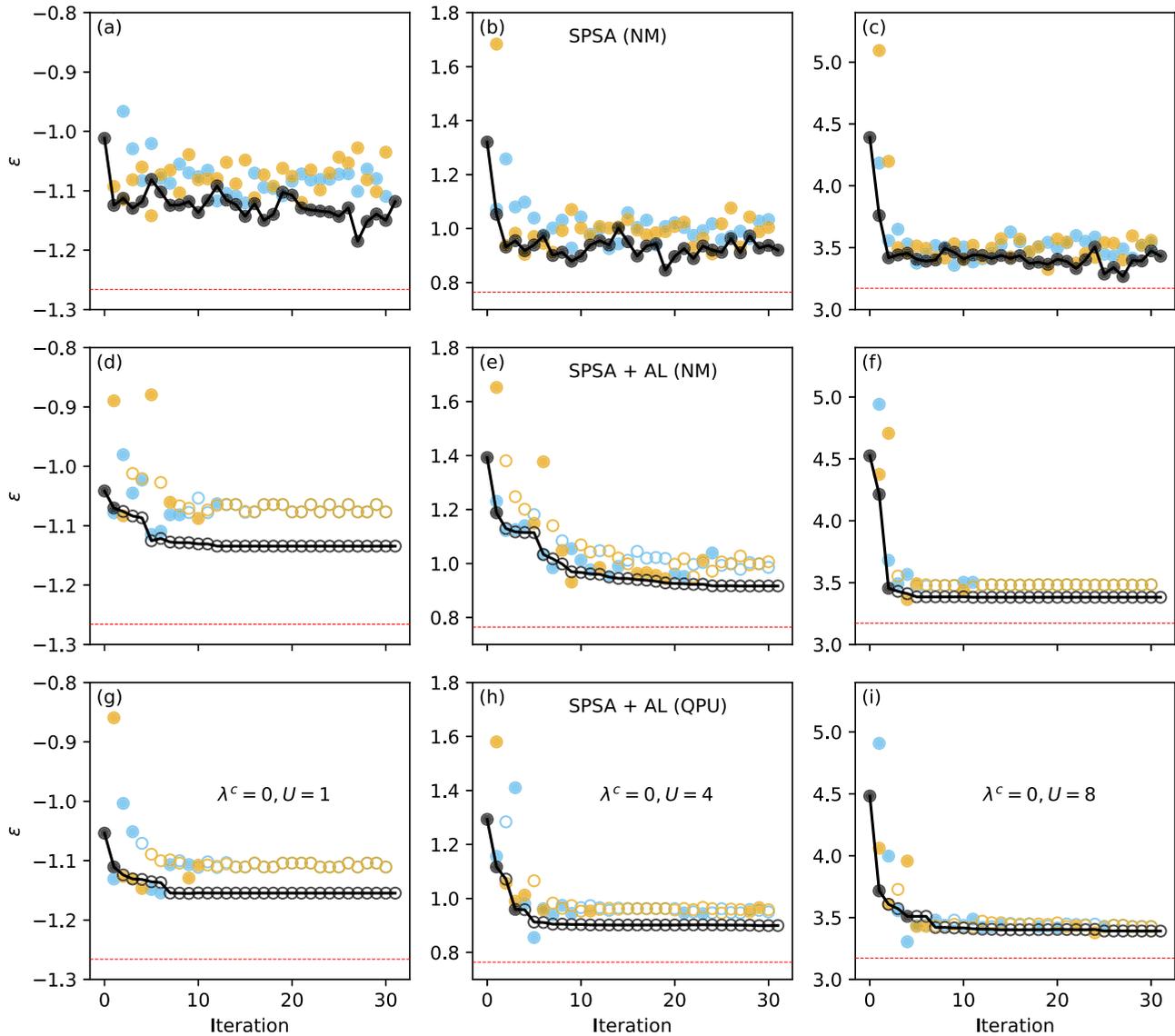
FIG. 2. Comparison of energy convergence for the impurity model as a function of iteration steps in the energy minimization procedure, using SPSA and SPSA+AL optimization methods on simulator with noise model (NM) derived from IBMQ_MUMBAI and IBMQ_MUMBAI QPU. Upper panels [(a)–(c)] show the results for SPSA (NM), middle panels [(d)–(f)] for SPSA+AL(NM), and lower panels [(g)–(i)] for SPSA+AL(QPU). For the impurity model, we fix $\lambda^c = 0$ and vary Hubbard interaction strength $U = 1$ (left panels), $U = 4$ (middle panels), and $U = 8$ (right panels), representing weakly interacting to strongly correlated regimes. The red-dashed horizontal lines indicate the exact state vector results. Full symbols indicate points obtained from direct energy measurement and added to GPR dataset D, while empty symbols represent points from GPR prediction without requiring additional measurement. Blue and orange circles indicate points used for gradient calculations, while black circles represents energy reached at each iteration. With 30 iterations, the number of times to measure the energy $\mathcal{E}(\boldsymbol{\theta})$ is 92 for SPSA, and falls in the range of [15,17] for SPSA+AL. We use $N_{\mathrm{sh}} = 2^{10}$ shots for the measurement of each term in the Hamiltonian (37).

SPSA+AL. In our SPSA+AL calculations, we set the following hyperparameters for the active learning framework, $\sigma_\alpha = \bar{\Sigma} = 0.005$ and $t = 1/(2\pi)^2$ such that $K(\boldsymbol{\theta}, \boldsymbol{\theta}) = K_c = 25$. We note that K_c is larger than the numerical estimation of it obtained by sampling a set of $N = 100$ random angles $\{\boldsymbol{\theta}_\alpha\}$ in the range of $[-\pi, \pi]$, $\frac{1}{N}\sum_{\alpha=1}^{N}[\mathcal{E}(\boldsymbol{\theta}_\alpha)]^2$, for the model at all the parameter points of λ^c and U, which falls in the range of (0.5, 15). We chose $\bar{\Sigma} = \sigma_\alpha$ to align the accuracy threshold for the GPR uncertainty, used in the active learning strategy, with the inherent limitations in the accuracy of the training data points. Although we focus on these specific

hyperparameter values for the paper, we have also tested other values, such as $\sigma_\alpha = \bar{\Sigma} = 0.05$ or $\sigma_\alpha = \bar{\Sigma}/2 = 0.005$, and found our results to be qualitatively consistent, indicating the robustness of our method. All points have been generated from either QPU simulators or real quantum devices and loaded within the GPR framework, following the active learning procedure described in Sec. IV.

Figure 2 illustrates the energy evolution as a function of the number of iteration steps throughout the energy minimization procedure, employing bare SPSA [panels (a), (b), and (c)] and SPSA+AL [panels (d), (e), and (f)] for $\lambda^c = 0$ and

TABLE I. Average and standard deviation of ground-state energies obtained using SPSA(NM) and SPSA+AL(NM) optimization methods of the impurity model over a sample of 20 runs each with different bath energy levels ($\lambda^c = 0, 1, 2$) and Hubbard interaction strengths ($U = 1, 4, 8$). Each row corresponds to a specific combination of U and λ^c. Columns present the average and standard deviation for SPSA(NM), and SPSA+AL(NM) results, with the last column displaying the exact state vector energies for comparison.

		SPSA(NM)		SPSA+AL(NM)		
λ^c	U	**AVG**	**STD**	**AVG**	**STD**	**EXACT**
0	1	–1.121	0.016	–1.134	0.009	–1.266
	4	0.921	0.046	0.917	0.016	0.764
	8	3.431	0.049	3.386	0.025	3.172
1	1	–2.295	0.019	–2.311	0.015	–2.454
	4	–0.164	0.041	–0.174	0.021	–0.323
	8	2.376	0.055	2.355	0.034	2.140
2	1	–3.782	0.029	–3.797	0.014	–3.968
	4	–1.420	0.050	–1.440	0.023	–1.604
	8	1.314	0.078	1.252	0.036	1.038

$U = 1, 4, 8$. Panels (g), (h), and (i) show the energy convergence of SPSA+AL for real quantum devices (QPU) across the range of interaction strengths, $U = 1, 4, 8$.

Our results show that, SPSA+AL exhibits rapid convergence compared with bare SPSA across all parameter regimes, necessitating a significantly smaller number of QPU energy evaluations. This holds true for both the data produced by the simulator and the data obtained from real quantum devices, demonstrating the ability of our method to adapt to the distinct noise characteristics of real hardware.

To systematically compare the performance of bare SPSA and SPSA+AL, we compiled the results in Table 1, which presents the average and standard deviation of ground-state energies obtained for 30 energy minimization steps of the impurity model with different Hubbard interaction strengths ($U = 1, 4, 8$) and bath energy levels ($\lambda^c = 0, 1, 2$). For both SPSA and SPSA+AL, the energy minimum evaluations were computed after 30 iterations, as the energy oscillated without further improvement beyond that point, as illustrated in Fig. 2. These results expand the analysis above in two significant ways. Firstly, by including 20 repetitions of the simulation, we demonstrate the robust performance of our SPSA+AL method. Secondly, by extending the analysis away from half-filling, we showcase the effectiveness of our approach for nonzero values of λ^c.

It should be noted that the data presented in Table 1 are derived exclusively from simulations, because of the limited availability of real quantum devices. However, the successful application of our method to real device data in the analysis of Fig. 2 suggests that the conclusions drawn from the simulated data remain valid and applicable to real quantum hardware.

The analysis presented in Table 1 highlights the superior performance of SPSA+AL compared to bare SPSA. For each iteration, SPSA queries two times for the value of $\mathcal{E}(\boldsymbol{\theta})$ to estimate the gradient, and one time for the energy at the current step, which allows to revert the change of variational angles if the updated energy shoots over a preset threshold

(0.5 for current calculations). Therefore, the total 30 SPSA iterations correspond to $30 \times 3 + 2 = 92$ times (including the initial and final energy measurement) of direct energy measurement for each run. In contrast, SPSA+AL requires only 15 to 27 times for energy measurement per run. Furthermore, SPSA+AL consistently delivers slightly more accurate energies for all values of U and λ^c considered, and the fidelities of the final variational states systematically exceed 99.7%. This demonstrates the effectiveness of our approach in enhancing the optimization process, reducing the number of QPU evaluations, and showcasing its robustness and applicability across different regimes. It is worth noting that the standard deviations observed for SPSA+AL method are systematically lower than the other methods, indicating reduced fluctuations compared to SPSA. This reduction in fluctuations could potentially improve the stability of QE methods using VQE as impurity solvers, as such methods require iterative impurity-model solutions with varying parameters, and output fluctuations can hamper stability and convergence.

VI. BENCHMARK CALCULATIONS OF HEISENBERG MODEL

To further demonstrate the effectiveness of SPSA+AL approach, we apply it to variational ground-state preparation of an L-site Heisenberg chain with Hamiltonian

$$\hat{\mathcal{H}} = J \sum_{i=1}^{L-1} \boldsymbol{\sigma}_i \cdot \boldsymbol{\sigma}_{i+1}. \tag{41}$$

Here $\boldsymbol{\sigma}_i$ is the vector of Pauli operators at site i, and we set $J = 1$ for antiferromagnetic coupling.

A. VQE ansatz with two parameters

The following Hamiltonian variational ansatz (HVA) is adopted for ground-state preparation [37],

$$|\phi(\boldsymbol{\theta})\rangle = e^{\frac{i}{2}\theta^1 H_1} e^{\frac{i}{2}\theta^2 H_2} |\phi_0\rangle, \tag{42}$$

with $H_1 = \sum_{i=1}^{L/2} \boldsymbol{\sigma}_{2i-1} \cdot \boldsymbol{\sigma}_{2i}$ and $H_2 = \sum_{i=1}^{L/2-1} \boldsymbol{\sigma}_{2i} \cdot \boldsymbol{\sigma}_{2i+1}$. The reference state $|\phi_0\rangle = \otimes_{i=1}^{L/2} \frac{1}{\sqrt{2}}(|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle)_{2i-1,2i}$ is a product state of Bell pairs (spin singlets), which is also the ground state of H_1. Each angle of $\boldsymbol{\theta}$ can be restricted to range of $[-\pi/2, \pi/2]$.

For the following numerical simulations, we set $L = 8$, corresponding to eight qubits. The ansatz of Eq. (42) corresponds to $d = 2$, $M_1 = 12$, and $M_2 = 9$ for generic expression of Eqs. (3) and (4). The variational energy $\mathcal{E}(\boldsymbol{\theta})$ can therefore be expressed as a linear combination of 25×19 trigonometric functions of the angles,

$$\mathcal{E}(\boldsymbol{\theta}) = \sum_{k_1=-12}^{12} \sum_{k_2=-9}^{9} \epsilon_{k_1,k_2} \frac{e^{i\mathbf{k}\cdot\boldsymbol{\theta}}}{2\pi}, \tag{43}$$

and the GPR framework can be applied with the Kernel function

$$K(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \frac{t^{-1}}{(2\pi)^2} \sum_{h_1=-6}^{6} \sum_{h_2=-4}^{4} e^{i2\mathbf{h}\cdot(\boldsymbol{\theta}_1-\boldsymbol{\theta}_2)}, \tag{44}$$
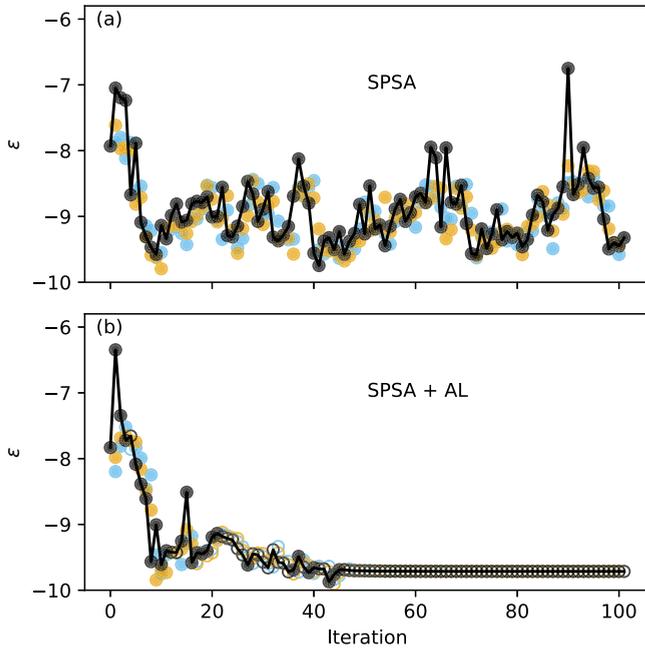
FIG. 3. Comparison of energy convergence for the Heisenberg model as a function of iteration steps in the energy minimization procedure, using SPSA and SPSA+AL optimization methods on simulator with noise model derived from IBMQ_MUMBAI QPU. Panel (a) shows the results with SPSA, while panel (b) shows that with SPSA+AL. We use the same color encoding as Fig. 2 for the data points: full symbols from direct energy measurement, while empty symbols from GPR prediction. Blue and orange circles indicate points used for gradient calculations, while black circle represents energy at each iteration. With 100 iterations, the number of times to measure the energy $\mathcal{E}(\boldsymbol{\theta})$ is 302 for SPSA, and 61 for SPSA+AL. The simulation is performed for an eight-site Heisenberg chain with antiferromagnetic coupling. We use $N_{\text{sh}} = 2^{10}$ shots for the measurement of each term in the Hamiltonian (41).

where we take into account that the period of $\mathcal{E}(\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$ is π rather than 2π. This originates from the special form of the generators of H_1 and H_2, where we have $\prod_{\mu \in \{x,y,z\}} \sigma_i^{\mu} \sigma_{i+1}^{\mu} \propto \mathbb{1}$, with σ^{μ} labeling the three components of Pauli matrices and $\mathbb{1}$ for the identity.

Numerical tests on QPU simulators

In Fig. 3, we present an energy-versus-iterations plot for the eight-site Heisenberg model using the SPSA and SPSA+AL optimization methods, illustrating a representative example of how a standard run proceeds. In line with our findings from the impurity model calculations, the SPSA+AL method exhibits significantly enhanced convergence efficiency, requiring on average only about 20% of the energy measurements compared to the bare SPSA method. To quantify this, we conducted an ensemble analysis involving 20 separate runs for each of the SPSA and SPSA+AL simulations. The results show that the averaged final energy for SPSA is -9.321 ± 0.311, while for SPSA+AL, it is notably lower at -9.701 ± 0.151, indicating a more precise convergence with a reduced variance. It is important to note, however, that the exact ground-state energy for this model

is -13.299. This significant discrepancy points to substantial biases in both simulation methods, consistent with the trends observed in the impurity model calculations.

While the primary objective of the AL method is not to directly address systematic errors, it plays a crucial role in ensuring consistency with the preestablished mathematical structure of the variational landscape. For the eight-site Heisenberg model studied here, it is feasible to obtain the exact ground state ψ_{G} by diagonalization on a classical computer. This allows us to calculate the fidelity of the converged solution, defined as $f = |\langle \phi(\boldsymbol{\theta})| \psi_{\text{G}}\rangle|^2$ at the optimized angles $\boldsymbol{\theta}$, which evaluates the efficacy of this indirect form of error mitigation. From the same sample set, we observe a fidelity of $f = 0.888 \pm 0.051$ for SPSA and a notably higher $f = 0.945 \pm 0.016$ for SPSA+AL, demonstrating enhanced fidelity and reduced variance. Remarkably, the obtained fidelity closely approaches the maximum achievable value of $f = 0.960$ as per the ansatz in Eq. (42). This suggests that, while our AL method primarily enforces consistency in the variational landscape, conventional error-mitigation techniques might still be beneficially applied post-SPSA+AL optimization to further diminish measurement biases, aligning with observations in existing literature [38].

We repeated these calculations also using the Kernel function defined in Eq. (24), obtaining -9.695 ± 0.091 for the energy and 0.943 ± 0.021 for the fidelity, requiring on average only about 13% of the energy measurements compared to the bare SPSA method. These results are consistent with those obtained using the Kernel function in Eq. (44). In the following section we also show SPSA+AL benchmark calculations of the Heisenberg model for a VQE with 27 variational parameters, leveraging that the kernel function in Eq. (24) it is free of scalability issues, as pointed out in Sec. III A.

B. Multilayer VQE ansatz with 27 parameters

To demonstrate the scalability of SPSA+AL to larger parameter spaces, we consider the following generalized N_l-layer HVA:

$$|\phi(\boldsymbol{\theta})\rangle = \Pi_{l=1}^{N_l} \left(\Pi_{j=1}^{L/2} e^{\frac{i}{2}\theta_{lj}^{(1)}\boldsymbol{\sigma}_{2i-1}\cdot\boldsymbol{\sigma}_{2i}}\right)\left(\Pi_{i=1}^{L/2-1} e^{\frac{i}{2}\theta_{li}^{(2)}\boldsymbol{\sigma}_{2i}\cdot\boldsymbol{\sigma}_{2i+1}}\right)|\phi_0\rangle. \tag{45}$$

Compared with ansatz of Eq. (42), we promote $\theta_{lj}^{(1)}, \theta_{lj}^{(2)}$ to be independent intra-layer variational parameters and adopt a multi-layer structure. The dimension of $\boldsymbol{\theta}$ is given by $N_l(L-1)$.

Numerical tests on QPU simulators

In Fig. 4, we present a representative energy-versus-iterations plot for the ten-site Heisenberg model with 27 variational parameters, using both SPSA and SPSA+AL with the Kernel function defined in Eq. (24).

We set a total number of 1000 iterations for the calculations to account for the relatively large number of parameters. Compared to the calculations with HVA of two parameters as shown in Fig. 3, about one order of magnitude more iterations are required for convergence.

Consistent with the previous benchmark calculations, the SPSA+AL method reduces quantum resources by
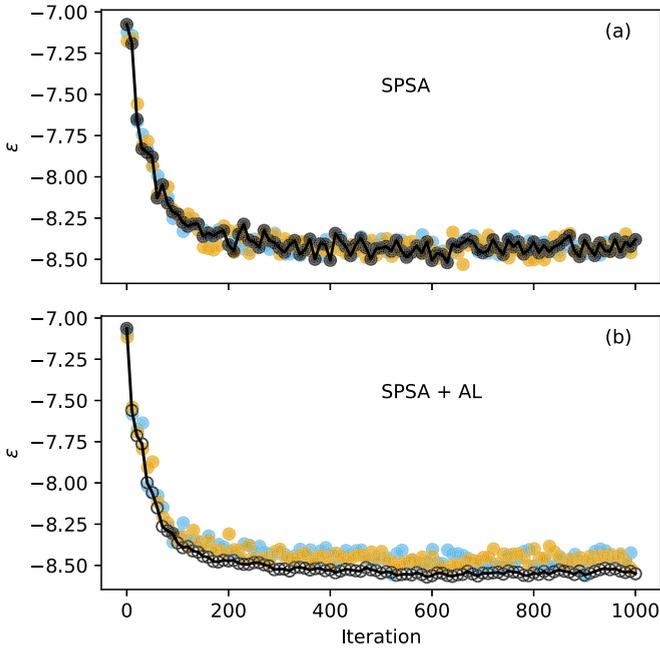
FIG. 4. Comparison of energy convergence for a Heisenberg model as a function of the iteration step in the energy minimization procedure with a three-layer HVA (27 parameters), using SPSA and SPSA+AL optimization methods on simulator with noise model derived from IBMQ_MUMBAI QPU. Panel (a) shows the results with SPSA, while panel (b) shows that with SPSA+AL. Full symbols indicate direct energy measurement, while empty symbols indicate GPR energy predictions. Blue and orange circles indicate points used for gradient calculations, while black circle represents the energy at each iteration. With 1000 iterations, the number of times to measure the energy $\mathcal{E}(\boldsymbol{\theta})$ is 3002 for SPSA, and 2005 for SPSA+AL. The simulation is performed for a ten-site Heisenberg chain with antiferromagnetic coupling. The number of shots used for evaluating each term in the Hamiltonian (41) is $N_{sh} = 2^{14}$. For clarity, only every tenth point is shown in the figure.

requiring about 2/3 of energy measurements in comparison with SPSA. We similarly conducted an ensemble analysis of 20 independent runs for each of the SPSA and SPSA+AL simulations. The results show that the averaged final energy for SPSA is -8.474 ± 0.052, while for SPSA+AL, it improves to -8.556 ± 0.024, indicating a better convergence with a reduced variance. Concerning fidelity, we observe $f = 0.958 \pm 0.008$ for SPSA and a slightly higher $f = 0.960 \pm 0.005$ for SPSA+AL. We note that the exact minimal energy for this ansatz is -17.031, which indicates substantial biases in both simulation methods, consistent with the trends observed in previous calculations.

VII. CONCLUSIONS

In this paper, we have presented a framework grounded on an active learning strategy for improving both accuracy and efficiency in quantum computations, with a focus on VQE. By employing a probabilistic machine learning based on GPR, our method leverages on our prior knowledge about the mathematical structure of the VQE ansatz. In particular, it accounts for exact cutoffs in the Fourier series of the variational

landscape, which quantitatively encode its smoothness and periodicity properties. A remarkable property of this framework is its compatibility with any VQE energy-minimization framework or error-mitigation method, enabling the integration of precise information about the variational landscape that is typically not utilized in standard approaches. We applied our active learning method in combination with SPSA, a standard optimization method, to a two-sites Anderson impurity model and to a eight-sites Heisenberg model, demonstrating its effectiveness across a range of parameters regimes, from weakly interacting to strongly correlated systems. Our results reveal that the SPSA+AL algorithm consistently outperforms the bare SPSA method, delivering more accurate and reliable ground-state energies, while requiring considerably fewer direct QPU energy evaluations. While these exploratory benchmarks are based on relatively simple systems and variational ansatzes, generalizations to more complex problems are plausible, and present promising directions for future research. Note also that we have, for fairness, compared performance utilizing the well-established SPSA algorithm with using identical settings. However, it is likely the GPR-based approach have the potential to open for other new algorithmic improvements. For example, efficiency and accuracy could be further improved by combining our active learning strategy with gradient-descent algorithms. This could be achieved by leveraging GPR for estimating not only the energy landscape but also its gradient, or by incorporating Bayesian optimization methods that utilize probabilistic models to optimize global functions under uncertainty. Exploring these research directions could lead to more efficient and accurate quantum computing methodologies, contributing to the development of advanced quantum matter simulation techniques. We anticipate that our framework will prove particularly valuable within the context of QE methods, enabling calculations with large impurities to describe dynamical-correlation effects beyond the capabilities of classical impurity solvers. In particular, we foresee promising applications to QE frameworks that require computing only the ground state of a finite Anderson impurity model, such as the recently developed "ghost Gutzwiller approximation" [39,40] and density matrix embedding theory [41]. Incorporating these QE frameworks with VQE has the potential to accelerate the advent of quantum advantage for simulations of large molecules and periodic materials, effectively capitalizing on the potential of near-term QPUs. In conclusion, our approach lays a solid foundation for future research and development in quantum error mitigation and optimization, potentially bringing us closer to realizing the full potential of quantum computing in the context of quantum matter simulation.

APPENDIX A: PATH INTEGRAL FORMULATION OF PARAMETRIC GPR

Our goal is to learn a real valued function $\mathcal{E}(\boldsymbol{\theta})$ ($\boldsymbol{\theta} \in \mathbb{R}^d$) from a finite set of training data points,

$$D = \{(\boldsymbol{\theta}_\alpha, \mathcal{E}_\alpha, \sigma_\alpha) \,|\, \alpha = 1, .., n\}, \qquad \text{(A1)}$$

where each $\mathcal{E}_\alpha$ is the outcome of the evaluation of $\mathcal{E}$ for the input parameter $\boldsymbol{\theta}_\alpha$, which is assumed to be sampled from a Gaussian distribution

$$P(\mathcal{E}_\alpha | \boldsymbol{\theta}_\alpha) \propto \exp\left\{-\frac{1}{2\sigma_\alpha^2}(\mathcal{E}(\boldsymbol{\theta}_\alpha) - \mathcal{E}_\alpha)^2\right\}, \qquad \text{(A2)}$$

i.e., the probability of evaluating $\mathcal{E}_1, .., \mathcal{E}_n$ (which are assumed to be independent) from a given underlying function $\mathcal{E}(\boldsymbol{\theta})$ is assumed to be

$$P[D|\mathcal{E}] \propto \prod_{\alpha=1}^{n} P(\boldsymbol{\theta}_\alpha | \mathcal{E}_\alpha)$$

$$\propto \exp\left\{-\sum_{\alpha=1}^{n} \frac{1}{2\sigma_\alpha^2}(\mathcal{E}(\boldsymbol{\theta}_\alpha) - \mathcal{E}_\alpha)^2\right\}, \qquad \text{(A3)}$$

Specifically, we aim to compute the so-called "posterior probability distribution" $P[\mathcal{E}|D]$, i.e., the probability that the function that we aim to learn is $\mathcal{E}(\boldsymbol{\theta})$, based on (i) the data D at our disposal and (ii) a Gaussian "prior probability distribution" $P[\mathcal{E}]$, encoding our prior knowledge before having any training data points.

Our first goal is to define precisely the concept of a probability distribution over a space of functions. Following the path integral procedure, this can be accomplished by first considering a discrete finite mesh with uniform spacing ϵ, over a d-dimensional rectangle R,

$$M_\epsilon = \{\boldsymbol{\theta}_1, .., \boldsymbol{\theta}_N\} \subset R \subset \mathbb{R}^d. \qquad \text{(A4)}$$

Over such discretized domain, probability measures can be rigorously represented as $p_\epsilon[\mathcal{E}]\mathcal{D}_\epsilon[\mathcal{E}]$, where

$$p_\epsilon[\mathcal{E}] = p_\epsilon[\mathcal{E}(\boldsymbol{\theta}_1), .., \mathcal{E}(\boldsymbol{\theta}_N)] \qquad \text{(A5)}$$

is a standard N-dimensional probability function, and

$$\mathcal{D}_\epsilon[\mathcal{E}] = \prod_{\boldsymbol{\theta} \in M_\epsilon} d\mathcal{E}(\boldsymbol{\theta}) \qquad \text{(A6)}$$

is the standard path integral measure.

1. The parametric prior

In our context of application, the prior probability distribution is designed to enforce the fact that $\mathcal{E}$ has to be of the following mathematical form:

$$\mathcal{E}(\boldsymbol{\theta}) = \sum_{s=1}^{S} \xi_{\mathbf{s}} T_{\mathbf{s}}(\boldsymbol{\theta}), \qquad \text{(A7)}$$

where $T_{\mathbf{s}} : R \subset \mathbb{R}^d \to \mathbb{R}$ are known functions, while the coefficients $\xi_{\mathbf{s}}$ are unknown. This information can be encoded in the following probability distribution,

$$P_\epsilon^\eta[\mathcal{E}] \propto \int \prod_{r=1}^{S} d\xi_r \, e^{-\frac{\epsilon}{2\eta^2} \sum_{\boldsymbol{\theta} \in M_\epsilon}(\mathcal{E}(\boldsymbol{\theta}) - \sum_{\mathbf{s}} \xi_{\mathbf{s}} T_{\mathbf{s}}(\boldsymbol{\theta}))^2}$$

$$\times \, e^{-\frac{1}{2} \sum_{\mathbf{ss}'} \mathcal{A}_{\mathbf{ss}'} \xi_{\mathbf{s}} \xi_{\mathbf{s}'}}, \qquad \text{(A8)}$$

where we have introduced a (strictly) positive-definite matrix $\mathcal{A}$, whose role is to encode the probability distribution for the coefficients $\xi_{\mathbf{s}}$, while also enforcing that the range of $\mathcal{E}$ is bounded, as we prove below. The parameter η will be considered in the limit as it approaches zero, i.e., in our formalism we take the limit $\eta \to 0$.

Let us prove that $P_\epsilon^\eta[\mathcal{E}]$ is a normalizable Gaussian probability distribution with zero mean for all finite values of η. By performing the Gaussian integral in Eq. (A8), we obtain that

$$P_\epsilon^\eta[\mathcal{E}] \propto e^{-\frac{1}{2}\frac{\epsilon}{\eta^2} \sum_{\boldsymbol{\theta},\boldsymbol{\theta}' \in M_\epsilon} \Pi_{\boldsymbol{\theta}\boldsymbol{\theta}'} \mathcal{E}(\boldsymbol{\theta})\mathcal{E}(\boldsymbol{\theta}')}, \qquad \text{(A9)}$$

where $\mathbb{1}$ is the $N \times N$ identity matrix and

$$\Pi = \mathbb{1} - T \frac{1}{\frac{\eta^2}{\epsilon}\mathcal{A} + T^\dagger T} T^\dagger, \qquad \text{(A10)}$$

$$T_{\boldsymbol{\theta}\mathbf{s}} = T_{\mathbf{s}}(\boldsymbol{\theta}) \,\forall\, \mathbf{s} = 1, .., S, \, \boldsymbol{\theta} \in M_\epsilon. \qquad \text{(A11)}$$

Note that Π is positive definite. In fact, since $\frac{\eta^2}{\epsilon}\mathcal{A}$ is positive definite, we have

$$\mathbb{1} - T \frac{1}{\frac{\eta^2}{\epsilon}\mathcal{A} + T^\dagger T} T^\dagger > \mathbb{1} - T \frac{1}{T^\dagger T} T^\dagger, \qquad \text{(A12)}$$

where the right-hand side is an orthogonal projector, as one can readily verify by inspection calculating its square. It follows that Eq. (A9) represents a normalizable zero-mean Gaussian distribution.

2. Posterior probability distribution

Let us assume to have a series of data D [see Eq. (A1)], where $\boldsymbol{\theta}_\alpha \in M_\epsilon \,\forall\, \alpha = 1, .., n$. As explained in the main text [see before Eq. (20)], from Bayes' theorem it follows that the posterior conditional probability distribution for the function $\mathcal{E}$ is the following:

$$P_\epsilon^\eta[\mathcal{E}|D] \propto P_\epsilon^\eta[\mathcal{E}] e^{-\sum_{\alpha=1}^{n} \frac{1}{2\sigma_\alpha^2}(\mathcal{E}(\boldsymbol{\theta}_\alpha) - \mathcal{E}_\alpha)^2}, \qquad \text{(A13)}$$

which represents the probability distribution for the function $\mathcal{E}$, given the data set D and the prior $P_\epsilon^\eta[\mathcal{E}]$ [see Eqs. (A3) and (A8)].

3. Probabilistic predictions at a test point

We are interested in calculating quantities of the following form:

$$\langle \mathcal{E}^l(\boldsymbol{\theta}) \rangle = \int \mathcal{D}_\epsilon[\mathcal{E}] P_\epsilon^\eta[\mathcal{E}|D] \, (\mathcal{E}(\boldsymbol{\theta}))^l$$

$$= \int \mathcal{D}_\epsilon[\mathcal{E}] P_\epsilon^\eta[\mathcal{E}] \, e^{-\sum_{\alpha=1}^n \frac{1}{2\sigma_\alpha^2}(\mathcal{E}(\boldsymbol{\theta}_\alpha)-\mathcal{E}_\alpha)^2} \mathcal{E}^l(\boldsymbol{\theta}), \quad \text{(A14)}$$

where $l \in \mathbb{N}$ and $\boldsymbol{\theta} \in M_\epsilon$ (which is assumed to be different from all of the $\boldsymbol{\theta}_\alpha$ in the training data set) is a so-called "test point," i.e., a point where we want to evaluate the probability distribution for $\mathcal{E}(\boldsymbol{\theta})$, based on our posterior probability distribution.

Equation (A14) can be conveniently rewritten by integrating out all variables except $\mathcal{E}(\boldsymbol{\theta}_1), .., \mathcal{E}(\boldsymbol{\theta}_\alpha)$, and $\mathcal{E}(\boldsymbol{\theta})$. From standard Gaussian identities, it follows that this gives the following expression:

$$\langle \mathcal{E}^l(\boldsymbol{\theta}) \rangle = \frac{\int \left[\prod_{\alpha=1}^n d\mathcal{E}(\boldsymbol{\theta}_\alpha) \right] d\mathcal{E}(\boldsymbol{\theta}) \, e^{-S_\epsilon^\eta - U} \, (\mathcal{E}(\boldsymbol{\theta}))^l}{\int \left[\prod_{\alpha=1}^n d\mathcal{E}(\boldsymbol{\theta}_\alpha) \right] d\mathcal{E}(\boldsymbol{\theta}) \, e^{-S_\epsilon^\eta - U}}, \quad \text{(A15)}$$

where:

$$U = \sum_{\alpha=1}^n \frac{1}{2\sigma_\alpha^2} (\mathcal{E}(\boldsymbol{\theta}_\alpha) - \mathcal{E}_\alpha)^2 \quad \text{(A16)}$$

and:

$$\begin{aligned} S_\epsilon^\eta &= \frac{1}{2} \sum_{\alpha,\beta=1}^n \left[\bar{\mathcal{K}}_\epsilon^\eta\right]_{\alpha,\beta}^{-1} \mathcal{E}(\boldsymbol{\theta}_\alpha)\mathcal{E}(\boldsymbol{\theta}_\beta) \\ &\quad + \frac{1}{2}\left[\bar{\mathcal{K}}_\epsilon^\eta\right]_{n+1,n+1}^{-1} \mathcal{E}(\boldsymbol{\theta})\mathcal{E}(\boldsymbol{\theta}) \\ &\quad + \frac{1}{2} \sum_{\alpha=1}^n \left[\bar{\mathcal{K}}_\epsilon^\eta\right]_{\alpha,n+1}^{-1} \mathcal{E}(\boldsymbol{\theta}_\alpha)\mathcal{E}(\boldsymbol{\theta}) \\ &\quad + \frac{1}{2} \sum_{\beta=1}^n \left[\bar{\mathcal{K}}_\epsilon^\eta\right]_{n+1,\beta}^{-1} \mathcal{E}(\boldsymbol{\theta})\mathcal{E}(\boldsymbol{\theta}_\beta), \quad \text{(A17)} \end{aligned}$$

where $[\bar{\mathcal{K}}_\epsilon^\eta]$ is the $(n+1) \times (n+1)$ matrix with entries

$$\left[\bar{\mathcal{K}}_\epsilon^\eta\right]_{\alpha,\beta} = K_\epsilon^\eta(\boldsymbol{\theta}_\alpha,\boldsymbol{\theta}_\beta) \quad \forall \alpha, \beta \in 1, .., n, \quad \text{(A18)}$$

$$\left[\bar{\mathcal{K}}_\epsilon^\eta\right]_{\alpha,n+1} = K_\epsilon^\eta(\boldsymbol{\theta}_\alpha,\boldsymbol{\theta}) \quad \forall \alpha \in 1, .., n, \quad \text{(A19)}$$

$$\left[\bar{\mathcal{K}}_\epsilon^\eta\right]_{n+1,\beta} = K_\epsilon^\eta(\boldsymbol{\theta},\boldsymbol{\theta}_\beta) \quad \forall \beta \in 1, .., n, \quad \text{(A20)}$$

$$\left[\bar{\mathcal{K}}_\epsilon^\eta\right]_{n+1,n+1} = K_\epsilon^\eta(\boldsymbol{\theta},\boldsymbol{\theta}), \quad \text{(A21)}$$

and

$$K_\epsilon^\eta(\boldsymbol{\theta},\boldsymbol{\theta}') = \int \mathcal{D}_\epsilon[\mathcal{E}] P_\epsilon^\eta[\mathcal{E}] \, \mathcal{E}(\boldsymbol{\theta})\mathcal{E}(\boldsymbol{\theta}') \quad \forall \boldsymbol{\theta}, \boldsymbol{\theta}' \in M_\epsilon \quad \text{(A22)}$$

is the so-called "kernel function" of the prior distribution P_ϵ^η.

As discussed in the main text, we are specifically interested in calculating

$$\bar{\mathcal{E}}_\epsilon^\eta(\boldsymbol{\theta}) = \int \mathcal{D}_\epsilon[\mathcal{E}] P_\epsilon^\eta[\mathcal{E}|D] \, \mathcal{E}(\boldsymbol{\theta}), \quad \text{(A23)}$$

$$\left(\Sigma_\epsilon^\eta(\boldsymbol{\theta})\right)^2 = \int \mathcal{D}_\epsilon[\mathcal{E}] P_\epsilon^\eta[\mathcal{E}|D] \, (\mathcal{E}^2(\boldsymbol{\theta}) - \langle \mathcal{E}(\boldsymbol{\theta}) \rangle^2), \quad \text{(A24)}$$

where Eq. (A23) represents our prediction for $\mathcal{E}(\boldsymbol{\theta})$ at any test point $\boldsymbol{\theta}$ and Eq. (A24) represents the uncertainty of our prediction. These quantities can be conveniently evaluated by computing first the "partition function",

$$Z_\epsilon^\eta(\lambda) := \int \left[\prod_{\alpha=1}^n d\mathcal{E}(\boldsymbol{\theta}_\alpha) \right] d\mathcal{E}(\boldsymbol{\theta}) \, e^{-S_\epsilon^\eta - U + \lambda \mathcal{E}(\boldsymbol{\theta})} \quad \text{(A25)}$$

and subsequently using the following identities:

$$\bar{\mathcal{E}}_\epsilon^\eta(\boldsymbol{\theta}) = \partial_\lambda \ln\left(Z_\epsilon^\eta(\lambda)\right), \quad \text{(A26)}$$

$$\left(\Sigma_\epsilon^\eta(\boldsymbol{\theta})\right)^2 = \partial_\lambda^2 \ln\left(Z_\epsilon^\eta(\lambda)\right). \quad \text{(A27)}$$

A direct calculation shows that

$$\bar{\mathcal{E}}_\epsilon^\eta(\boldsymbol{\theta}) = \sum_{\alpha,\beta=1}^n K_\epsilon^\eta(\boldsymbol{\theta},\boldsymbol{\theta}_\alpha)\left[\bar{K}_\epsilon^\eta\right]_{\alpha\beta}^{-1} \mathcal{E}_\beta, \quad \text{(A28)}$$

$$\left(\Sigma_\epsilon^\eta(\boldsymbol{\theta})\right)^2 = K_\epsilon^\eta(\boldsymbol{\theta},\boldsymbol{\theta}) - \sum_{\alpha,\beta=1}^n K_\epsilon^\eta(\boldsymbol{\theta},\boldsymbol{\theta}_\alpha)\left[\bar{K}_\epsilon^\eta\right]_{\alpha\beta}^{-1} K_\epsilon^\eta(\boldsymbol{\theta}_\beta,\boldsymbol{\theta}), \quad \text{(A29)}$$

where $\bar{K}_\epsilon^\eta$ is the $n \times n$ matrix with entries

$$\left[\bar{K}_\epsilon^\eta\right]_{\alpha\beta} = K_\epsilon^\eta(\boldsymbol{\theta}_\alpha,\boldsymbol{\theta}_\beta) + \sigma_\alpha^2 \delta_{\alpha\beta} \quad \forall \alpha, \beta \in 1, .., n. \quad \text{(A30)}$$

4. Calculation of the Kernel function

As shown in the previous section, the GPR estimate of our prediction for $\mathcal{E}(\boldsymbol{\theta})$ and the corresponding uncertainty [see Eqs. (A28) and (A29), respectively] depend explicitly on η and ϵ through the Kernel function,

$$K_\epsilon^\eta(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2) = \int \mathcal{D}_\epsilon[\mathcal{E}] P_\epsilon^\eta[\mathcal{E}] \, \mathcal{E}(\boldsymbol{\theta}_1)\mathcal{E}(\boldsymbol{\theta}_2), \quad \text{(A31)}$$

which is defined $\forall \boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in M_\epsilon$.

Since we aim to enforce Eq. (A7) *exactly*, we need to evaluate Eq. (A31) for $\eta \to 0$. In this limit we obtain

$$K(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2) = \lim_{\epsilon\to 0}\lim_{\eta\to 0} K_\epsilon^\eta(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2) \propto \lim_{\epsilon\to 0}\lim_{\eta\to 0} \int \mathcal{D}_\epsilon[\mathcal{E}] \int \prod_{r=1}^S d\xi_r \, e^{-\frac{\epsilon}{2\eta^2}\sum_{\boldsymbol{\theta}\in M_\epsilon}(\mathcal{E}(\boldsymbol{\theta})-\sum_{\mathbf{s}}\xi_{\mathbf{s}}T_{\mathbf{s}}(\boldsymbol{\theta}))^2} e^{-\frac{1}{2}\sum_{ss'}\mathcal{A}_{\mathbf{ss'}}\xi_{\mathbf{s}}\xi_{\mathbf{s'}}} \, \mathcal{E}(\boldsymbol{\theta}_1)\mathcal{E}(\boldsymbol{\theta}_2)$$

$$\propto \lim_{\epsilon\to 0} \int \mathcal{D}_\epsilon[\mathcal{E}] \int \prod_{r=1}^S d\xi_r \, \delta\left(\mathcal{E}(\boldsymbol{\theta}) - \sum_{\mathbf{s}}\xi_{\mathbf{s}}T_{\mathbf{s}}(\boldsymbol{\theta})\right) e^{-\frac{1}{2}\sum_{ss'}\mathcal{A}_{\mathbf{ss'}}\xi_{\mathbf{s}}\xi_{\mathbf{s'}}} \, \mathcal{E}(\boldsymbol{\theta}_1)\mathcal{E}(\boldsymbol{\theta}_2)$$

$$= \int \prod_{r=1}^{S} d\xi_r \, e^{-\frac{1}{2} \sum_{ss'} \mathcal{A}_{ss'} \xi_s \xi_{s'}} \left(\sum_{s_1=1}^{S} \xi_{s_1} T_{s_1}(\boldsymbol{\theta}_1) \right) \left(\sum_{s_2=1}^{S} \xi_{s_2} T_{s_2}(\boldsymbol{\theta}_2) \right)$$

$$= \sum_{s_1, s_2=1}^{S} T_{s_1}(\boldsymbol{\theta}_1) T_{s_2}(\boldsymbol{\theta}_2) \int \prod_{r=1}^{S} d\xi_r \, e^{-\frac{1}{2} \sum_{s,s'=1}^{S} \mathcal{A}_{ss'} \xi_s \xi_{s'}} \xi_{s_1} \xi_{s_2} = \sum_{s_1, s_2=1}^{S} \Delta_{s_1 s_2} T_{s_1}(\boldsymbol{\theta}_1) T_{s_2}(\boldsymbol{\theta}_2) \,, \tag{A32}$$

where

$$\Delta = \mathcal{A}^{-1} \,. \tag{A33}$$

The final step is to compute the Kernel function in the continuum limit $\epsilon \to 0$, which is given by the following equation:

$$K(\boldsymbol{\theta}, \boldsymbol{\theta}') = \sum_{s,s'=1}^{S} \Delta_{ss'} \, T_s(\boldsymbol{\theta}) T_{s'}(\boldsymbol{\theta}) \,. \tag{A34}$$

APPENDIX B: SUMMARY OF FINAL EQUATIONS

In summary, we derived the following equations, previously introduced in the main text:

$$\bar{\mathcal{E}}(\boldsymbol{\theta}) = \sum_{\alpha, \beta=1}^{n} K(\boldsymbol{\theta}, \boldsymbol{\theta}_\alpha) [\bar{K}]_{\alpha\beta}^{-1} \mathcal{E}_\beta \,, \tag{B1}$$

$$(\Sigma(\boldsymbol{\theta}))^2 = K(\boldsymbol{\theta}, \boldsymbol{\theta}) - \sum_{\alpha, \beta=1}^{n} K(\boldsymbol{\theta}, \boldsymbol{\theta}_\alpha) \bar{K}_{\alpha\beta}^{-1} K(\boldsymbol{\theta}_\beta, \boldsymbol{\theta}) \,, \tag{B2}$$

where $\bar{K}$ is the $n \times n$ matrix with entries

$$[\bar{K}]_{\alpha\beta} = K(\boldsymbol{\theta}_\alpha, \boldsymbol{\theta}_\beta) + \sigma_\alpha^2 \delta_{\alpha\beta} \quad \forall \alpha, \beta \in 1, .., n \,. \tag{B3}$$

The Kernel function is given by the following equation:

$$K(\boldsymbol{\theta}, \boldsymbol{\theta}') = \sum_{s,s'=1}^{S} \Delta_{ss'} \, T_s(\boldsymbol{\theta}) T_{s'}(\boldsymbol{\theta}') \,, \tag{B4}$$

where

$$\Delta_{ss'} = \langle \xi_s \xi_{s'} \rangle \,, \tag{B5}$$

and the expectation value fully encodes the Gaussian prior probability distribution for the coefficients ξ_s.

APPENDIX C: CALCULATION OF KERNEL EQ. (20) OF THE MAIN TEXT

Let us consider the equation for the VQE landscape introduced in Eq. (6) of the main text,

$$\mathcal{E}(\boldsymbol{\theta}) = \mathcal{S} \sum_{i_1=0}^{2M_1} \cdots \sum_{i_d=0}^{2M_d} \xi_{i_1, \ldots, i_d} \prod_{l=1}^{d} \cos\left(\frac{\theta_l}{2}\right)^{i_l} \sin\left(\frac{\theta_l}{2}\right)^{2M_l - i_l} \,. \tag{C1}$$

Let us now calculate the Kernel function based on the following identity introduced in the main text:

$$\langle \xi_s \xi_{s'} \rangle_p = \langle \xi_{i_1, \ldots, i_d} \xi_{j_1, \ldots, j_d} \rangle_p = C \prod_{l=1}^{d} \delta_{i_l j_l} \binom{2M_l}{i_l} \,, \tag{C2}$$

where C is a positive constant

$$K(\boldsymbol{\theta}, \boldsymbol{\theta}') = \langle \mathcal{E}(\boldsymbol{\theta}) \mathcal{E}(\boldsymbol{\theta}') \rangle = \sum_{ss'} \langle \xi_s \xi_{s'} \rangle_p T_s(\boldsymbol{\theta}) T_{s'}(\boldsymbol{\theta}') = C \sum_{i_1=0}^{2M_1} \cdots \sum_{i_d=0}^{2M_d} \prod_{l=1}^{d} \binom{2M_l}{i_l} \cos\left(\frac{\theta_l}{2}\right)^{i_l} \cos\left(\frac{\theta_l'}{2}\right)^{i_l} \sin\left(\frac{\theta_l}{2}\right)^{2M_l - i_l} \sin\left(\frac{\theta_l'}{2}\right)^{2M_l - i_l}$$

$$= C \prod_{l=1}^{d} \left[\cos\left(\frac{\theta_l}{2}\right) \cos\left(\frac{\theta_l'}{2}\right) + \sin\left(\frac{\theta_l}{2}\right) \sin\left(\frac{\theta_l'}{2}\right) \right]^{2M_l} = C \prod_{l=1}^{d} \left[\cos\left(\frac{\theta_l - \theta_l'}{2}\right) \right]^{2M_l} \,, \tag{C3}$$

which is the result previously stated in Eq. (19) of the main text.

APPENDIX D: CALCULATION OF KERNEL IN EQ. (24) OF THE MAIN TEXT

Let us consider the symmetrized equation for the VQE landscape introduced in Eq. (23) of the main text,

$$\mathcal{E}(\boldsymbol{\theta}) = \mathcal{S} \sum_{i_1=0}^{2M_1} \cdots \sum_{i_d=0}^{2M_d} \xi_{i_1, \ldots, i_d} \prod_{l=1}^{d} \cos\left(\frac{\theta_l}{2}\right)^{i_l} \sin\left(\frac{\theta_l}{2}\right)^{2M_l - i_l} \,, \tag{D1}$$

where

$$\mathcal{S} = 2^{-\frac{d}{2}} \prod_{l=1}^{d} [1 + \mathcal{T}_l] \,. \tag{D2}$$

and $\mathcal{T}_l$ represents a translation of π on the lth dimension, defined by

$$\mathcal{T}_l g(\theta_1, .., \theta_d) = g(\theta_1, .., \theta_l - \pi, ..., \theta_d) \,. \tag{D3}$$

Let us now calculate the Kernel function based on the following identity introduced in Eq. (19) of the main text,

$$\langle \xi_s \xi_{s'} \rangle_p = \langle \xi_{i_1, \ldots, i_d} \xi_{j_1, \ldots, j_d} \rangle_p = C \prod_{l=1}^{d} \delta_{i_l j_l} \binom{2M_l}{i_l} \,, \tag{D4}$$

where C is a positive constant:

$$
\begin{aligned}
K(\boldsymbol{\theta}, \boldsymbol{\theta}') &= \langle \mathcal{E}(\boldsymbol{\theta})\mathcal{E}(\boldsymbol{\theta}') \rangle = \sum_{\mathbf{ss}'} \langle \xi_\mathbf{s} \xi_{\mathbf{s}'} \rangle_p T_\mathbf{s}(\boldsymbol{\theta}) T_{\mathbf{s}'}(\boldsymbol{\theta}') = C\, 2^{-d} \sum_{i_1=0}^{2M_1} \cdots \sum_{i_d=0}^{2M_d} \prod_{l=1}^d \binom{2M_l}{i_l} \\
&\quad \times \left(\cos\left(\frac{\theta_l}{2}\right)^{i_l} \sin\left(\frac{\theta_l}{2}\right)^{2M_l-i_l} + \left[-\sin\left(\frac{\theta_l}{2}\right)\right]^{i_l} \left[\cos\left(\frac{\theta_l}{2}\right)\right]^{2M_l-i_l} \right) \\
&\quad \times \left(\cos\left(\frac{\theta_l'}{2}\right)^{i_l} \sin\left(\frac{\theta_l'}{2}\right)^{2M_l-i_l} + \left[-\sin\left(\frac{\theta_l'}{2}\right)\right]^{i_l} \left[\cos\left(\frac{\theta_l'}{2}\right)\right]^{2M_l-i_l} \right) \\
&= C\, 2^{-d} \prod_{l=1}^d \sum_{i_l=0}^{2M_l} \binom{2M_l}{i_l} \left(\cos\left(\frac{\theta_l}{2}\right)^{i_l} \sin\left(\frac{\theta_l}{2}\right)^{2M_l-i_l} + \left[-\sin\left(\frac{\theta_l}{2}\right)\right]^{i_l} \left[\cos\left(\frac{\theta_l}{2}\right)\right]^{2M_l-i_l} \right) \\
&\quad \times \left(\cos\left(\frac{\theta_l'}{2}\right)^{i_l} \sin\left(\frac{\theta_l'}{2}\right)^{2M_l-i_l} + \left[-\sin\left(\frac{\theta_l'}{2}\right)\right]^{i_l} \left[\cos\left(\frac{\theta_l'}{2}\right)\right]^{2M_l-i_l} \right) \\
&= C\, 2^{-d} \prod_{l=1}^d \left\{ \sum_{i_l=0}^{2M_l} \binom{2M_l}{i_l} \left[\cos\left(\frac{\theta_l}{2}\right)\cos\left(\frac{\theta_l'}{2}\right)\right]^{i_l} \left[\sin\left(\frac{\theta_l}{2}\right)\sin\left(\frac{\theta_l'}{2}\right)\right]^{2M_l-i_l} \right. \\
&\quad + \sum_{i_l=0}^{2M_l} \binom{2M_l}{i_l} \left[\sin\left(\frac{\theta_l}{2}\right)\sin\left(\frac{\theta_l'}{2}\right)\right]^{i_l} \left[\cos\left(\frac{\theta_l}{2}\right)\cos\left(\frac{\theta_l'}{2}\right)\right]^{2M_l-i_l} \\
&\quad + \sum_{i_l=0}^{2M_l} \binom{2M_l}{i_l} \left[-\cos\left(\frac{\theta_l}{2}\right)\sin\left(\frac{\theta_l'}{2}\right)\right]^{i_l} \left[\sin\left(\frac{\theta_l}{2}\right)\cos\left(\frac{\theta_l'}{2}\right)\right]^{2M_l-i_l} \\
&\quad \left. + \sum_{i_l=0}^{2M_l} \binom{2M_l}{i_l} \left[-\sin\left(\frac{\theta_l}{2}\right)\cos\left(\frac{\theta_l'}{2}\right)\right]^{i_l} \left[\cos\left(\frac{\theta_l}{2}\right)\sin\left(\frac{\theta_l'}{2}\right)\right]^{2M_l-i_l} \right\} \\
&= C \prod_{l=1}^d \left\{ \sum_{i_l=0}^{2M_l} \binom{2M_l}{i_l} \left[\cos\left(\frac{\theta_l}{2}\right)\cos\left(\frac{\theta_l'}{2}\right)\right]^{i_l} \left[\sin\left(\frac{\theta_l}{2}\right)\sin\left(\frac{\theta_l'}{2}\right)\right]^{2M_l-i_l} \right. \\
&\quad \left. + \sum_{i_l=0}^{2M_l} \binom{2M_l}{i_l} \left[-\cos\left(\frac{\theta_l}{2}\right)\sin\left(\frac{\theta_l'}{2}\right)\right]^{i_l} \left[\sin\left(\frac{\theta_l}{2}\right)\cos\left(\frac{\theta_l'}{2}\right)\right]^{2M_l-i_l} \right\} \\
&= C \prod_{l=1}^d \left\{ \left[\cos\left(\frac{\theta_l}{2}\right)\cos\left(\frac{\theta_l'}{2}\right) + \sin\left(\frac{\theta_l}{2}\right)\sin\left(\frac{\theta_l'}{2}\right)\right]^{2M_l} + \left[\cos\left(\frac{\theta_l}{2}\right)\sin\left(\frac{\theta_l'}{2}\right) - \sin\left(\frac{\theta_l}{2}\right)\cos\left(\frac{\theta_l'}{2}\right)\right]^{2M_l} \right\} \\
&= C \prod_{l=1}^d \left(\left[\cos\left(\frac{\theta_l - \theta_l'}{2}\right)\right]^{2M_l} + \left[\sin\left(\frac{\theta_l - \theta_l'}{2}\right)\right]^{2M_l} \right),
\end{aligned}
\tag{D5}
$$

which is the result previously stated in Eq. (24) of the main text.

---

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

[2] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, Simulated quantum computation of molecular energies, Science **309**, 1704 (2005).

[3] J. D. Whitfield, J. Biamonte, and A. Aspuru-Guzik, Simulation of electronic structure Hamiltonians using quantum computers, Mol. Phys. **109**, 735 (2011).

[4] D. Wecker, B. Bauer, B. K. Clark, M. B. Hastings, and M. Troyer, Gate-count estimates for performing quantum chemistry on small quantum computers, Phys. Rev. A **90**, 022305 (2014).

[5] R. Babbush, J. McClean, D. Wecker, A. Aspuru-Guzik, and N. Wiebe, Chemical basis of Trotter-Suzuki errors in quantum chemistry simulation, Phys. Rev. A **91**, 022311 (2015).

[6] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, Rev. Mod. Phys. **92**, 015003 (2020).

[7] P. R. C. Kent and G. Kotliar, Toward a predictive theory of correlated materials, Science **361**, 348 (2018).

[8] S. Qiming and C. Garnet Kin-Lic, Quantum embedding theories, Acc. Chem. Res. **49**, 2705 (2016).

[9] H. Ma, M. Govoni, and G. Galli, Quantum simulations of materials on near-term quantum computers, npj Comput. Mater. **6**, 85 (2020).

[10] B. Bauer, D. Wecker, A. J. Millis, M. B. Hastings, and M. Troyer, Hybrid quantum-classical approach to correlated materials, Phys. Rev. X **6**, 031045 (2016).

[11] Y. Yao, F. Zhang, C.-Z. Wang, K.-M. Ho, and P. P. Orth, Gutzwiller hybrid quantum-classical computing approach for correlated materials, Phys. Rev. Res. **3**, 013184 (2021).

[12] B. Jaderberg, A. Agarwal, K. Leonhardt, M. Kiffner, and D. Jaksch, Minimum hardware requirements for hybrid quantum–classical DMFT, Quantum Sci. Technol. **5**, 034015 (2020).

[13] Y. Li and S. C. Benjamin, Efficient variational quantum simulator incorporating active error minimization, Phys. Rev. X **7**, 021050 (2017).

[14] A. Kandala, K. Temme, A. D. Córcoles, A. Mezzacapo, J. M. Chow, and J. M. Gambetta, Error mitigation extends the computational reach of a noisy quantum processor, Nature (London) **567**, 491 (2019).

[15] S. Endo, S. C. Benjamin, and Y. Li, Practical quantum error mitigation for near-future applications, Phys. Rev. X **8**, 031027 (2018).

[16] 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).

[17] P. J. J. O'Malley, R. Babbush, I. D. Kivlichan, J. Romero, J. R. McClean, R. Barends, J. Kelly, P. Roushan, A. Tranter, N. Ding et al., Scalable quantum simulation of molecular energies, Phys. Rev. X **6**, 031007 (2016).

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

[19] J. Romero, R. Babbush, J. R. McClean, C. Hempel, P. J. Love, and A. Aspuru-Guzik, Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz, Quantum Sci. Technol. **4**, 014008 (2018).

[20] K. Temme, S. Bravyi, and J. M. Gambetta, Error mitigation for short-depth quantum circuits, Phys. Rev. Lett. **119**, 180509 (2017).

[21] J. J. Wallman and J. Emerson, Noise tailoring for scalable quantum computation via randomized compiling, Phys. Rev. A **94**, 052325 (2016).

[22] R. LaRose, A. Mari, S. Kaiser, P. J. Karalekas, A. A. Alves, P. Czarnik, M. E. Mandouh, M. H. Gordon, Y. Hindy, A. Robertson et al., Mitiq: A software package for error mitigation on noisy quantum computers, Quantum **6**, 774 (2022).

[23] Y. Kim, C. J. Wood, T. J. Yoder, S. T. Merkel, J. M. Gambetta, K. Temme, and A. Kandala, Scalable error mitigation for noisy quantum circuits produces competitive expectation values, Nat. Phys. **19**, 752 (2023).

[24] S. Zhang, Y. Lu, K. Zhang, W. Chen, Y. Li, J.-N. Zhang, and K. Kim, Error-mitigated quantum gates exceeding physical fidelities in a trapped-ion system, Nat. Commun. **11**, 1 (2020).

[25] X. Bonet-Monroig, R. Sagastizabal, M. Singh, and T. E. O'Brien, Low-cost error mitigation by symmetry verification, Phys. Rev. A **98**, 062339 (2018).

[26] S. McArdle, X. Yuan, and S. Benjamin, Error-mitigated digital quantum simulation, Phys. Rev. Lett. **122**, 180501 (2019).

[27] J. R. McClean, Z. Jiang, N. C. Rubin, R. Babbush, and H. Neven, Decoding quantum errors with subspace expansions, Nat. Commun. **11**, 636 (2020).

[28] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)* (MIT Press, Cmbridge, MA, 2005).

[29] H. Abraham, AduOffei, R. Agarwal, I. Y. Akhalwaya, G. Aleksandrowicz, T. Alexander, M. Amy, E. Arbel, Arijit02, A. Asfaw et al., Qiskit: An open-source framework for quantum computing, https://www.ibm.com/quantum/qiskit (2019).

[30] H. K. Cummins, G. Llewellyn, and J. A. Jones, Tackling systematic errors in quantum logic gates with composite rotations, Phys. Rev. A **67**, 042308 (2003).

[31] K. M. Nakanishi, K. Fujii, and S. Todo, Sequential minimal optimization for quantum-classical hybrid algorithms, Phys. Rev. Res. **2**, 043158 (2020).

[32] M. Ostaszewski, E. Grant, and M. Benedetti, Structure optimization for parameterized quantum circuits, Quantum **5**, 391 (2021).

[33] G. Schmitz, E. L. Klinting, and O. Christiansen, A Gaussian process regression adaptive density guided approach for potential energy surface construction, J. Chem. Phys. **153**, 064105 (2020).

[34] S. B. Bravyi and A. Y. Kitaev, Fermionic quantum computation, Ann. Phys. (NY) **298**, 210 (2002).

[35] A. Anand, P. Schleich, S. Alperin-Lea, P. W. K. Jensen, S. Sim, M. Díaz-Tinoco, J. S. Kottmann, M. Degroote, A. F. Izmaylov, and A. Aspuru-Guzik, A quantum computing view on unitary coupled cluster theory, Chem. Soc. Rev. **51**, 1659 (2022).

[36] J. C. Spall, An overview of the simultaneous perturbation method for efficient optimization, Johns Hopkins APL Tech. Dig. **19**, 482 (1998).

[37] W. W. Ho and T. H. Hsieh, Efficient variational simulation of non-trivial quantum states, SciPost Phys. **6**, 029 (2019).

[38] A. Mukherjee, N. F. Berthusen, J. C Getelina, P. P. Orth, and Y.-X. Yao, Comparative study of adaptive variational quantum eigensolvers for multi-orbital impurity models, Commun. Phys. **6**, 4 (2023).

[39] N. Lanatà, T.-H. Lee, Y.-X. Yao, and V. Dobrosavljević, Emergent Bloch excitations in Mott matter, Phys. Rev. B **96**, 195126 (2017).

[40] M. S. Frank, T.-H. Lee, G. Bhattacharyya, P. K. H. Tsang, V. L. Quito, V. Dobrosavljević, O. Christiansen, and N. Lanatà, Quantum embedding description of the Anderson lattice model with the ghost Gutzwiller approximation, Phys. Rev. B **104**, L081103 (2021).

[41] G. Knizia and G. K.-L. Chan, Density matrix embedding: A simple alternative to dynamical mean-field theory, Phys. Rev. Lett. **109**, 186404 (2012).