


Training Variational Quantum Circuits with CoVaR: Covariance Root Finding with Classical Shadows

Gregory Boyd¹ and Bálint Koczor^{1*}

Department of Materials, University of Oxford, Parks Road, Oxford OX1 3PH, United Kingdom

 (Received 3 May 2022; revised 2 September 2022; accepted 7 October 2022; published 28 November 2022)

Exploiting near-term quantum computers and achieving practical value is a considerable and exciting challenge. Most prominent candidates as variational algorithms typically aim to find the ground state of a Hamiltonian by minimizing a single classical (energy) surface which is sampled from by a quantum computer. Here, we introduce a method we call CoVaR, an alternative means to exploit the power of variational circuits: We find eigenstates by finding joint roots of a polynomially growing number of properties of the quantum state as covariance functions between the Hamiltonian and an operator pool of our choice. The most remarkable feature of our CoVaR approach is that it allows us to fully exploit the extremely powerful classical shadow techniques; i.e., we *simultaneously* estimate a very large number $> 10^4$ – 10^7 of covariances. We randomly select covariances and estimate analytical derivatives at each iteration applying a stochastic Levenberg-Marquardt step via a large but tractable linear system of equations that we solve with a classical computer. We prove that the cost in quantum resources per iteration is comparable to a standard gradient estimation; however, we observe in numerical simulations a very significant improvement by many orders of magnitude in convergence speed. CoVaR is directly analogous to stochastic gradient-based optimizations of paramount importance to classical machine learning while we also offload significant but tractable work onto the classical processor. As we demonstrate numerically, the approach shares features with phase-estimation protocols that prepare eigenstates with a dominant initial fidelity contribution.

DOI: [10.1103/PhysRevX.12.041022](https://doi.org/10.1103/PhysRevX.12.041022)

Subject Areas: Quantum Information

I. INTRODUCTION

Quantum computers are becoming a reality, and with an accelerating pace experiments set more and more impressive records [1–5]. Current generations of machines are already well beyond the 50-qubit frontier and have been demonstrated to being capable of significant computational advantage over the best classical supercomputers. Despite rapid progress in improving hardware, it is generally believed the fault-tolerant, error-corrected systems that are expected to emerge ultimately require significantly better and larger hardware and may, thus, not be within reach in the near term. The reason is that quantum states are highly vulnerable to experimental imperfections, and correcting those errors requires highly nontrivial measures, such as encoding a single logical qubit into potentially thousands of physical qubits.

It is, thus, a very exciting challenge in the near term to achieve practical value with these noisy intermediate-scale quantum (NISQ) [6] devices despite the damaging noise in the hardware. The most promising candidates, generally known as variational quantum algorithms (VQAs) [7–11], are robust against noise given the quantum circuit is restricted to a shallow depth. The most prominent example is the variational quantum eigensolver (VQE) whereby a circuit of shallow depth is constructed of parametrized quantum gates such that the emerging quantum state is powerful enough to express the ground state of a problem of interest, e.g., the Hamiltonian of a chemical system. Nearly all such techniques proceed by efficiently estimating the energy (expected value of the Hamiltonian) or an equivalent cost function via sampling with a quantum computer, and then the circuit parameters are variationally optimized to find the solution to the desired problem. While these techniques seem promising, there are many challenges, especially in reducing high sampling costs and performing nonlinear parameter optimizations which suffer from the presence of local traps and possibly flat regions as barren plateaus [12–15].

Here, we make significant progress toward addressing these challenges: First, our approach converges faster than VQE in order(s) of magnitude fewer iterations and has a

*balint.koczor@materials.ox.ac.uk

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

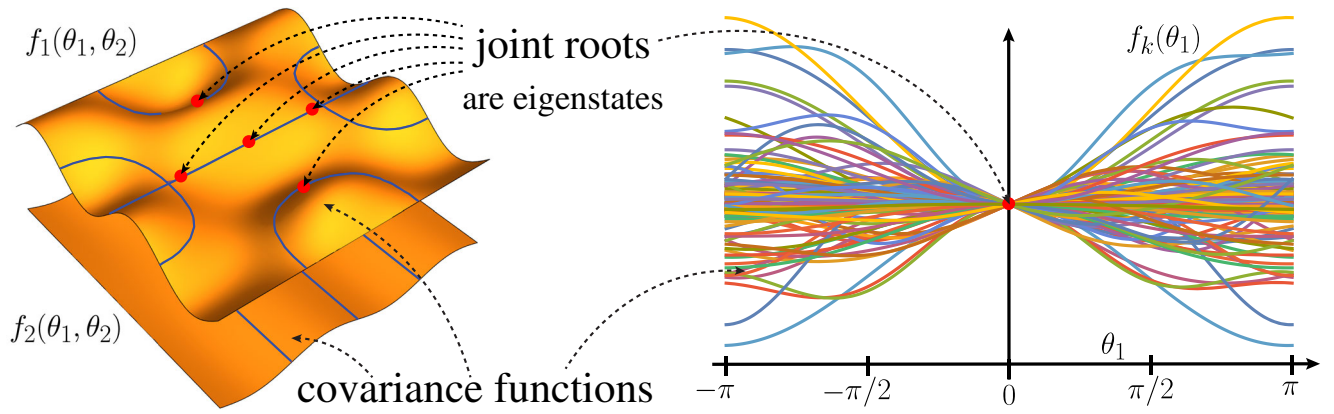


FIG. 1. Left: a toy example of a two-qubit problem whose eigenstates we aim to find by finding parameters of a variational quantum state $|\psi(\theta_1, \theta_2)\rangle$ prepared by two parametrized gates. Covariances $f_k := \langle O_k, \mathcal{H} \rangle_\psi$ between our problem Hamiltonian and between observables O_k span classical surfaces (orange surfaces) and express uncertainty relations between those operators. Blue lines show roots as regions in parameter space where these uncertainties (covariances) vanish; i.e., at roots $\underline{\theta}^*$, the equation is satisfied $f_1(\underline{\theta}^*) = 0$. Intersections of the lines in the above surface with those of the below surface (red dots) guarantee eigenstates of the problem Hamiltonian as joint roots $f_1(\underline{\theta}) = f_2(\underline{\theta}) = 0$. Right: We use the extremely powerful classical shadow techniques to determine a very large number of these covariances $f_k(\theta_1)$ whose slices along the parameter θ_1 are shown in a practically relevant variational circuit (solid lines). We initialize at $\theta_1 \neq 0$ and iteratively find the joint root at $\theta_1^* = 0$ (red dot): We use a Levenberg-Marquardt step whereby we linearize the covariances through computing a Jacobian and solve the resulting large, overdetermined linear system of equations.

logarithmic measurement cost via classical shadows when increasing our constraint size—finding a solution as an eigenstate, thus, has a significantly reduced sampling cost. Second, VQE optimizations are shown to be *NP* hard [12] due to local traps—our approach is particularly robust against local traps due to a stochastic generation of a large number of constraints. Third, since in the present work we resort to local Hamiltonians as we use the NISQ-friendly variant of classical shadows [16], barren plateaus do not necessarily exist and, thus, pose a less significant issue than local traps [17,18].

In contrast to the usual variational minimization of a single cost function, we define an entirely new class of algorithms by leveraging the following observation: In order to find an eigenstate, a large number of properties of the variational quantum state must satisfy certain uncertainty relations with respect to observable measurements. We define these properties as covariances [19–21] between the problem Hamiltonian and elements of an operator pool of our choice. This definition leaves us great flexibility in choosing our operator pools and the ability to pose the problem of finding eigenstates as joint roots of covariances. As we illustrate in Fig. 1 (left), these covariances form surfaces as a function of circuit parameters, and roots of the individual covariances form submanifolds [blue lines in Fig. 1 (left)]. Intersections of these as joint roots (red dots) then correspond to eigenstates of the problem Hamiltonian. In our covariance root-finding (CoVaR) approach, we randomly select a large number of such covariances as illustrated in Fig. 1 (right) and apply powerful classical numerical techniques: We linearize the surfaces by computing their analytical

Jacobian with a quantum computer and solve a large but tractable linear system of equations to estimate the root [$\theta_1^* = 0$ in Fig. 1 (right)]. We iteratively repeat this procedure until a sufficiently good approximation of an eigenstate is found—which we can verify classically efficiently from our reconstructed covariances.

The most significant advantage of CoVaR is that we can use classical shadows to reconstruct these covariances with an extreme efficiency: We prove that the cost of estimating a very large Jacobian is comparable to a standard gradient estimation and grows only logarithmically with the number of covariances. CoVaR is directly analogous to stochastic gradient-based optimizers that have been the *de facto* standard choice for most typical variants of machine learning; e.g., Levenberg-Marquardt is considered to be the fastest method for training classical neural networks [22–25]. As such, CoVaR is a quantum-classical hybrid that ideally combines the fast convergence speed of Levenberg-Marquardt with the logarithmically efficient (quantum) computation of our large Jacobian.

We demonstrate in a comprehensive set of numerical experiments that the efficacy of root finding is significantly increased by employing such large datasets and our optimization procedure is robust against local traps, circuit noise, shot noise, and noise due to random sampling of constraints. We cover a number of important practical applications, such as recompilation, finding ground and excited states of local Hamiltonians, where CoVaR is particularly powerful as we demonstrate in numerical simulations. Given the rapidly growing literature on variational quantum algorithms, we discuss in detail connections and differences to similar approaches.

The structure of this work is the following. In the rest of this introduction section, we briefly introduce covariances and related basic notations in Sec. IA which are both fundamental to quantum mechanics but also form the basis of our approach. In Sec. IB, we then briefly recapitulate notations related to shallow variational quantum circuits. Our main, general results are presented in Sec. II, where we state conditions for finding eigenstates based on covariances and pose our problem as root finding. In Sec. III, we introduce our CoVaR approach that uses classical shadows to find eigenstates of local Hamiltonians and relies on finding joint roots of very large systems. In Sec. IV, we numerically demonstrate the power and utility of CoVaR in solving practical problems, while we compare our technique to various others in Sec. V.

A. Preliminaries: Operator covariances and their properties

In this section, we introduce all necessary tools for deriving our main results. First, recall that a *pure* quantum state is an element of the complex Hilbert space $|\psi\rangle \in \mathbb{C}^d$ with the dimension, e.g., in a system of N qubits $d = 2^N$. We consider observables as Hermitian operators that act on this Hilbert space as complex Hermitian matrices $A \in \mathbb{C}^{d \times d}$. For any pair of such Hermitian operators, we can define the following bilinear form that we refer to as a covariance.

Definition 1 (covariances).—Given two arbitrary Hermitian operators $A, B \in \mathbb{C}^{d \times d}$, we can define a covariance functional between them that depends on a pure quantum state $|\psi\rangle$ via the bilinear form

$$\langle A, B \rangle_\psi := \langle \psi | AB | \psi \rangle - \langle \psi | A | \psi \rangle \langle \psi | B | \psi \rangle \in \mathbb{C}. \quad (1)$$

These covariances are fundamentally important in quantum mechanics, and they are closely related to the statistics when an observable property of a quantum system is measured—covariances then express the compatibility of these observable properties of a quantum system.

It simplifies following derivations to introduce an orthonormal set of Hermitian operators: For example, Pauli strings $O_k \in \{\text{Id}_2, X, Y, Z\}^{\otimes N}$ form a complete orthonormal set with respect to the Hilbert-Schmidt scalar product $\text{Tr}[P_k^\dagger P_l]/2^N = \delta_{kl}$. Here, δ_{kl} denotes the Kronecker delta, and X, Y , and Z are Pauli matrices. Let us now define our operator pool as a suitable set of such operators.

Definition 2 (operator pool).—We define an operator pool \mathcal{P} as a collection of r_p orthonormal Hermitian operators as

$$\mathcal{P} := \{O_k\}_{k=1}^{r_p}, \quad (2)$$

where $\text{Tr}[O_k^\dagger O_l]/2^N = \delta_{kl}$ and $O_k^\dagger = O_k$. For example, our operator pool can be constructed of Pauli strings as

$O_k \in \{\text{Id}_2, X, Y, Z\}^{\otimes N}$, where $k \in \{0, 1, 2, 3\}^N$ and the overall number of terms is denoted as $r_p \leq 4^N$.

Let us define the covariance matrix associated to our operator pool from Definition 2 which depends on a quantum state $|\psi\rangle$.

Definition 3 (complex covariance matrix).—Given a collection of operators $O_k, O_l \in \mathcal{P}$ from Definition 2, we define an associated Hermitian covariance matrix $\mathbf{C}(\psi) \in \mathbb{C}^{r_p \times r_p}$ that depends on a pure quantum state $|\psi\rangle$ and has the matrix entries

$$\begin{aligned} [\mathbf{C}^\dagger(\psi)]_{kl} &= [\mathbf{C}(\psi)]_{kl} := \langle O_k, O_l \rangle_\psi \\ &= \langle \psi | O_k O_l | \psi \rangle - \langle \psi | O_k | \psi \rangle \langle \psi | O_l | \psi \rangle. \end{aligned} \quad (3)$$

Note that the above covariance matrix $\mathbf{C}(\psi)$ expresses fundamental uncertainty relations between the observables O_k in the operator pool via the matrix inequation $\mathbf{C} \geq 0$ [21]. We remark that our definition above of a covariance matrix has complex entries: While this definition simplifies our following arguments, it is worth noting that in the literature other conventions are also commonly used [19–21]. For example, Ref. [21] defines a covariance matrix in terms of the anticommutator as the real part

$$\frac{1}{2} \langle \psi | \{O_k, O_l\} | \psi \rangle - \langle \psi | O_k | \psi \rangle \langle \psi | O_l | \psi \rangle = \text{Re}[\mathbf{C}(\psi)], \quad (4)$$

while the imaginary part is often referred to as the commutator matrix $\frac{1}{2} \langle \psi | [O_k, O_l] | \psi \rangle = i \text{Im}[\mathbf{C}(\psi)]$. We find it convenient to compactly describe the covariance matrix with complex entries, thereby simultaneously referring to both the anticommutator and commutator matrices.

Given the above definitions, we can straightforwardly derive a number of useful identities which we need not prove here.

Corollary 1.—Given the decompositions $A = \sum_k a_k O_k$ and $B = \sum_k b_k O_k$ of Hermitian operators A and B in terms of the orthonormal operator basis from Definition 2, we can obtain the covariance functional from the covariance matrix as

$$\langle A, B \rangle_\psi = \sum_{k,l} a_k b_l [\mathbf{C}(\psi)]_{kl} = \underline{a}^\top \mathbf{C}(\psi) \underline{b},$$

where $\underline{a}, \underline{b} \in \mathbb{R}^{r_p}$ are coefficient vectors of the operators and $\mathbf{C}(\psi)$ is the covariance matrix in this operator basis. We later find it useful to express the special cases as the vector of covariances, or *covariance functions* $\langle O_k, A \rangle_\psi = \mathbf{C} \underline{a}$; i.e., our primary quantities of concern are covariances with the system Hamiltonian $\langle O_k, \mathcal{H} \rangle_\psi$.

The variance of any Hermitian operator A can also be calculated conveniently from the covariance matrix using the decomposition of A into orthogonal operators.

Lemma 1.—Given the decomposition $A = \sum_k a_k O_k$ of any Hermitian operator A in terms of the orthonormal

operator basis from Definition 2, we obtain the variance of the operator as

$$\text{Var}[A] = \langle A, A \rangle_{\psi} = \sum_{k,l} a_k a_l [\mathbf{C}(\psi)]_{kl} = \underline{a}^T \mathbf{C} \underline{a},$$

where $\underline{a}^T \in \mathbb{R}^{r_p}$ is a coefficient vector and $\underline{a} \mathbf{C} \underline{a} \geq 0$ guarantees that \mathbf{C} is positive semidefinite given its Hermiticity. We later find it useful to express the variance in terms of covariance functions $\text{Var}[A] = \sum_k a_k \langle O_k, A \rangle_{\psi}$.

B. Parametrization through variational quantum circuits

A variational quantum circuit $U(\underline{\theta})$ usually refers to a series of ν parametrized quantum gates $U(\underline{\theta}) := U_{\nu}(\theta_{\nu}) \dots U_2(\theta_2) U_1(\theta_1)$ [10] in some specific form or *Ansatz* which often has layers of the form

$$U_l(\theta_l) = \prod_m e^{-i\theta_l H_m} W_m, \quad (5)$$

where H_m is the Hermitian generator of the gate parametrized by θ_m and W_m can be a nonparametrized unitary associated with this layer. We find it useful later to consider specific parametrized gates where $H_m \in \{\text{Id}_2, X, Y, Z\}^{\otimes N}$ is a Pauli string as typical in practice—we refer to these as Pauli gates. The number of gates in a shallow *Ansatz* circuit is usually chosen such that the circuit depth grows slowly, such as $\text{polylog}(N)$ [7–11]. Let us list here a number of well-studied *Ansätze*.

Hardware-efficient Ansätze are designed to be optimized for low circuit depth and maximal expressivity. $U_l(\theta_l)$ are typically chosen as the native gates of the given hardware platform, and the rotation angles θ_l of each gate are treated as parameters to be optimized [26].

The *unitary coupled cluster Ansatz* (UCC) is a problem-inspired *Ansatz* for quantum chemistry. It proposes the candidate ground state using excitations of orbitals from some reference state $|\psi_0\rangle$, typically the Hartree-Fock state as $e^{T(\theta) - T(\theta)^\dagger} |\psi_0\rangle$. Here, the cluster operator $T(\theta)$ [27] is often restricted to single and double excitations, leading to the ‘‘UCCSD’’ *Ansatz* (SD for single and double).

The *quantum alternating operator Ansatz*, *Hamiltonian variational Ansatz*, and *further variants* are motivated by a time-discretized and Trotterized adiabatic evolution that is guaranteed to drag the eigenstate of a trivial Hamiltonian \mathcal{H}_0 to the desired problem Hamiltonian \mathcal{H} for a sufficiently deep *Ansatz*. The evolution time θ_l of each piecewise constant, Trotterized evolution $U_l(\theta_l) = \prod_m e^{-i\theta_l H_m}$ is variationally optimized to find the ground state.

Applying this quantum circuit to an easy-to-prepare reference state of N qubits defines our parametrized *Ansatz* states as

$$|\psi(\underline{\theta})\rangle := U(\underline{\theta})|0\rangle^{\otimes N}.$$

Parameters of the *Ansatz* circuit $\underline{\theta}$ are then varied through classical optimization techniques such that the quantum state $|\psi(\underline{\theta}_{\text{opt}})\rangle$ at the optimal set of parameters is a solution to our problem [10]. Usually, this optimization is done by minimizing a cost function, most typically the energy of a problem Hamiltonian $E(\underline{\theta}) = \langle \psi(\underline{\theta}) | \mathcal{H} | \psi(\underline{\theta}) \rangle$ [8,28], but note that variants of the VQE paradigm allow for the optimization of other cost functions, such as the variance of the Hamiltonian [29] or nonlinear functions of expected values [30].

In the usual case when the gate generators H_m in Eq. (5) are Pauli gates, the cost function $E(\underline{\theta})$ is shown to be a trigonometric polynomial [31]. Finding the global minimum of $E(\underline{\theta})$ as trigonometric functions is shown to be *NP* hard [12] given the rapidly increasing number of local minima.

Here, we introduce a different paradigm; instead of searching for the minimum of a single classical function $E(\underline{\theta})$, we efficiently estimate a large number of covariances that each depend on the set of parameters $\underline{\theta}$, and, thus, each corresponds to a unique surface as a function of $\underline{\theta}$ as illustrated in Fig. 1. We prove that these parametrized covariances are indeed smooth functions of the circuit parameters $\underline{\theta}$.

Lemma 2 (smooth covariance functions).—Given a variational quantum state $|\psi(\underline{\theta})\rangle := U(\underline{\theta})|0\rangle^{\otimes N}$ as defined via a variational quantum circuit $U(\underline{\theta}) \in \text{SU}(2^N)$, we define the parametrized covariances as

$$f_k(\underline{\theta}) := \langle O_k, \mathcal{H} \rangle_{\psi(\underline{\theta})}, \quad \text{with } O_k \in \mathcal{P}. \quad (6)$$

The covariance functions $f_k: \mathbb{R}^{\nu} \mapsto \mathbb{C}$ are smooth, infinitely differentiable functions of the circuit parameters $\underline{\theta} \in \mathbb{R}^{\nu}$ for any Hermitian operator O_k and problem Hamiltonian \mathcal{H} .

Refer to Appendix B 1 for a proof. Above, we introduce the more compact notation for these covariance functions as $f_k(\underline{\theta})$ to highlight that we pose our problem of finding eigenstates by finding simultaneous roots of a cohort of smooth functions $\{f_k(\underline{\theta})\}_{k=1}^{r_p}$. Furthermore, in the practically important special case when the *Ansatz* circuit is composed of Pauli gates, we show that the covariances $f_k(\underline{\theta})$ are actually trigonometric polynomials in $\underline{\theta}$ via Ref. [31].

Corollary 2 (trigonometric polynomials).—In the specific but pivotal scenario when parametrized gates in the *Ansatz* circuit in Eq. (5) are Pauli gates, the covariances are trigonometric polynomials as $f_k(\underline{\theta}) = \sum_{j=1}^{3^{2\nu}} C_j \mathcal{T}_j(\underline{\theta})$, where the prefactors $C_j \in \mathbb{C}$ depend on the index k while $\mathcal{T}_j(\underline{\theta})$ are trigonometric monomials, i.e., products of single-variate sine and cosine functions.

As such, finding parameters such that $f_k(\underline{\theta}) = 0$ for all k is equivalent to finding roots of the corresponding (trigonometric) polynomial system.

II. GENERAL RESULTS: FINDING EIGENSTATES BY FINDING ROOTS

This section introduces the main theoretical underpinnings of our approach in a general setting, i.e., without making any assumptions about the problem Hamiltonian or the type of operator pool. In contrast, in Sec. III, we introduce CoVaR which is a specific, practically motivated approach where we restrict operators to local Pauli strings, which in return allows us to utilize the powerful classical shadow technique. We note that we also investigate another theoretically interesting special case of operator pools in Appendix A.

A. Finding eigenstates of a problem Hamiltonian

Finding approximate representations of eigenstates of a problem Hamiltonian is a key application of near-term quantum computers. The primary hope for quantum advantage in the near term is usually placed on variational quantum algorithms whereby the solution to a problem is encoded into the ground state of a Hamiltonian. Most notable is the variational quantum eigensolver [8], which aims to find the ground state of a Hamiltonian via a variational minimization of the system's energy. Finding excited states is also of particular importance for, e.g., analyzing chemical reactions in drug discovery or in catalysis [32].

Furthermore, applications for finding eigenstates also exist outside of quantum simulation, for example, in solving classical optimization problems using quantum hardware via the quantum approximate optimization algorithm (QAOA). These were introduced to solve problems such as constraint satisfaction and max-cut [7] but have been extended beyond. Furthermore, finding eigenstates is also highly relevant to the continued design and improvement of applications and quantum algorithms. For example, recompilation problems are highly relevant as we show. Similarly, the preparation of logical states in quantum error correction can be cast as eigenstate-finding procedures [33,34].

The aforementioned techniques typically proceed by exploiting the fact that the problem Hamiltonians of interest $\sum_{a=1}^r h_a \mathcal{H}_a$ decompose into a polynomially growing number $r \in \text{poly}(N)$ of Pauli operators whose expected values can be estimated efficiently with a quantum computer. In the following, we denote the collection of these Pauli strings as $\mathcal{Q} = \{\mathcal{H}_a\}_{a=1}^r$. Let us now introduce our main result which uses covariances described in the previous section to finding eigenstates of a problem Hamiltonian.

Theorem 1.—Given the decomposition of a fixed problem Hamiltonian $\mathcal{H} = \sum_{a=1}^r h_a \mathcal{H}_a$ into a set of basis operators $\mathcal{H}_a \in \mathcal{Q} \subseteq \mathcal{P}$ which form a subset of our operator pool, this subset usually has a polynomial size as $r \in \text{poly}(N)$. Given a fixed quantum state $|\psi\rangle$, simultaneous roots of all covariances

$$\text{sufficient conds } \langle \mathcal{H}_a, \mathcal{H} \rangle_\psi = 0, \quad \forall \mathcal{H}_a \in \mathcal{Q}, \quad (7)$$

provide a sufficient condition for the eigenvalue equation $\mathcal{H}|\psi\rangle = \langle \mathcal{H} \rangle_\psi |\psi\rangle$ to hold. Further necessary conditions can be introduced via roots of the covariances

$$\text{necessary conds } \langle O_k, \mathcal{H} \rangle_\psi = 0, \quad O_k \in \mathcal{P}, \quad (8)$$

with respect to any basis operator in our pool O_k .

Proof: Sufficient conds.—A direct calculation shows that the variance of the operator \mathcal{H} can be expressed as $\text{Var}[\mathcal{H}] = \langle (\mathcal{H} - \langle \mathcal{H} \rangle_\psi) | (\mathcal{H} - \langle \mathcal{H} \rangle_\psi) \rangle_\psi$ and, therefore, the condition $\text{Var}[\mathcal{H}] = 0$ immediately implies the eigenvalue equation $\mathcal{H}|\psi\rangle = \langle \mathcal{H} \rangle_\psi |\psi\rangle$. Given our expression $\text{Var}[\mathcal{H}] = \sum_a h_a \langle \mathcal{H}_a, \mathcal{H} \rangle_\psi$ from Lemma 1, simultaneous roots as $\langle \mathcal{H}_a, \mathcal{H} \rangle_\psi = 0$ for all a guarantee that $\text{Var}[\mathcal{H}] = 0$.

necessary conds.—The explicit form of the covariance as $\langle A, \mathcal{H} \rangle_\psi = \langle \psi | A \mathcal{H} | \psi \rangle - \langle \psi | A | \psi \rangle \langle \psi | \mathcal{H} | \psi \rangle$ simplifies when the eigenvalue equation $\mathcal{H}|\psi\rangle = \langle \mathcal{H} \rangle_\psi |\psi\rangle$ is satisfied as $\langle A, \mathcal{H} \rangle_\psi = 0$ for Hermitian operators $A \in \mathbb{C}^{d \times d}$. ■

We note that the individual covariance functions may vanish without implying the presence of eigenstates of the problem Hamiltonian; for example, $\langle \mathcal{H}_a, \mathcal{H} \rangle_\psi = 0$ can be satisfied for a single index a in the special case when $|\psi\rangle$ is an eigenstate of \mathcal{H}_a . These form a submanifold of the smooth covariances when viewed as a function of circuit parameters as illustrated with blue lines in Fig. 1. We therefore predicate that all covariance functions in our operator pool simultaneously vanish (red dots in Fig. 1) for all indexes k , which necessarily implies an eigenstate of the problem Hamiltonian. The problem of searching for eigenstates of \mathcal{H} then becomes that of finding simultaneous roots of a system of covariances.

The above theorem ensures us that in an eigenstate all the exponentially many covariances vanish (necessary conditions); however, it is sufficient to verify only that the polynomially growing number of covariances are zero (sufficient conditions). Of course, $\text{Var}[\mathcal{H}] = 0$ certainly guarantees an eigenstate; however, the experimental estimation of $\text{Var}[\mathcal{H}]$ proceeds by computing expected values of individual Pauli terms and is, thus, informationally equivalent to estimating the above covariances [9–11].

While Eq. (7) lists all sufficient conditions with respect to the minimal operator pool that contains only the Pauli-decomposition terms of our problem Hamiltonian as \mathcal{Q} , in the following, we consider unions such that our operator pool is enlarged as $\mathcal{Q}' := \{\mathcal{H}_a\}_{a=1}^r \cup \{O_k\}_{k=1}^{N_c-r}$ with a polynomially growing number of operators O_k that are

orthogonal to our problem Hamiltonian $\text{Tr}[\mathcal{H}O_k] = 0$ (via not including common terms $\{\mathcal{H}_a\} \cap \{O_k\} = \emptyset$). Roots of all covariances with respect to our enlarged operator pool then signify an eigenstate, and we show below that the enlarged operator pool increases the efficacy of our optimization algorithm, i.e., by overconstraining the Jacobian of our root-finding approach. We refer to the size of our enlarged pool $|\mathcal{Q}'| = N_c$ as the number N_c of constraints.

B. Finding joint eigenstates of commuting observables

Many problems of practical interest are concerned with finding joint eigenstates of a group of observables that all commute with each other. For example, to prepare logical states for quantum error correction, we wish to produce an eigenstate of the generators of the corresponding stabilizer group—these generators mutually commute [35]. Another example is the case of recompilation of quantum circuits. Here, we wish to transform a given gate sequence V into a native gate sequence U with an optimal circuit depth, e.g., to make it resilient to noise.

In the case of both full unitary matrix compilation [33] and fixed input state compilation [36], the problem can be stated as applying U^\dagger after V onto our reference state $|\underline{0}\rangle$ (see Sec. IV A for details), which at the solution $V = U$ corresponds to the identity operation $U^\dagger V = \text{Id}$, and the resulting state $|\underline{0}\rangle$ is then the ground state of the Hamiltonian $-\sum_{j=1}^N Z_j$. While one ultimately aims to find the ground state of this Hamiltonian, note that we can also accept any computational basis states $|n\rangle$ which are simultaneous eigenstates of the mutually commuting terms $\{Z_j\}$. This motivates our approach of finding joint eigenstates of the individual Hamiltonian terms.

Corollary 3.—Let us consider a set of mutually commuting Hermitian operators $\{\mathcal{H}_a\}_{a=1}^r =: \mathcal{Q} \subseteq \mathcal{P}$ as our operator pool with $[\mathcal{H}_a, \mathcal{H}_b] = 0$ for all a and b . Simultaneous roots of all variances

$$\text{sufficient conds } \text{Var}[\mathcal{H}_a] = \langle \mathcal{H}_a, \mathcal{H}_a \rangle_\psi = 0, \quad \forall a,$$

provide a set of sufficient conditions such that the fixed quantum state $|\psi\rangle$ is a simultaneous eigenstate of all \mathcal{H}_a . We can consider further necessary constraints as the simultaneous roots of all covariances

$$\text{necessary conds } \langle O_k, \mathcal{H}_a \rangle_\psi = 0, \quad O_k \in \mathcal{P}, \quad \mathcal{H}_a \in \mathcal{Q},$$

that need to be satisfied by $|\psi\rangle$ for any pair of operators O_k and \mathcal{H}_a .

Proof.—For each individual index a , we can apply Theorem 1 to the corresponding operator \mathcal{H}_a and obtain necessary and sufficient conditions such that $|\psi\rangle$ is an eigenstate of the particular operator \mathcal{H}_a . It follows that if all

sufficient conditions from Theorem 1 are satisfied for all indexes a , as listed above, then $|\psi\rangle$ is a simultaneous eigenstate of all \mathcal{H}_a . ■

C. Conventional techniques for finding roots

As introduced above, our approach is based on estimating operator covariances with a quantum computer which we use to inform our decision of updating parameters of our variational quantum circuit. Our aim ultimately is to find a simultaneous root of these covariances at which parameters the variational state is guaranteed to be an eigenstate of our problem Hamiltonian.

There are a large number of well-established techniques for finding simultaneous roots of vector-valued functions, and almost all such techniques are in some way related to Newton's original method [37,38]. Newton's method proceeds by linearizing the nonlinear (but smooth) vector of covariance functions $\mathbf{f}(\underline{\theta})$ via the first-order Taylor expansion as

$$\mathbf{f}(\underline{\theta} + \Delta\underline{\theta}) = \mathbf{f}(\underline{\theta}) + \mathbf{J}\Delta\underline{\theta} + \mathcal{O}(\|\Delta\underline{\theta}\|^2). \quad (9)$$

Given each covariance function is an infinitely differentiable, smooth function of the parameters $\underline{\theta}$, one can indeed apply Newton's method and can approximate roots by solving the equation $\mathbf{f}(\underline{\theta} + \Delta\underline{\theta}) = \underline{0}$ using the above expansion $\mathbf{f}(\underline{\theta}) = -\mathbf{J}\Delta\underline{\theta}$ and neglecting second-order terms. This results in a linear system of equations which can be solved using techniques from linear algebra.

The approach results in an iterative procedure whereby at every iteration we compute the Jacobian \mathbf{J} with a quantum computer and apply its (regularized pseudo)inverse to the vector of covariances \mathbf{f} to compute the parameter-update rule as

$$\underline{\theta}_{t+1} = \underline{\theta}_t - \mathbf{J}^{-1}\mathbf{f} \quad \text{with} \quad [\mathbf{J}]_{kn} := \partial_n f_k(\underline{\theta}). \quad (10)$$

We derive expressions for computing the Jacobian with a quantum computer in Appendix B 3 using well-established techniques from the literature [9–11], e.g., parameter-shift rules. Furthermore, we also discuss in Appendix C 4 that stacking real parts of \mathbf{J} and \mathbf{f} on top of the imaginary parts results in real $\tilde{\mathbf{J}}$ and $\tilde{\mathbf{f}}$ —enforcing that the solution of the linear system of equations is a real vector $\Delta\underline{\theta}$.

While we aim to compute the Jacobian and the covariances with a quantum computer, we note that the resulting linear systems of equations are solved with a classical computer. It is important to note that we would obtain an underdetermined system of equations if our operator pool were smaller than the number of *Ansatz* parameters as $r_p < \nu$. This is the reason why we require that our operator pool, and thus the dimension of the vector \mathbf{f} , is at least as large as the number of circuit parameters—indeed, later we aim to set up highly overdetermined systems of equations.

While powerful, the vanilla Newton method has its limitations and is guaranteed to converge only when starting near a root—given the linear model in Eq. (9) is accurate only for small $\|\Delta\theta\|$. Nevertheless, a number of advanced techniques have been developed to increase the radius of convergence, and some variants of the Newton method have been proved to be globally convergent under mild continuity conditions of the functions [37,39,40]. In particular, the simplest globally convergent approach first attempts a conventional Newton step, and if the norm of the vector-valued function $\|\tilde{\mathbf{f}}\|$ does not decrease, then a line search is attempted in the step direction $\lambda\tilde{\mathbf{J}}^{-1}\tilde{\mathbf{f}}$ [37], whereby one searches for λ that minimizes $\|\tilde{\mathbf{f}}\|$ along the one-dimensional search direction; see Appendix C 4 for more details. This approach is guaranteed to converge to a root as long as the Jacobian is nonsingular and well conditioned [37].

Another family of closely related approaches are the Levenberg-Marquardt (LM) methods which are additionally robust against ill-conditioned Jacobian matrices. The approach can be shown to be equivalent to the Gauss-Newton algorithm for least-squares minimization with a trust-region method [37]. It attempts steps along what is formally a “regularized Newton direction” via the regularized inverse $\Delta\theta = [A + \lambda\text{Id}]^{-1}\tilde{\mathbf{f}}$ with $A = \tilde{\mathbf{J}}^T\tilde{\mathbf{J}}$ and accepts the regularization parameter λ based on some condition, e.g., such that $\|\tilde{\mathbf{f}}\|$ decreases. The regularization matrix is either Id, but in practice it is often chosen to be the diagonal matrix $\text{diag}(\tilde{\mathbf{J}}^T\tilde{\mathbf{J}})$. In many practical applications of non-linear least-squares fitting, LM can be interpreted as an approximate Hessian optimization; however, we detail in Sec. V D that this is not the case for our root-finding approach.

III. COVARIANCE ROOT FINDING VIA CLASSICAL SHADOWS

In the previous sections, we describe the general theory as the basis for our quantum optimization algorithm. We now detail concrete settings where our approach may achieve significant practical value in exploiting near-term quantum devices. Our aim is that the number of constraints N_c in the linear system of equations is tractable but is significantly larger than the number of circuit parameters as $\nu \ll N_c$. For this reason, we choose a p -local operator pool of size $r_p \in \mathcal{O}(N^p)$ that consists of all Pauli strings that act nontrivially on only p qubits as $\mathcal{P} = \{P_k \in \{\text{Id}_2, X, Y, Z\}^{\otimes N} : P_k \text{ is } p\text{-local}\}$.

This fits very well with the use of classical shadows for determining a very large number of local Pauli strings. In particular, the recent development of the classical shadows method [16] allows us in a NISQ-friendly way to measure N_c covariances and their derivatives using a measurement count that is *only logarithmic* in N_c . Therefore, it is possible to offload processing to the classical computer

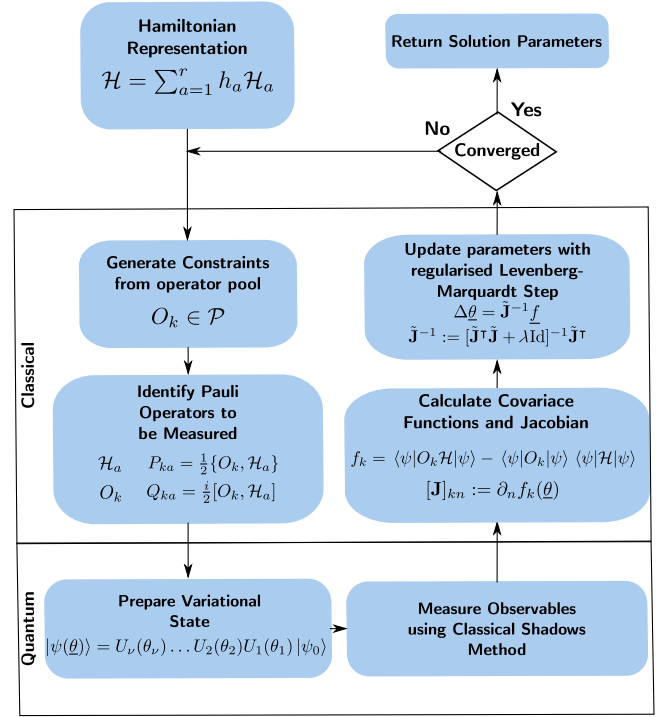


FIG. 2. Flowchart depicting the CoVaR algorithm showing the separation between quantum and classical computations.

with only a small increase in the number of measurements (quantum resources) required. This combination is ideal for NISQ-era algorithms where quantum resources are limited, and it is generally to our benefit if we can offload large but tractable calculations to a classical computer. In the remainder of this section, we describe the application of this method to local Hamiltonians, using a measurement channel of single-qubit Pauli gates [16]. In Fig. 2, we provide a diagrammatic representation of the CoVaR algorithm.

A. Stochastic optimization with very large operator pools

Despite very promising experimental progress [1–5], near-term quantum devices are noisy, and, in order to avoid practically prohibitive accumulation of errors, the circuit depth $D(N)$ is required to be shallow and is usually assumed to grow polylogarithmically as $D(N) \in \text{polylog}(N)$. The Jacobian is generally a non-square matrix with dimension $\mathbf{J} \in \mathbb{C}^{N_c \times \nu}$, where ν is number of *Ansatz* parameters typically scaling as $\nu = N \text{polylog}(N)$ due to shallow circuit depth. As such, for a sufficiently large system we can always overconstrain the Jacobian just by including covariances with respect to only 2-local Pauli strings given then the number of constraints $N_c = \mathcal{O}(N^2)$ grows faster than the number of circuit parameters. We can, thus, conveniently define a very large operator pool for Theorem 1 relative to the number of parameters in the *Ansatz* circuit.

For this reason, we set our operator pool \mathcal{P} to contain all p -local Pauli strings, and we randomly select constraints $f \in \mathbb{C}^{N_c}$ of a large size N_c but much smaller than the full operator pool as $\nu \ll N_c \ll r_p$ —but still much larger than the number of circuit parameters. This construction has the following advantages. First, the large (but tractable) size of the Jacobian yields an overconstrained linear system of equations in Eq. (10), which significantly improves convergence speed as we demonstrate below. Second, randomly choosing constraints has the advantage of navigating out of local traps as we numerically simulate in Appendix D 2. Third, we employ stochastic LM methods that adaptively regularize the Jacobian and are, thus, by construction robust against the noise produced by random choice of constraints, as well as the hardware and shot noise on expectation values—and rigorous proofs of convergence are available in the literature [41,42]. These are indeed properties why stochastic LM and stochastic gradient descent are extremely popular in the classical machine learning context, i.e., due their robustness against noise as well as their robustness against getting stuck in local traps [43–45].

In Fig. 3, we confirm numerically on a 14-qubit recompilation problem that indeed the performance of root finding increases as the number of constraints N_c/ν in the linear system of equations is increased. As we detail below in Sec. IV A, this recompilation problem is a hard benchmarking task with the advantage that our *Ansatz* is capable of expressing the exact solution. In Fig. 3, we run CoVaR for a fixed number 20 of iterations and plot how close the evolution comes to the solution, i.e., the infidelity with respect to the ground state. We assume an idealized simulation with no shot noise or circuit noise; thus, the only source of “error” is the linearization of the nonlinear covariances via Eq. (9), while the performance is significantly improved as we increase the number of constraints.

Let us attempt to intuitively explain on an analytical example why such an increasingly overconstrained system of equations improves our ability to find the solution. Take, for example, the simple case when the *Ansatz* circuit has a *single parameter* θ , and [as illustrated in Fig. 1 (right)], thus, the covariance function vector $\mathbf{f}(\theta)$ can be linearized via Eq. (9) as

$$\mathbf{f}(\theta + \Delta\theta) = \mathbf{f}(\theta) + \mathbf{J}\Delta\theta + \mathcal{O}(|\Delta\theta|^2). \quad (11)$$

Here, the Jacobian is $J_k = f'_k(\theta)$ (assuming J_k and f_k are real, as we stack real and imaginary parts on top of each other). For each individual function $f_k(\theta)$, Newton’s single-variate parameter update approximates the root as $\Delta_k\theta = -f_k(\theta)/J_k$; however, we incur an error $\|\mathbf{f}(\theta + \Delta\theta)\|^2 = \sum_{j \neq k} [f_j(\theta) - f'_j \Delta_k\theta]^2 + \dots$ due to the nonlinearity of $f_k(\theta)$ as we illustrate in Fig. 4 (blue lines). On the other hand, the least-squares solution simultaneously takes into account all linearized constraints as $\mathbf{f}(\theta + \Delta\theta) = \mathbf{0}$

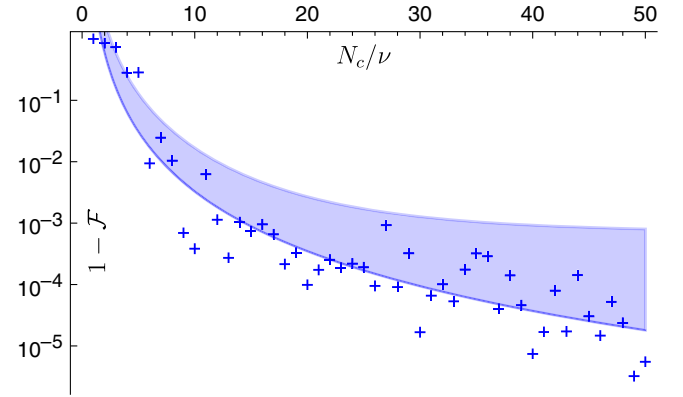


FIG. 3. Performance improvement when we increase the number of constraints N_c illustrated on a 14-qubit recompilation problem of “rediscovering” unknown parameters ($\nu = 124$) of an *Ansatz* (refer to Sec. IV A for more details). CoVaR is run for a fixed number 20 of iterations (initial average fidelity $\mathcal{F} = 46 \pm 7\%$), and the final achieved infidelity $1 - \mathcal{F}$ with respect to the ground state is reported: Blue crosses show the best of three runs of CoVaR, and we plot their fit with the function $a(N_c/\nu)^{-b} + c$ (blue curve) as we detail in Appendix E 1. The worst of three runs are also fitted with the same function (blue shade), confirming a polynomially (in N_c) increasing performance with $b \approx 3.2$ —although we expect this degree to depend on the problem. The large spread of data points is due to our randomly generated constraints (with respect to 3-local Pauli strings).

and is given analytically as $\Delta\theta = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \mathbf{f} = -\sum_k J_k f_k(\theta) / [\sum_k J_k^2]$, which inherently minimizes the aforementioned error via $\|\mathbf{f}(\theta + \Delta\theta)\|^2 \rightarrow \min$. Indeed, the least-squares solution (Fig. 4, orange line) approximates the solution much better than either the individual, single-variate Newton solutions (blue lines) or their average (black line).

Let us now analyze the time complexity of classically computing the (pseudo)inverse of the Jacobian \mathbf{J} . We prove in Appendix C 6 that computing the least-squares solution to the linear systems of equations is dominated by the step of computing $A := \mathbf{J}^T \mathbf{J}$, which can be performed in time $\mathcal{O}(\nu^2 N_c)$ and, as such, scales linearly with the number of constraints. The rest of the procedure, including the

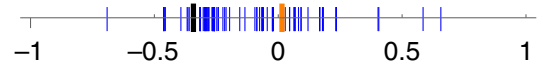


FIG. 4. Illustrating the advantage of overdetermined systems of equations. The same variational circuit as in Fig. 3 with all parameters optimal except for a single variable, which is disturbed as $\theta = 0.5$, and, thus, the exact root is at $\theta^* = 0$. Blue lines show single-variate Newton steps computed for 100 individual covariance functions $\theta + \Delta_k\theta$, and their average is represented by the black line. With the orange line, we obtain a better approximation of the root by solving a $N_c = 176$ overdetermined system of linear equations (see the main text).

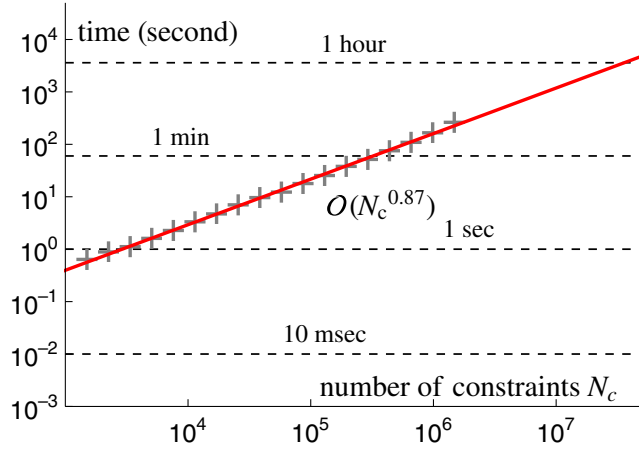


FIG. 5. Computation time with a *Mathematica* code of the update rule $\Delta\theta = \tilde{\mathbf{J}}^{-1}\mathbf{f}$ via the regularized inverse of the non-square matrix Jacobian $\tilde{\mathbf{J}} \in \mathbb{R}^{2N_c \times \nu}$ with a fixed number of circuit parameters $\nu = 1000$ and increasing number of constraints (covariances) N_c . The time complexity is $\mathcal{O}(N_c)$ linear in the leading dimension, as the number of constraints and the absolute time is very reasonable, i.e., less than an hour even for very large matrices with $N_c = 10^7$ (computed with a desktop PC).

computation of the inverse of the small square matrix $A \in \mathbb{C}^{\nu \times \nu}$, can then be computed in negligible additive time. Given the number of constraints grows at most as $N_c \leq \mathcal{O}(N^p)$ for our specific choice of p -local Pauli strings, the computation time t grows at most as $t \leq \mathcal{O}[N^{p+2}\text{polylog}(N)]$ with the number of qubits N .

We confirm these expectations in Fig. 5 and estimate that a very large matrix with $N_c = 10^6$ constraints for a variational circuit of $\nu = 10^3$ can straightforwardly be computed in a matter of minutes and fits into the RAM of a typical single node—while distributed computation for larger datasets $> 10^7$ is possible with negligible communication between nodes. We expect determining necessary expected values from classical shadows has a comparable computation time, which we detail below.

B. Noise robustness

Let us now demonstrate the aforementioned noise robustness of our approach: As we experimentally estimate the Jacobian and the covariances, we always incur a certain amount of shot noise (due to finite sampling) but also possible noise due to experimental imperfections. While we demonstrate in a noise-free environment that performance is improved when increasing the number of constraints, one might think that it could also lead to an accumulation of noise. For this reason, we prove in Appendix C 5 that the error in our estimate of the update rule $\Delta\theta$ in Sec. III A does not accumulate as we increase the number of constraints; i.e., the error is constant bounded by the worst-case error in a *single Jacobian or covariance entry*.

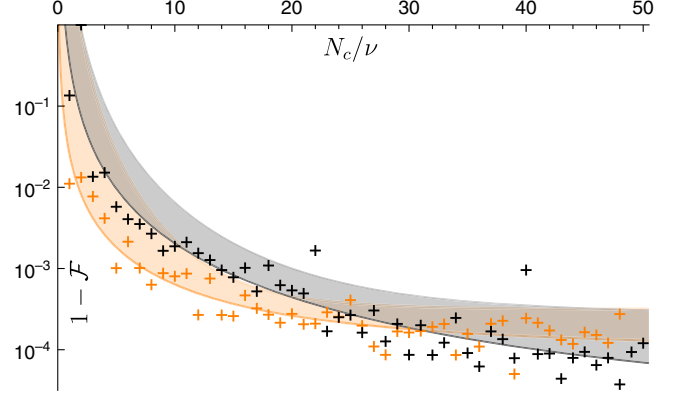


FIG. 6. Simulations and fits are identical to those in Fig. 3 but now with added noise. Orange shows results including a shot noise for $N_s = 10^5$ shots, modeled as independent Gaussian noise on all expectation values. At small N_c , shot noise regularizes the ill-conditioned linear system of equations and does seemingly improve performance. Black results show the performance of CoVaR under simulated circuit noise as described in Appendix E 2, indicating the resilience of the method to reasonable levels of circuit noise. In the large- N_c limit, a noise floor is approached whose magnitude in our particular example is comparable for both cases of only shot noise (orange) and only circuit noise (black)—large spread of the data points is due to our randomly generated constraints, and fits of the best (solid curves) and worst (shaded area) of three runs of CoVaR are included.

We obtain a similar conclusion for the error (shot noise) propagation in the general multivariate case by applying the error propagation formula in Ref. [15] for matrix inversion. In particular, the error in the update rule scales with the fourth power of the smallest inverse singular value (or regularization parameter λ^{-4}) of \mathbf{J} . Given singular values of our $\nu \times N_c$ -dimensional Jacobian matrix grow with N_c , we expect CoVaR is particularly robust against shot noise. In our $\nu = 1$ -dimensional example in the previous subsection, we have a singular value $[\sum_{k=1}^{N_c} J_k^2]^{1/2}$ of $\mathbf{J}^\dagger \mathbf{J}$ which indeed grows with the square root of N_c for nonzero derivatives $|J_k| > 0$.

In Fig. 6, we repeat our simulations from Fig. 3 with added shot noise and circuit noise. In particular, Fig. 6 (orange) shows our simulations with only shot noise added and confirms our above analytical arguments: As we increase N_c/ν , the optimization is able to come closer and closer to the root in a fixed number 20 of iterations up until a point when we reach a shot-noise floor \mathcal{E} where the performance is no longer increased. This shot-noise floor is indeed below the precision of determining individual entries $N_s^{-1/2} \propto 10^{-2.5}$.

Furthermore, Fig. 6 (black) shows the performance of root finding under simulated circuit noise but without shot noise. As we detail in Appendix E 2, we have assumed two- and single-qubit gate error rates $\epsilon_2 = 10^{-3}$ and $\epsilon_1 = \epsilon_2/4$, respectively, which is comparable to the performance of

state-of-the-art hardware. While the optimization is performed with noise, the plotted infidelities are calculated without noise to reflect that the correct parameters are found, as, e.g., error mitigation techniques are typically applied for extracting noise-suppressed expected values from a final state [46–49]. These results show a very similar performance to the case with shot noise only in Fig. 6 (orange): The performance is increased up until a point where a noise floor is reached—and the magnitude of this noise floor in our example appears to be very close to the case of shot noise only. Interestingly, our approach finds circuit parameters very close to the ideal ones (small final infidelities) despite circuit noise—this indeed resembles the phenomenon of optimal parameter resilience [50], meaning this recompilation task is not merely learning in the applied circuit noise. These simulations confirm the robustness of our approach against experimental noise.

C. Estimating a large number of covariances via classical shadows

Recall that a p -local problem Hamiltonian can be specified in terms of its Pauli decomposition as $\mathcal{H} = \sum_{a=1}^r h_k P_a^{(p)}$ where the Pauli strings $P_a^{(p)} \in \mathcal{Q}^{(p)}$ are p -local; i.e., they act on p qubits only nontrivially. Such local Hamiltonians are highly relevant in many important problems which include, for example, recompilation, spin models in materials science, boolean satisfiability problems (3SAT), and fermionic models using mappings that retain operator locality [9,11,51–54].

Let us consider an operator pool \mathcal{P}^q that contains all q -local Pauli strings and, thus, has size $r_p \in \mathcal{O}(N^q)$. We randomly choose covariances f_k from this operator pool such that $N_c \ll r_p$, and as we discuss we aim to estimate a large number of covariances (constraints) via a large but tractable N_c . Let us now establish that we need only reconstruct expected values of at most $p + q$ -local Pauli strings in order to determine all covariances.

Statement 1.—Given a p -local problem Hamiltonian \mathcal{H} , we can estimate covariances $\langle P_k^{(q)}, \mathcal{H} \rangle_\psi$ with respect to at most q -local Pauli strings by reconstructing expected values of at most $p + q$ -local Pauli strings of the form $\langle \psi | P_k^{(p+q)} | \psi \rangle$.

Proof.—Let us explicitly write the covariances as

$$\begin{aligned} \langle P_k^{(q)}, \mathcal{H} \rangle_\psi &= \sum_{a=1}^r h_k \langle P_k^{(q)}, P_a^{(p)} \rangle_\psi = \sum_{a=1}^r h_k \langle \psi | P_k^{(q)} P_a^{(p)} | \psi \rangle \\ &\quad - \sum_{a=1}^r h_k \langle \psi | P_k^{(q)} | \psi \rangle \langle \psi | P_a^{(p)} | \psi \rangle. \end{aligned} \quad (12)$$

Above, the product of the Pauli strings $P_k^{(q)} P_a^{(p)}$ is proportional to a $p + q$ -local Pauli string as $P_k^{(p+q)}$ up to possibly a prefactor $\pm i$ depending on whether $P_k^{(q)}$ and $P_a^{(p)}$

commute or anticommute, etc., as discussed in Appendix B 3. As such, above, we obtain a weighted sum of only expected values of Pauli strings, and, thus, we conclude that any covariance of the form $\langle P_k^{(q)}, \mathcal{H} \rangle_\psi$ can be reconstructed by estimating expected values of at most $p + q$ -local Pauli strings. ■

Note that determining all covariances that satisfy the sufficient conditions in Eq. (7) requires that the locality of the operator pool is at least as large as the locality of the problem Hamiltonian via $q \geq p$.

The classical shadow procedure [16] fits very well with our CoVaR approach, as it allows us to estimate a very large number of these covariances such that the number of samples (quantum resources) increase only logarithmically with the number of constraints—while the required measurements are very NISQ friendly. Let us briefly recapitulate the main steps to reconstructing Pauli strings using classical shadows.

- (i) We apply a random unitary U to rotate the state. In our case of local Pauli strings, the unitaries are chosen randomly from single-qubit Clifford gates on each qubit, and the procedure is, thus, equivalent to randomly selecting to measure in the X , Y , or Z bases—we measure each qubit to obtain N -bit measurement outcomes $|\hat{b}_i\rangle \in \{0, 1\}^N$.
- (ii) We then generate the classical shadows by applying the inverse of the measurement channel \mathcal{M} , which can be done efficiently as the channel chosen is a distribution over Clifford circuits. The classical snapshots are generated as $\hat{\rho}_i = \mathcal{M}^{-1}(U^\dagger |\hat{b}_i\rangle \langle \hat{b}_i| U)$, and the classical shadows are collections of these snapshots $\mathcal{S}(\rho; N) = [\hat{\rho}_1, \dots, \hat{\rho}_N]$.
- (iii) From these classical shadows, we can construct K estimators of ρ from our N_{batch} snapshots as $\hat{\rho}^{(k)} = (1/r) \sum_{i=(k-1)r+1}^{kr} \hat{\rho}_i$ with $r = \lfloor N_{\text{batch}}/K \rfloor$ and classically calculate estimators of the Pauli expectation values $\hat{\rho}_i(N, K) = \text{median}\{\text{Tr}(O_i \hat{\rho}_{(1)}), \dots, \text{Tr}(O_i \hat{\rho}_{(K)})\}$. The classical computational resources are quite modest.
- (iv) The sample complexity of obtaining these estimators of M Pauli operators of locality l to error ϵ is $\mathcal{O}[3^l \log(M)/\epsilon^2]$.

Using classical shadows allows us to reconstruct all $(p + q)$ -local Pauli strings with a sample complexity that is merely logarithmic in the system size. This fits particularly well with the preset approach: When the locality $p + q$ of Pauli strings is modest, then we can obtain a large, polynomially growing number of constraints $N_c \in \mathcal{O}(N^q)$. Furthermore, given the covariances are fully determined by expected values (of local Pauli strings), we show that their analytical derivatives can be estimated using expected values at shifted circuit parameters via the so-called parameter-shift rules [55]. In particular, each partial derivative in the Jacobian $[\mathbf{J}]_{kn} := \partial_n f_k(\underline{\theta})$ is determined by

estimating expected values at two different shifted parameters as discussed in Appendix C 1. As such, we can fully determine our Levenberg-Marquardt step just using expected values of local Pauli strings.

Let us now state the sample complexity of CoVaR whose (quantum) cost is dominated by estimating the Jacobian, and let us compare it to the cost of determining a gradient vector used in energy minimization.

Statement 2.—Given a p -local problem Hamiltonian \mathcal{H} , we use classical shadows to determine a large number N_c of covariances with respect to q -local Pauli strings. The sample complexity of determining the Jacobian of size $\mathbf{J} \in \mathbb{C}^{N_c \times \nu}$ to an error ϵ is

$$N_s = \mathcal{O}[3^{p+q}\nu \log(rN_c)/\epsilon^2].$$

In contrast, determining the gradient of the energy expected value $\langle \mathcal{H} \rangle$ using classical shadows has a complexity $\mathcal{O}[3^p \nu \log(r)/\epsilon^2]$. As such, determining a very large Jacobian is only logarithmically more expensive than determining an energy gradient (up to a multiplicative constant 3^q that depends on the modest locality of our choice, e.g., $q = 2, 3$).

Proof.—Theorem 1 in Ref. [16] establishes that M Pauli strings O_i of locality l can be estimated to precision parameters ϵ and δ via the number of batches $K = 2 \log(2M/\delta)$ and the number of samples in the individual batches as $N_{\text{batch}} = (34/\epsilon^2) \max_i \|O_i\|_{\text{shadow}}^2$. This results in an overall number of samples $N_{\text{batch}}K$, and the norm is given in Lemma 3 in Ref. [16] as $\|O_k\|_{\text{shadow}}^2 = 3^l$.

We establish in Lemma 6 that we can determine the Jacobian matrix $\mathbf{J} \in \mathbb{C}^{N_c \times \nu}$ by applying the classical shadow procedure $2\nu + 1$ times at different circuit-parameter configurations with $M \leq 3rN_c$. As such, determining these Pauli strings of locality $l = p + q$ requires the number of samples $N_{\text{batch}}K \leq (68/\epsilon^2)3^{p+q} \log(6rN_c/\delta)$. Given we apply the classical shadow procedure $2\nu + 1$ times, we obtain the following upper bound on the sample complexity:

$$N_s \leq (2\nu + 1)3^{p+q} \frac{68}{\epsilon^2} \log(6rN_c/\delta).$$

In contrast, determining a gradient vector for gradient descent requires 2ν applications of the classical shadow procedure, each with $M = r$, and, thus, we obtain the sample complexity

$$N_s^{(\text{grad})} \leq 2\nu 3^p \frac{68}{\epsilon^2} \log(2r/\delta).$$

In both cases, we determine the necessary Pauli strings to precision ϵ , and both the energy gradient and the covariance Jacobian are then obtained from these as a linear

combination with respect to Hamiltonian coefficients which leads to a worst-case error propagation of $\mathcal{O}(r)$. ■

Actually, this bound on the number of required measurements in terms of the locality is noted to be conservative, and it is expected that the actual constants are *much smaller* in practice [16]. Furthermore, the inclusion of the development of derandomized classical shadows [56] has the potential to significantly reduce the number of required measurements; i.e., an order of magnitude reduction has been demonstrated in numerical experiments [16]. These techniques could, thus, greatly improve the speed at which the covariances can be extracted by optimizing the Pauli measurement basis to the specific Pauli strings in our operator pool—but we do not consider these in our above performance bounds. Furthermore, the overhead of CoVaR relative to determining a single gradient vector in gradient descent is the constant 3^q (up to the logarithmic dependence on N_c) and is due only to the increase in the locality of Pauli strings with $q = 2, 3$, etc. We can, thus, expect that determining a *very large* Jacobian has a comparable complexity to determining just a single gradient vector in gradient descent. We demonstrate in the following that this increased size of the Jacobian has significant advantages in practical applications.

IV. APPLICATIONS

A. Recompilation

The ability to recompile a given quantum circuit into an equivalent but practically feasible or more favorable representation is crucial for the successful exploitation of quantum computers. The ideas exist in many variants, from the application of classically tractable analytical gate-replacement rules to automatic discovery techniques [57–59]. In variational recompilation, we want to find a parametrized unitary circuit $U(\underline{\theta})$ that approximates a target unitary V . This target unitary is required to approximate the action of U on the entire Hilbert space in case of recompiling a full unitary [33] in Fig 7(b) or just to approximate the action on a specific input state in Fig. 7(a). After applying the circuits V and $U(\underline{\theta})^\dagger$ consecutively, the goal is to find circuit parameters $\underline{\theta}$ such that the state of the registers is in the ground state $|0\rangle$ of the Hamiltonian $\mathcal{H} = -\sum_{j=1}^N Z_j$ [36]. However, the problem would be equally solved by finding *Ansatz* parameters to produce any computational basis state (i.e., any eigenstate of \mathcal{H}), given we can then just append single-qubit X rotations to the *Ansatz* to produce the desired operation. This feature, along with the local Hamiltonian allowing for the efficient measurement of many covariances, makes it particularly amenable to root finding which is not limited to searching only for the ground state.

For this reason, we apply Corollary 3 to the present problem and define our problem Hamiltonians as $\mathcal{Q} := \{Z_a\}_{a=1}^N$. We can indeed enlarge this pool by further

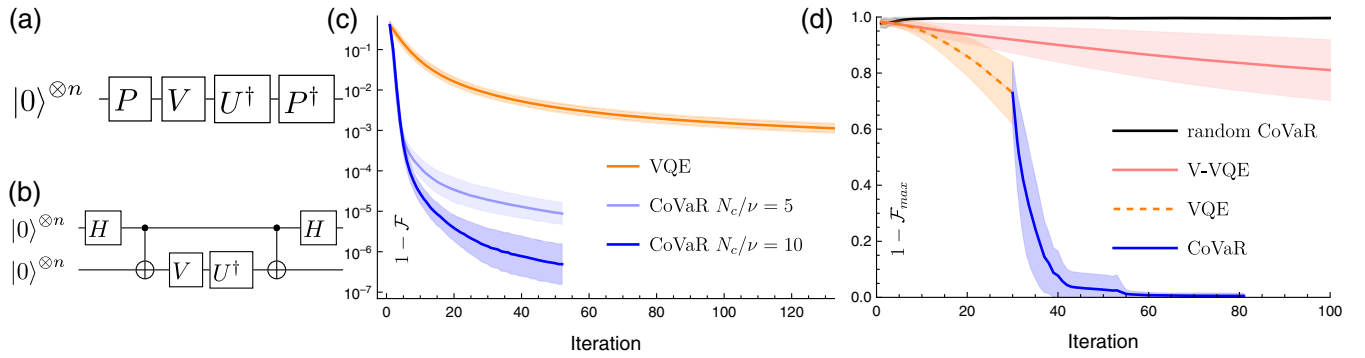


FIG. 7. Demonstration of CoVaR in fixed input state recompilation for a ten-qubit and two-layer ($\nu = 88$) *Ansatz*. The circuit in (a) is used with V and U being the same *Ansatz* but with V having solution parameters θ^* that are randomly chosen as $|\theta_k^*| \leq 2\pi$ but remain fixed over 20 runs. We rediscover these hidden parameters using the *Ansatz* $U(\theta)$, where $\theta = \theta^* + \Delta\theta$. All curves show mean and standard deviation of infidelities over 20 runs, and simulations include shot noise ($N_s = 10^5$). (b) shows the circuit for the recompilation of a full unitary. (c) Performance of CoVaR (blue lines) using $N_c/\nu = 5, 10$ compared with gradient descent (orange). Infidelity $1 - \mathcal{F}$ from the ground state is plotted as we initialize close to the solution via small random parameter perturbations $|\Delta\theta_k| \leq 0.3$ resulting in an initial fidelity $1 - \mathcal{F} = 41 \pm 8\%$. (d) Performance when initialized completely randomly in parameter space via $|\Delta\theta_k| \leq 2\pi$, showing the infidelity $1 - \mathcal{F}_{\max}$ to the nearest computational basis state. Black shows progress for CoVaR and red for variance-VQE—when both are initialized randomly, both fail to make significant progress, regardless of choice of N_c . Additionally shown is the use of a short period of gradient descent to “initialize” (orange, dashed line) and then CoVaR (blue, $N_c/\nu = 20$), which reaches a final infidelity of 0.5% on average (the blue curve includes only the 16/20 runs which are able to converge).

considering products of single-qubit Pauli Z operators. Our aim is then to find joint roots of the covariances from Corollary 3 which then guarantee that the solution corresponds to a joint eigenstate of all operators in \mathcal{Q} as one of the computational basis states. After having found one of these computational states, we just apply single-qubit X rotations to our *Ansatz* to map to the $|0\rangle$ state.

Here, we consider an example of *parameter rediscovery* as a benchmark, whereby we recompile a unitary $V = U(\theta^*)$ that has the same form as the parametrized quantum circuit $U(\theta)$ but with the parameters fixed at some random solution values θ^* . This has the advantage of being a very hard problem to solve variationally while also giving us a guarantee that our circuit is capable of expressing the solution—note that below we benchmark our CoVaR approach on practical problems as well.

Figure 7(c) shows the performance of root finding when we initialize relatively close to the solution (by disturbing parameters $|\Delta\theta_k| \leq 0.3$) on a ten-qubit, two-layer parameter rediscovery problem and compares it to the performance of gradient descent. In this recompilation problem, our operator pool \mathcal{P} contains all 3-local Pauli strings, and we choose randomly N_c operators at every iteration; see details in Appendix E 2. Indeed, Fig. 7(c) confirms that root finding is able to converge significantly faster to the shot-noise floor, i.e., a limitation due to finite sampling of expected values. Furthermore, Fig. 7(c) confirms that root finding has a significantly improved convergence rate (steeper slope), which is improved with a greater number of constraints (blue vs light blue), while note that the quantum resources required for a single iteration are comparable to those of gradient descent.

Figure 7(d) (blue) shows applying root finding to an initial state that we obtain by a short period of gradient descent from a random state—applying gradient descent to a random state has the effect of producing a state with an appreciable overlap with the lowest-energy eigenstates, allowing root finding to efficiently converge. In contrast, in Fig. 7(d) (black), we demonstrate the performance of root finding when we start from a randomly chosen initial point in parameter space on the same problem. It fails to make any progress; this is due to the fact that root finding works well when there are only a small number of eigenstates that significantly contribute to the state produced by the parametrized quantum circuit. In contrast, random states as nearly equal superpositions of a large number of basis states do not contain a dominant eigenstate toward which root finding could converge; thus, CoVaR fails to make significant progress. This is very much analogous with fault-tolerant phase-estimation protocols which do indeed similarly fail under random initialization but enable us to efficiently prepare any eigenstate given a good initialization is possible. These signify the importance of initialization when searching for eigenstates and clearly demonstrates that even a short period of gradient descent may be sufficient for these purposes.

The performance of variance-VQE (a gradient based method that minimizes the variance of the Hamiltonian; see Sec. V B for more details) is also shown for comparison—it is another method which, like root finding, is not searching only for the ground state of the Hamiltonian. Variance-VQE is not stuck in the same way as root finding but makes slow progress due to its relatively (compared to root finding) slow convergence speed.

B. Spin models

Spin models are highly relevant in the study of condensed matter physics and quantum statistical mechanics, and many problems in practice can be mapped to spin models, such as *NP* problems [60]; cf. approximate optimization algorithms (QAOA) or spin systems in materials science [9,11,52–54]. Furthermore, lattice models of quantum field theories [61] typically have local Hamiltonians.

Here, we simulate CoVaR when used to search for low-energy excited states of a spin chain described by the Hamiltonian

$$\mathcal{H} = J \sum_{i=1}^N \vec{\sigma}_i \cdot \vec{\sigma}_{i+1} + c_i Z_i. \quad (13)$$

We use periodic boundary conditions ($N + 1 = 1$), c_i are randomly selected on-site interactions comparable in strength to the couplings J , and $\vec{\sigma}_i$ is the Pauli vector for the i th qubit. This Hamiltonian cannot be simulated classically for large N with reasonable computational resource despite its simple structure [62,63]—and it is relevant for studying the phenomenon of many-body localization in condensed matter systems [64].

We use a hardware-efficient *Ansatz* of 20 layers for ten qubits. Before root finding is performed, the *Ansatz* parameters are initialized to $\underline{\theta}_{\text{imag}}$ using imaginary time evolution [65] to a state with low expected energy. This ensures that only a limited number of eigenstates contribute significantly to the state produced by the *Ansatz*. CoVaR is then performed from points in parameter space with small random variations from $\underline{\theta}_{\text{imag}}$ to map out the low-energy subspace. This process is shown in Fig. 8 in mapping out the lowest-energy subspace of a ten-qubit spin chain; CoVaR rapidly finds a state an order of magnitude closer to the ground state in energy error than imaginary time evolution is able to converge to. This indeed confirms that root finding has a significantly improved convergence rate when compared to imaginary time evolution, where the latter is equivalent to natural gradient [66,67] and is, thus, a second-order method that requires increased absolute quantum resources (samples) [66]. While here we focus on practical applications of CoVaR, we demonstrate additional numerical simulations in Fig. 12 whereby we explicitly compare CoVaR to VQE; we conclude that, in comparison to other variational methods, CoVaR again exhibits superior performance on the present spin-ring problem.

C. Finding excited states

Searching for excited states of a Hamiltonian is important in many practical applications, and, indeed, variational quantum algorithms have been proposed for solving this class of problems. On the other hand, finding excited states using orthogonality constrained VQE techniques [36,68]

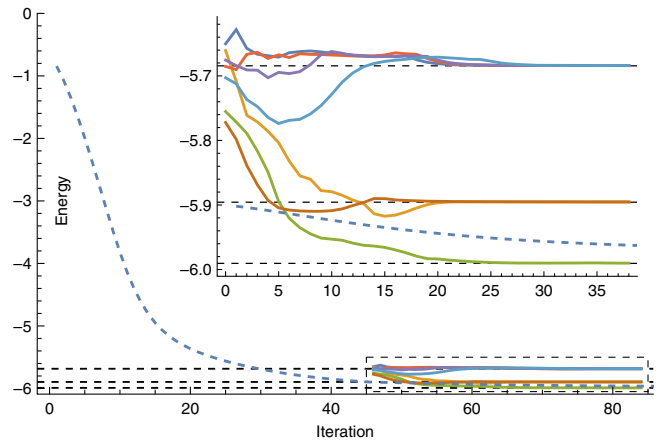


FIG. 8. Using root finding to explore a low-energy subspace of a ten-qubit spin chain via an *Ansatz* circuit of 20 layers ($\nu = 610$). The initialization to the low-energy subspace is performed by imaginary time evolution (blue, dashed line) and is followed by runs of CoVaR to find the three lowest eigenstates (marked by dashed lines), to an accuracy of $\Delta E \leq 4 \times 10^{-4}$ in all cases. Imaginary time evolution is run for 250 iterations after the point where root finding starts and converges to a state with $\Delta E = 0.012$. These simulations do not include shot noise.

can be difficult, because we need to discover the parameters for and project out every state from the ground state up to the energy of the state we wish to find. CoVaR is agnostic to the energy of the state and acts to find states close to the one it is initialized into. Although CoVaR could converge to states that have been found in previous runs of the algorithm, similar techniques of projecting out previously found eigenstates could also be applied. We can also potentially use classical techniques, such as interval analysis [57], to find all the roots within an area of parameter space, reducing the problem converging to an already known eigenstate (root).

In Fig. 9, we show the probability of converging to the ground state of our spin chain as a function of the overlap between the ground state and the initial state. These initial states are obtained by performing imaginary time evolution from random points in parameter space down to energies between -5.8 and the ground state (at $E = -5.99$). The initial fidelity where root finding is started is recorded (Fig. 9, horizontal axis), and the fraction of runs that converge to the ground state is listed. Indeed, we find a nearly linear relationship, and the vast majority of runs that do not converge to the ground state converge to one of the other low-lying energy levels. We provide the distribution of how these runs converge in Appendix D 3. Furthermore, the observed relation between fidelity and probability is directly analogous to phase estimation whereby a measurement is used to collapse the system into the desired eigenstate with a probability that is given by the fidelity with respect to that state. In contrast, variational quantum algorithms converge to local minima, and only an

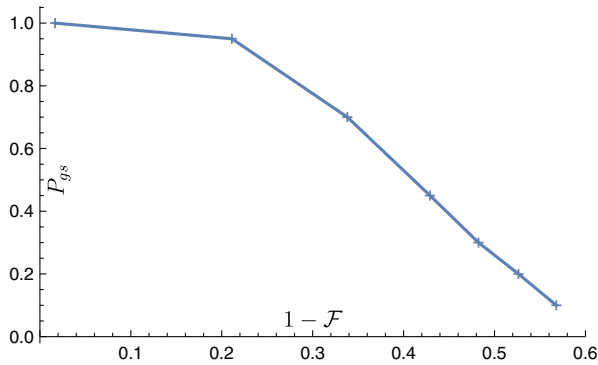


FIG. 9. Probability P_{gs} as the proportion of 20 runs where CoVaR converges to the ground state of the spin chain as a function of starting infidelity from the ground state. While here we report only the probability of converging to the ground state specifically, we note that 97.1% of all runs converge to one of the low-energy eigenstates—the distribution of eigenstates found is detailed in Appendix D 3. Initialization is done the same way as in Fig. 8 by running imaginary time evolution from a random initial state until reaching selected energies between -5.8 and the ground state. A linear decay of success probability—reminiscent of phase-estimation protocols—with ground-state overlap is found, although we expect the rate of this decay to strongly depend on the population distribution of the lowest-energy eigenstates.

exponentially small fraction of such local energy minima may be close to the ground state as proved in Ref. [17].

V. COMPARISON TO EXISTING TECHNIQUES

A. Relation to variational quantum algorithms

A large subset of variational quantum algorithms is concerned with minimizing a cost function $E(\theta)$ which is usually the expected value of a Hamiltonian $\langle \mathcal{H} \rangle$, such as in the case of VQE. Among other practical limitations, this surface may have a large number of local minima that can trap local optimizers [69]. The main difference is that CoVaR uses a large number of such surfaces that are randomly selected, and computing these large data requires similar quantum resources (shots) as a standard gradient estimation in the case of variational quantum algorithms. In the following sections, we compare CoVaR in more detail to specific variational quantum algorithms and related techniques.

B. Comparison to variance minimization

Minimizing the variance $\sigma^2 := \langle \mathcal{H}^2 \rangle - \langle \mathcal{H} \rangle^2$ of a Hamiltonian allows us to find eigenstates and has been explored in the context of quantum chemistry [29]. Furthermore, the so-called variance-VQE [70] approach uses variational quantum circuits and estimates this variance as well as its derivative using a quantum computer. Note that the tools we introduce can naturally be applied in this context: Given a minimal pool $\mathcal{P} = \{\mathcal{H}_a\}_{a=1}^r$ that

contains only Pauli terms of the Hamiltonian and, thus, a covariance matrix $\langle \mathcal{H}_a, \mathcal{H}_b \rangle_\psi$ of the Hamiltonian terms only, our Jacobian allows us to compute the gradient of the variance $\partial_n \sigma^2 = \mathbf{J}^\top \mathbf{h}$, as we detail in Appendix C 3.

While our CoVaR approach contains full information about the variance and its gradient, it is important to recognize, however, that a gradient descent optimization has an inferior convergence rate when compared to the quadratic Newton method. In Fig. 7, we compare the two techniques and indeed find that CoVaR has a superior performance. Furthermore, note that CoVaR uses strictly more information than variance minimization: While the operator pool $\mathcal{P} = \{\mathcal{H}_a\}_{a=1}^r$ containing only the Hamiltonian terms is sufficient in Theorem 1, we significantly enlarge this pool such that $N_c \gg r$ and aim to find joint roots of this large number of covariances.

C. Comparison to subspace expansion

Subspace expansion [71] is a method for discovery of low-energy excited states starting from an estimated ground state $|\tilde{\psi}_G\rangle$. One then explores directions in Hilbert space by applying low-weight operators as excitations to the ground state. Typically, operator pools of Pauli operators are used to produce a new set of states $|\psi_k\rangle = \mathcal{P}_k |\tilde{\psi}_G\rangle$ for calculating the overlaps $H_{kj} = \langle \psi_k | \mathcal{H} | \psi_j \rangle$ and $S_{kj} = \langle \psi_k | \psi_j \rangle$. Diagonalizing H_{kj} then reveals better ground-state energies than that of $|\tilde{\psi}_G\rangle$. As opposed to CoVaR, subspace expansion cannot prepare the “good quality” representation of the eigenstates with a quantum computer but is rather limited to estimating their energies with a classical computer.

The connection to CoVaR is elucidated further in Appendix A 1, where we express covariance functions in terms of a set of quantum-state overlaps $\langle O_k, \mathcal{H} \rangle_\psi = \langle \phi_{O_k} | \phi_{\mathcal{H}} \rangle$, where we can define the (unnormalized) vectors $|\phi_A\rangle := (A - \langle A \rangle) |\psi\rangle$. The covariance matrix in Eq. (3) is, thus, a positive-semidefinite overlap matrix $[\mathbf{C}(\psi)]_{kl} = \langle \phi_{O_k} | \phi_{O_l} \rangle$. We can make a weak analogy to subspace expansion based on the following observation. Given covariance functions can be expressed as state overlaps, the method explores possible directions in Hilbert space via our operator pool that is beyond the capabilities of the *Ansatz*, and we gain information through a linearization as the Jacobian at what parameters the nearest eigenstate may be found via the vanishing overlaps. Similarly, subspace expansion also uses operators additional to the *Ansatz* to explore around the estimated ground state to extract low-energy excited states. As such, we may be able to use existing heuristics from subspace expansion for selecting problem-specific operator pools.

D. Comparison to Hessian optimization

A Hessian-based Newton-Raphson optimization [72] of the function $E(\theta)$ uses the update rule

$$\theta^{(t+1)} = \theta^{(t)} - \eta \mathbf{H}^{-1} \nabla E(\theta), \quad (14)$$

where \mathbf{H} is the Hessian matrix of second derivatives $\partial_n \partial_n E(\theta)$ which can be estimated on a quantum device using standard techniques such as the parameter-shift rule [31,73].

Let us compare our root finding with this Hessian optimization. A similarity may be that both methods use additional information to provide improved convergence over methods that use only gradient information. The Hessian obtains information about the local curvature of the $E(\theta)$ manifold which is a ‘‘classical’’ multivariate function, and its local curvature can even be accurately captured by analytical approximations [31]—with the use of a quantum computer. In contrast, CoVaR uses information from operator covariances of the variational state $|\psi(\theta)\rangle$ extracted from exploring directions in Hilbert space through a randomly chosen, large operator pool as detailed in Appendix A 1. In this sense, CoVaR is a quantum-aware method, as it uses not only the fact that $E(\theta)$ (and its derivatives) are efficiently calculable on a quantum computer, but also the relationship between $|\psi(\theta)\rangle$, the Hamiltonian, and possible directions in Hilbert space. The most pronounced difference between the two techniques is from a practical point of view: While we can use the classical shadow procedure ν times to estimate a very large Jacobian, for the Hessian we need to use it $\mathcal{O}(\nu^2)$ times at different circuit-parameter configurations. In this sense, CoVaR obtains significantly more information using fewer measurements.

It is also interesting to point out that the Gauss-Newton and LM techniques can be interpreted as approximate Hessian optimizations of the vector norm $\|\mathbf{f}\|^2$ when the Jacobian $\mathbf{H} \approx 2\mathbf{J}^\dagger \mathbf{J}$ gives a good approximation of the Hessian matrix by keeping only the first-order derivatives $[\partial_n f_k(\theta)][\partial_m f_k(\theta)]$ but neglecting second-order derivatives of the form $\partial_n \partial_m f_k(\theta)$. This, however, does not apply to CoVaR, given in our case the functions $f_k(\theta)$ are trigonometric polynomials for which the second-order derivatives are dominant, especially near solutions. Take, for example, the simple function $g(\theta) = -\prod_{n=1}^{\nu} \cos(\theta_n)$ which has a minimum at $\theta = (0, 0, \dots, 0)$ and, while its first derivatives vanish near the minimum, the second derivatives $|\partial_n \partial_n g(\theta)| \approx 1$ are dominant. Note also that the vector norm $\|\mathbf{f}\|^2$ has no immediate relation with the energy surface $E(\theta)$, and, thus, it is not related to a second-order energy minimization.

E. Comparison to natural gradient

Quantum natural gradient, which is equivalent to imaginary time evolution for ideal, unitary circuits, improves over both the convergence rate and ability to avoid traps of vanilla gradient descent by taking into account the geometry of the space of quantum states. Here, we estimate the quantum Fisher information [30,65–67] as

$$\mathbf{F}_{ij}(\underline{\theta}) = \langle \partial_i \psi | \partial_j \psi \rangle - \langle \partial_i \psi | \psi \rangle \langle \psi | \partial_j \psi \rangle, \quad (15)$$

which is equivalent to the real part of the quantum geometric tensor $\mathbf{F}(\underline{\theta}) = \text{Re}[\mathbf{F}_{ij}]$ and is used to compute the parameter update

$$\underline{\theta}_{t+1} = \underline{\theta}_t - \eta \mathbf{F}^+(\underline{\theta}_t) \nabla E(\underline{\theta}_t). \quad (16)$$

Similarly to Hessian optimization, this method has a complexity $\mathcal{O}(\nu^2)$ for calculating the tensor which, as opposed to the Hessian, is independent of the energy surface and rather expresses relations between states reached by varying different parameters.

CoVaR can also be thought of as an optimization using additional information about the space of quantum states but in this case extracted from the many covariance functions (which do depend on \mathcal{H}) instead of the geometric tensor.

F. Relation to parent Hamiltonians

Let us now relate CoVaR to existing techniques that do not aim to find eigenstates but rather aim to find Hamiltonians $\mathcal{H}_{\text{parent}}$ that encode a fixed state $|\psi\rangle$ as an eigenstate. The $\mathcal{H}_{\text{parent}}$ are then called as parent Hamiltonians to the state $|\psi\rangle$.

In particular, these techniques proceed by assuming that the parent Hamiltonian can be expressed in terms of the *Ansatz* as a linear combination of basis operators \mathcal{H}_a as [74,75]

$$\mathcal{H}_{\text{parent}} = \sum_{a=1}^r h_a \mathcal{H}_a, \quad (17)$$

via the real coefficient vector $\underline{h} \in \mathbb{R}^r$. Here, r is the rank of the decomposition, i.e., the number of independent basis operators. The covariance matrix $\mathbf{C}(\psi) \in \mathbb{C}^{r \times r}$ then depends on our trial quantum state $|\psi\rangle$ that we define in Eq. (3). The parent Hamiltonian is then found by finding the null space of this covariance matrix given every coefficient vector \underline{h} in the null space satisfies $\mathbf{C}\underline{h} = 0$ and given our expression for the variance in Lemma 1 guarantees that $\underline{h}^\dagger \mathbf{C} \underline{h} = \text{Var}[\mathcal{H}_{\text{parent}}] = 0$ in the particular state $|\psi\rangle$.

CoVaR clearly works according to a reverse logic whereby the problem Hamiltonian is fixed and we search for quantum states that result in 0 covariances. We then search the space of quantum states via an efficient parametrization, i.e., variational circuits, using a quantum computer.

VI. DISCUSSION

We demonstrate that CoVaR shows significantly improved performance by many orders of magnitude compared to analogous variational algorithms due to its

effective use of classical shadows. However, a main limitation is its vulnerability to random parameter initialization. Although this seemingly has a resemblance to barren plateaus whereby expected-value landscapes suffer from flat regions due to vanishing gradients [13,17,18], the present limitation is quite different in nature. In particular, recall that phase-estimation protocols provably efficiently find an eigenstate of an efficiently simulable Hamiltonian given an initial state is provided with a sufficiently large overlap with the desired eigenstate. Our approach is quite similar, as it gets attracted to any eigenstate with a significant contribution to the initial state. For this reason, we expect the main limitation of the present approach is not decoupled from the general challenge of finding good initialization for fault-tolerant phase-estimation protocols or finding problem specific *Ansätze* for VQE problems—and this challenge may be attributed to general hardness results of finding ground states or eigenstates [76]. Interestingly, we demonstrate in numerical simulations that even a short period of gradient descent evolution provides sufficient initialization in practice.

Furthermore, barren plateaus do not necessarily exist for our focus of local Hamiltonians and shallow *Ansätze* [17,18]. Nevertheless, random initialization of gradient-based VQE optimizers still prohibits finding eigenstates of large systems due to local traps [17]: First, optimizing VQAs has been shown to be *NP* hard due to persistent local minima [69]; second, Ref. [17] proves that a broad class of shallow VQA models that exhibit no barren plateaus are untrainable due to local traps. As we demonstrate in numerical simulations, our stochastic Levenberg-Marquardt approach indeed does mitigate the effect of these local traps: While gradient-based optimizers fail to make progress around a local trap as the gradient of the *energy surface* vanishes, CoVaR is not an energy minimizer, and those specific parameters may well yield a nonzero step for CoVaR. Furthermore, CoVaR is also less vulnerable to getting stuck due to our randomly generated constraints (covariances): Even if a single iteration makes no progress, in the next iteration a new, randomly generated set of constraints may well yield a nonzero step as we demonstrate in Fig. 13, where sometimes several steps of CoVaR are required to escape a trap. This is analogous to the well-known advantage of stochastic gradient descent in the machine-learning context [43], and we note that exploring globally convergent root-finding techniques may also be a fruitful direction for future research [37,39,40].

There are a number of apparent extensions to our approach that we do not consider here. First, given the classical shadows are stored in a classical computer, we can, in principle, determine multiple sets of update rules from them and apply the one that most decreases the variance or any other metric as opposed to our fully randomized scheme. Second, it would be worth exploring some specific use cases of CoVaR in more detail, such as

finding highly excited states. Third, in this work, we use classical shadows to extract a large number of covariances in the case where both the Hamiltonian and operator pool are constructed of local Pauli strings. It is an interesting direction for further work to attempt to use other randomized measurement channels to measure covariances with similar efficiency for nonlocal Hamiltonians or operator pools. Several works have appeared recently that make significant progress by developing shadow-measurement channels that interpolate between Pauli measurements (as in this work) and the powerful global Clifford measurements [77,78]. These intermediate-depth techniques are amenable to NISQ devices and allow for the measurement of nonlocal properties. As a matter of fact, related techniques leveraging simultaneous measurements of commuting observables are also highly relevant, given they allow the efficient reconstruction of a large number of not necessarily local Pauli strings as crucial in applications of quantum chemistry [79–82].

Finally, we expect the present approach to be resilient against reasonable levels of experimental noise. First, our update rule in Eq. (10) is invariant under global depolarizing noise when the expected value $\langle \mathcal{H} \rangle$ is known to high precision, e.g., via well-established error-mitigation techniques [46–49]. Second, we numerically simulate an approximate noise model that goes beyond global depolarization, and we observe a very good robustness against experimental noise. While these observations speak for the practicality of the present approach, we leave it to future work to confirm the performance in current and near-future generation hardware.

VII. CONCLUSION

In this work, we considered powerful variational quantum circuits that have been extensively investigated in a hope to exploit near-term quantum computers. Most of these near-term quantum algorithms aim to encode the solution to a practical problem of interest to eigenstates of a Hamiltonian, typically the ground state. As a direct analogy to successful variational techniques in quantum chemistry, nearly all quantum variational algorithms so far have proceeded by posing the problem as a variational search. In this paradigm, we minimize the single classical cost function—typically, the expected value of a Hamiltonian—with respect to circuit parameters.

Our work opens a new research direction in the efforts of achieving practical value with near-term quantum computers: We observe that the condition for finding eigenstates can be posed as finding joint roots of a large number of properties of the quantum state as covariance functions—these express fundamental quantum-mechanical uncertainty relations. We have devised the powerful root-finding technique CoVaR and demonstrated that increasing the number of these constraining covariances significantly increases the efficacy of the search procedure.

The most remarkable feature of CoVaR is that it allows us to fully exploit the extremely powerful classical shadow techniques in a way that prior variational techniques could not; i.e., we *simultaneously* estimate a very large number of randomly chosen properties, e.g., $> 10^4$ – 10^7 , of the quantum state and their derivatives with respect to circuit parameters. These inform our search procedure via a large but tractable linear system of equations that we solve with a classical computer.

Our approach can be viewed as directly analogous with (stochastic gradient descent and) stochastic Levenberg-Marquardt techniques that have been extremely popular in the context of classical machine learning—and we generally expect CoVaR inherits the fast convergence speed of Levenberg-Marquardt as we indeed demonstrated in practical examples. In fact, Levenberg-Marquardt is the default and fastest method for training classical neural networks [22–25]—but with a limitation that handling a large Jacobian becomes the bottleneck for too-deep neural networks. In stark contrast, we view this limitation of the classical technique a major advantage of our approach given we can populate the large Jacobian using only *logarithmic* quantum resources. CoVaR thus allows us to offload nontrivial but tractable calculations onto the classical computer and combines the best of both worlds, i.e., fast convergence and fast (quantum) computation of the Jacobian. Furthermore, as we demonstrated, using a large number of randomly generated constraints makes CoVaR particularly robust against getting stuck in local traps in analogy with stochastic techniques in the classical machine learning context [22,43].

We proved that the quantum resources, using classical shadows, required for a single iteration of our procedure are comparable to those of a standard gradient estimation in conventional VQE. In addition to its significantly improved convergence speed, our approach exhibits a robustness against shot noise and against experimental imperfections thanks to our large dataset. We have explored a number of practically motivated important applications whereby the problem Hamiltonian is local given the classical shadow procedure is very NISQ friendly in such scenarios, requiring only single-qubit measurements in a random basis. These include recompiling quantum circuits and finding ground and excited states. Our numerical simulations confirm the superiority of our approach and indicated that it can significantly outperform others by many orders of magnitude. Furthermore, previous techniques for finding excited states of Hamiltonians assumed a sequential search, whereas ours naturally converges to any of the eigenstates—and can, thus, be applied naturally in this context. Similarly, recompilation is another natural set of problems for CoVaR given any eigenstate of the problem Hamiltonian can be accepted as a solution. While the presented applications tackling local Hamiltonian problems are ideal for CoVaR, important quantum chemistry

problems may be nonlocal and may, thus, be challenging for classical shadows depending on the encoding. Fortunately, two fields of active research are making progress to alleviate this issue: First, compact fermion encodings result in local Hamiltonians at the cost of a modest qubit overhead, and, second, recent advancements in classical shadows allow for efficiently measuring nonlocal properties or, specifically, measuring fermionic operators [77,78,83,84]. It is, thus, expected that the present approach will be highly competitive and will spark further developments in the field.

Finally, our work makes exciting connections to various fields, including fundamental uncertainty relations in quantum mechanics as covariances, exploitation of classical shadows, stochastic optimization in machine learning, and working with big data. We believe it will be interesting to explore these connections to further improve the presented techniques in the hope of achieving practical value with near-term quantum computers.

ACKNOWLEDGMENTS

We thank Jonathan Foldager, Hsin-Yuan Huang, and Suguru Endo for providing us with useful comments. We thank Simon C. Benjamin for his support throughout this work. G. B. and B. K. acknowledge the EPSRC Hub grant under Agreement No. EP/T001062/1 for hardware provision. B. K. thanks the University of Oxford for a Glasstone Research Fellowship and Lady Margaret Hall, Oxford for a Research Fellowship. The numerical modeling involved in this study made use of the quantum exact simulation toolkit (QuEST) and the recent development QuESTlink [85], which permits the user to use *Mathematica* as the integrated front end. We are grateful to those who have contributed to both these valuable tools.

B. K. conceived the idea and contributed to writing the manuscript, G. B. performed numerical simulations and contributed to writing the manuscript.

APPENDIX A: ORTHOGONAL CONSTRAINTS

We present the general theory of our approach in Sec. II, whereby we compute covariances with respect to an arbitrary operator pool \mathcal{P} in order to search for eigenstates of an arbitrary Hamiltonian \mathcal{H} . While our practically motivated CoVaR approach leverages powerful classical shadows, it restricts the problem Hamiltonian and the operator pool to local Pauli strings. While Pauli strings are orthonormal in operator space, their actions on quantum states are generally not orthogonal directions in Hilbert space. In the present section, we explore another kind of operator pool \mathcal{P} whereby the operators represent orthogonal directions in state space. Let us first start by interpreting covariances as overlaps in Hilbert spaces.

1. Interpretation as state overlaps

Lemma 3.—The covariances from Definition 1 can be interpreted as overlaps between quantum states as

$$\langle A, B \rangle_\psi = \langle \phi_A | \phi_B \rangle, \quad (\text{A1})$$

where we can define the (unnormalized) vectors $|\phi_A\rangle := (A - \langle A \rangle)|\psi\rangle$ for any operator $A \in \mathbb{C}^{d \times d}$ with norm $\|\phi_A\|^2 = \text{Var}[A]$.

Proof.—The above property immediately follows from the defining expression of covariances from Definition 1. ■

The above lemma informs us that in Theorem 1 and in Corollary 3 we compute overlaps as $\langle O_k, \mathcal{H} \rangle_\psi = \langle \phi_k | \phi_{\mathcal{H}} \rangle$, and, thus, we actually decompose the quantum state $\mathcal{H}|\psi\rangle$ into a set of quantum states $|\phi_l\rangle$ that we obtain by acting on $|\psi\rangle$ with elements of our operator pool. Given that $|\phi_{\mathcal{H}}\rangle$ must be the null vector when the eigenvalue equation is satisfied, it is necessary that any (nonparallel) overlap with this vector must vanish.

While Pauli strings form an orthonormal basis of operator space, they have the disadvantage that in Hilbert space they result in nonorthogonal actions $\langle \phi_k | \phi_l \rangle \neq \delta_{kl}$; i.e., we decompose the vector $\mathcal{H}|\psi\rangle$ into a nonorthogonal basis. On the other hand, it is possible to define an operator pool that corresponds to orthogonal directions in Hilbert space.

Lemma 4 (orthogonal operator pool).—Let us consider strings of X Pauli operators as $X_k \in \{\text{Id}, X\}^{\otimes N}$ using the binary index $k \in \{0, 1\}^N$. We define the orthogonal operator pool $\mathcal{P} := \{O_k := UX_kU^\dagger, k \in \{0, 1\}^N\}$ via the operators, where the unitary quantum circuit U maps our reference $|\psi\rangle = U|0\rangle$ onto our quantum state. The quantum states $|\phi_k\rangle := O_k|\psi\rangle$ form an orthonormal system $\langle \phi_k | \phi_l \rangle = \delta_{kl}$, and, thus, the operators O_k map to orthogonal directions in Hilbert space.

Proof.—Orthonormality in Hilbert space follows from $\langle O_k | O_l \rangle = \delta_{kl}$ and via

$$\langle \psi | O_k O_l | \psi \rangle = \langle 0 | U^\dagger U X_k U^\dagger U X_l U^\dagger U | 0 \rangle = \langle k | l \rangle,$$

where $|k\rangle$ are standard basis vectors with $\langle k | l \rangle = \delta_{kl}$. ■

The resulting covariances can actually be shown to be entries in a column vector of the Hamiltonian matrix. In particular, given the states $|\phi_k\rangle$ form an orthonormal basis, they can be used to represent the Hamiltonian matrix as the covariances $f_k = \langle \psi | O_k \mathcal{H} | \psi \rangle = \langle \phi_k | \mathcal{H} | \phi_0 \rangle = \text{Col}_0(\mathcal{H})$, which are then actually elements of the first column vector of the problem Hamiltonian. These covariances, as entries of the Hamiltonian matrix, can be computed using the Hadamard-test techniques presented in Ref. [86]. In particular, the covariances are obtained by applying our variational quantum circuit U onto the standard computational basis states as $|\phi_k\rangle = UX_kU^\dagger|\psi\rangle = U|k\rangle$, and we compute the overlap of this state with our variational quantum state $|\psi\rangle$. The approach can straightforwardly

be implemented via the Hadamard test, whereby we apply the X_k operations in $|k\rangle = X_k|0\rangle$ controlled on an ancilla qubit. Derivatives of these covariances can be similarly computed by applying the generator of the quantum gate controlled on the same ancilla. Let us now show that sum of squares of the covariances is equivalent to the variance of the Hamiltonian.

Lemma 5.—Given the orthogonal operators introduced in Lemma 4, we compute the corresponding covariances. While $f_0 = \langle \psi | \mathcal{H} | \psi \rangle$ is the energy expected value, we can show that

$$\sum_{k=1}^{2^N-1} |f_k|^2 = \|f\|^2 = \text{Var}[\mathcal{H}]. \quad (\text{A2})$$

Proof.—

$$\sum_{k=1}^{2^N-1} |f_k|^2 = \sum_{k=0}^{2^N-1} \langle \psi | \mathcal{H} | \phi_k \rangle \langle \phi_k | \mathcal{H} | \psi \rangle - \langle \psi | \mathcal{H} | \psi \rangle^2.$$

As the $|\phi_k\rangle$ are a complete basis set due to being a unitary transformation of the computational basis, $\sum_k |\phi_k\rangle \langle \phi_k| = \text{Id}$, and we, therefore, obtain

$$\|f\|^2 = \langle \psi | \mathcal{H}^2 | \psi \rangle - \langle \psi | \mathcal{H} | \psi \rangle^2 = \text{Var}[\mathcal{H}].$$

■

Importantly, while this operator pool has the advantage that the covariances represent independent, orthogonal directions in state space, it is clear that we generally need to compute all $2^N - 1$ of these orthogonal constraints, as elements of the first column of the Hamiltonian matrix, in order to be able to compute the variance and, thus, to verify that the quantum state $|\psi\rangle$ is an eigenstate of the Hamiltonian. In stark contrast, in Theorem 1, we show that, given a decomposition of a Hamiltonian into an operator basis $\mathcal{H} = \sum_{k=1}^r h_a \mathcal{H}_a$ which typically grows polynomially with the system size, it suffices to compute only the corresponding polynomial number of covariances. Although these operators, such as Pauli strings, are orthonormal in operator space, they do not correspond to orthogonal directions in Hilbert space.

2. Finding eigenstates via orthogonal operator pools

Let us now apply our orthogonal constraints to finding eigenstates by finding roots. It is interesting to note that it follows from our relation in Lemma 5 that the Newton step through the inverse Jacobian from Eq. (10) as $\mathbf{J}^{-1}\underline{f}$ is guaranteed to represent a descent direction for the variance $\text{Var}[\mathcal{H}]$ given that the gradient vector $\text{grad}(\|f\|^2) = \mathbf{J}^\top \mathbf{f} = \text{grad}(\text{Var}[\mathcal{H}])$.

Furthermore, we write our problem as a least-squares minimization of the constraints f_k , and, thus, the Gauss-Newton and the Levenberg-Marquardt approaches can be

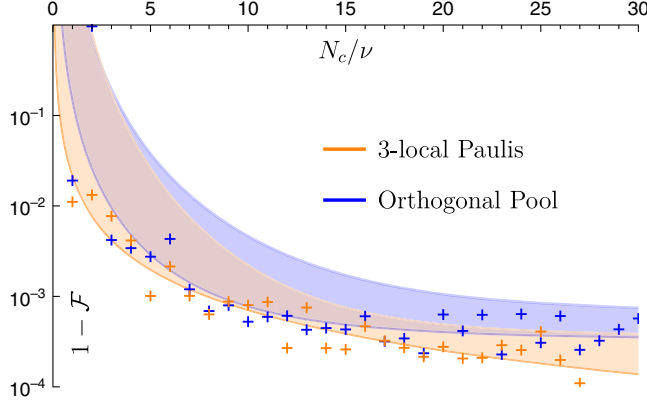


FIG. 10. Performance of recompiling the parameters of a 2-layer unitary on 14 qubits. Plot shows the minimum infidelity reached over 3 runs of 20 iterations of root finding with the ratio of constraints to ansatz parameters ν , for two different choices of operator pool with shot noise equivalent to taking 10^5 shots. Blue for orthogonal pool and orange for the set of all 3-local Pauli strings. Fits shown are of the form $1 - \mathcal{F}_{\min} = a(N_c/\nu)^{-b} + c$.

interpreted straightforwardly: Our root-finding approach is equivalent to a nonlinear least-squares minimization.

An advantage of this scheme is that we can randomly sample the constraints according to an importance sampling; i.e., the constraints are picked with a probability proportional to their magnitude $p_k = |f_k|^2$. We can efficiently upper bound these probabilities in an experiment as $p_k = \langle \phi_k | \mathcal{H} | \psi \rangle = |\langle k | U^\dagger \mathcal{H}_a | \psi \rangle|^2$. In particular, we run the quantum circuit $U^\dagger \mathcal{H}_a U$ and measure samples in the standard basis, whereby we obtain the binary string k with probability $|\langle k | U^\dagger \mathcal{H}_a | \psi \rangle|^2$. It then follows that $p_k \leq \sum_a c_a |\langle k | U^\dagger \mathcal{H}_a | \psi \rangle|^2$. There is, of course, no guarantee that these probabilities have structure; however, when performing energy minimization first, the probabilities are more likely to be peaked around the lower-energy basis vectors $|\phi_k\rangle$.

In case of finding eigenstates of a set of mutually commuting Hamiltonians \mathcal{H}_a , we can compute covariances as $\langle \phi_k | \mathcal{H}_a | \psi \rangle$ individually for all operators \mathcal{H}_a . If all such variances vanish, then we are guaranteed that $\text{Var}[\mathcal{H}_a] = 0$ for all a . In Fig. 10, we compare the performance of root finding for two choices of operator pool on a parameter rediscovery problem (3-local Pauli strings and the orthogonal operator pool), showing that both pools give very similar performance.

APPENDIX B: PROPERTIES OF COVARIANCES

1. Proof of Lemma 2: Smooth covariance functions

Let us prove that the covariance functions are smooth functions of the parameters $\underline{\theta}$ of the variational quantum state $|\psi(\underline{\theta})\rangle$. In particular, let us expand our expression from Eq. (6) as

$$\begin{aligned} f_k(\underline{\theta}) &= \langle \psi(\underline{\theta}) | O_k \mathcal{H} | \psi(\underline{\theta}) \rangle - \langle \psi(\underline{\theta}) | O_k | \psi(\underline{\theta}) \rangle \langle \psi(\underline{\theta}) | \mathcal{H} | \psi(\underline{\theta}) \rangle \\ &= A_{\text{re}}(\underline{\theta}) + iA_{\text{im}}(\underline{\theta}) - B(\underline{\theta})C(\underline{\theta}), \end{aligned} \quad (\text{B1})$$

which we rewrite in terms of four expectation values of four different Hermitian operators via using the real and imaginary parts as in Eq. (4) as

$$\begin{aligned} A_{\text{Re}} &:= \left\langle \frac{1}{2} \{O_k, \mathcal{H}\} \right\rangle, & A_{\text{Im}} &:= \left\langle -\frac{i}{2} [O_k, \mathcal{H}] \right\rangle, \\ B &:= \langle O_k \rangle, & C &:= \langle \mathcal{H} \rangle, \end{aligned} \quad (\text{B2})$$

where we use $\langle \cdot \rangle$ to denote the expected value with respect to the parametrized quantum state $\psi(\underline{\theta})$ and we drop the dependence on $\underline{\theta}$ for ease of notation. Indeed, above in Eq. (B1), all terms are expected values of Hermitian operators.

It suffices to show that the expectation value of any Hermitian observable $O \in \mathbb{C}^{d \times d}$ is a smooth function of the circuit parameters as

$$\langle O \rangle(\underline{\theta}) = \langle \psi(\underline{\theta}) | O | \psi(\underline{\theta}) \rangle = \langle 0 | U^\dagger(\underline{\theta}) O U(\underline{\theta}) | 0 \rangle. \quad (\text{B3})$$

Indeed, the *Ansatz* circuit is by definition [via Eq. (5)] a smooth mapping as $U(\underline{\theta}) \in \text{SU}(2^N)$, and, thus, $\langle O \rangle(\underline{\theta})$ is a smooth function of the parameters for any O .

2. Proof of Corollary 2: Trigonometric polynomials

Given the specific but pivotal scenario when every parametrized gate in the *Ansatz* circuit is a Pauli gate, as in the present work as illustrated in Fig. 11, Ref. [31] establishes the following. The expected value of any Hermitian observable is a trigonometric polynomial of the form

$$\langle O \rangle(\underline{\theta}) = \sum_{j=1}^{3^\nu} c_j T_j(\underline{\theta}),$$

where $c_j \in \mathbb{R}$ are real coefficients that depend on O and $T_j(\underline{\theta})$ are trigonometric monomials as products $T_j(\underline{\theta}) \in \prod_{n=1}^\nu \{a(\theta_n), b(\theta_n), c(\theta_n)\}$ of single-variate trigonometric functions [87], for example, $b(\theta_1) \prod_{n=2}^\nu a(\theta_n)$.

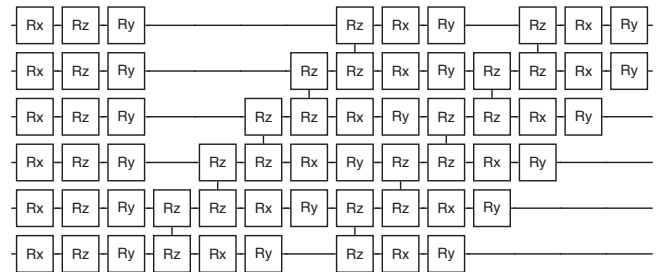


FIG. 11. One layer of the hardware-efficient *Ansatz* used in numerics for six qubits.

The single-variate functions are $a(\theta) := (1 + \cos[\theta])/2$, $b(\theta) := \sin[\theta]/2$, and $c(\theta) := (1 - \cos[\theta])/2$. Given in Eq. (B2) all four terms must be of this form, we obtain the expression for the covariances via Eq. (B1) as

$$f_k(\boldsymbol{\theta}) = \sum_{j=1}^{3^v} \tilde{c}_j T_j(\boldsymbol{\theta}) - \sum_{j,l=1}^{3^v} c'_j c''_l T_j(\boldsymbol{\theta}) T_l(\boldsymbol{\theta}) = \sum_{j=1}^{3^{2v}} C_j \mathcal{T}_j(\boldsymbol{\theta}),$$

where $\tilde{c}_j \in \mathbb{C}$ and $c'_j, c''_j \in \mathbb{R}$ are coefficients that depend on the index k . Indeed, here, $T_j(\boldsymbol{\theta}) T_l(\boldsymbol{\theta})$ are also trigonometric monomials, and in the last equation we denote these as $\mathcal{T}_j \in [\prod_{n=1}^v \{a(\theta_n), b(\theta_n), c(\theta_n)\}]^2$ with prefactors $C_j \in \mathbb{C}$.

3. Experimentally estimating covariances

Let us first compute covariances assuming the Hamiltonian $\mathcal{H} = \sum_{a=1}^r h_a \mathcal{H}_a$ is given in terms of Pauli strings $\mathcal{H}_a \in \{\text{Id}_2, X, Y, Z\}^{\otimes N}$. The covariances $f_k = \langle O_k, \mathcal{H} \rangle_\psi = A_{\text{re}} + iA_{\text{im}} - BC$ are completely determined by the expected values from Eq. (B2) of Hermitian operators. We can significantly simplify these when O_k and \mathcal{H}_a are Pauli strings as

$$A_{\text{re}} = \sum_{a=1}^r h_a \left\langle \frac{1}{2} \{O_k, \mathcal{H}_a\} \right\rangle = \sum_{a=1}^r h_a \langle P_{ka} \rangle, \quad (\text{B4})$$

$$A_{\text{im}} = \sum_{a=1}^r h_a \left\langle -\frac{i}{2} [O_k, \mathcal{H}_a] \right\rangle = \sum_{a=1}^r h_a \langle Q_{ka} \rangle, \quad (\text{B5})$$

where indeed $P_{ka}, Q_{ka} \in \pm\{\text{Id}_2, X, Y, Z\}^{\otimes N}$ are Hermitian Pauli strings given any two Pauli strings O_k and \mathcal{H}_a either commute or anticommute and, thus,

$$\begin{aligned} & \frac{1}{2} \{O_k, \mathcal{H}_a\} \\ &= \begin{cases} 0, & \text{if } O_k \text{ and } \mathcal{H}_a \text{ anticommute,} \\ P_{ka} \in \pm\{\text{Id}_2, X, Y, Z\}^{\otimes N} & \text{otherwise,} \end{cases} \end{aligned}$$

and similarly for the imaginary part

$$\begin{aligned} & \frac{i}{2} [O_k, \mathcal{H}_a] \\ &= \begin{cases} 0, & \text{if } O_k \text{ and } \mathcal{H}_a \text{ commute,} \\ Q_{ka} \in \pm\{\text{Id}_2, X, Y, Z\}^{\otimes N}, & \text{otherwise.} \end{cases} \end{aligned}$$

Here, the particular Pauli strings P_{ka} and Q_{ka} and their signs \pm can be determined straightforwardly and efficiently from the indexes $k, a \in \{0, 1, 2, 3\}^N$ using the algebra of Pauli matrices, i.e., the Pauli group. We, therefore, conclude that the covariances can be computed in terms of only expected values of Hermitian Pauli strings as

$$f_k = \sum_{a=1}^r h_a (\langle P_{ka} \rangle + i \langle Q_{ka} \rangle) + \langle \mathcal{H}_a \rangle \langle O_k \rangle. \quad (\text{B6})$$

We need to estimate overall $3r$ expected values of Pauli strings to estimate a covariance f_k given $P_{ka} = 0$ when $Q_{ka} \neq 0$ and vice versa.

APPENDIX C: PROPERTIES AND APPLICATIONS OF THE JACOBIAN

1. Computing the Jacobian

We consider the covariances $f_k(\boldsymbol{\theta})$ with respect to Pauli strings $O_k \in \mathcal{P}$ in our operator pool as defined in Eq. (6) for a fixed Hamiltonian $\mathcal{H} = \sum_{a=1}^r h_a \mathcal{H}_a$. Recall that the Jacobian is defined in terms of the partial derivatives $\mathbf{J}_{kn} := \partial_n f_k(\boldsymbol{\theta})$. We can explicitly compute these derivatives by recalling that the covariances can be expressed in terms of expected values of Pauli strings via Eq. (B6) as

$$\begin{aligned} \mathbf{J}_{kn} &= \sum_{a=1}^r h_a \left(\frac{\partial \langle P_{ka} \rangle}{\partial \theta_n} + i \frac{\partial \langle Q_{ka} \rangle}{\partial \theta_n} \right. \\ &\quad \left. + \langle O_k \rangle \frac{\partial \langle \mathcal{H}_a \rangle}{\partial \theta_n} + \langle \mathcal{H}_a \rangle \frac{\partial \langle O_k \rangle}{\partial \theta_n} \right), \quad (\text{C1}) \end{aligned}$$

where $P_{ka}, Q_{ka}, \mathcal{H}_a$, and O_k are Pauli strings. Above, we can use well-established techniques for experimentally estimating derivatives of general expected values for a variety of *Ansatz* constructions and gate sets [9–11]. Furthermore, in Sec. III, we focus on the typical practical scenario when the *Ansatz* circuit consists of Pauli gates, and, thus, we can use parameter-shift rules [55] for computing derivatives—while generalizations in Refs. [88,89] are also applicable— as linear combinations of two expected values as

$$\frac{\partial \langle O \rangle(\boldsymbol{\theta})}{\partial \theta_n} = \frac{1}{2} \langle O \rangle(\boldsymbol{\theta} + \underline{v}_n \pi/2) - \frac{1}{2} \langle O \rangle(\boldsymbol{\theta} - \underline{v}_n \pi/2),$$

for any Hermitian observable O , where \underline{v}_n is the n th standard Euclidean basis vector. As such, we can compute all derivatives in Eq. (C1) by applying parameter-shift rules.

2. Computing the Jacobian using classical shadows

Let us now describe an explicit measurement protocol in the specific case when our Hamiltonian is local and the operator pool $\mathcal{P} = \{O_k\}_{k=1}^{N_c}$ is also local as in Statement 1, and, thus, we can use classical shadows to determine a large number M of local Pauli-string expected values. We then compute the large Jacobian by estimating Pauli strings at different shifted circuit parameters via the parameter-shift rules for computing partial derivatives.

Lemma 6.—Given an operator pool $\mathcal{P} = \{O_k\}_{k=1}^{N_c}$ and a problem Hamiltonian $\mathcal{H} = \sum_{a=1}^r h_a \mathcal{H}_a$ in terms of local Pauli strings O_k and \mathcal{H}_a and a variational circuit with ν parametrized Pauli gates, we can determine the Jacobian

$\mathbf{J} \in \mathbb{C}^{N_c \times \nu}$ by applying the classical shadow procedure $2\nu + 1$ times at different circuit-parameter configurations with each estimating $M \leq 3rN_c$ Pauli expected values. In contrast, determining a gradient vector for gradient descent requires 2ν applications of the classical shadow procedure each with $M = r$.

Proof.—First, at parameters $\underline{\theta}$, we determine overall $M = r + N_c$ expected values as $\langle O_k \rangle$ and $\langle \mathcal{H}_a \rangle$ from Eq. (C1) using a single application of the classical shadow procedure. Second, we estimate derivatives of expected values ($\partial \langle \mathcal{H}_a \rangle / \partial \theta_n$, $\partial \langle O_k \rangle / \partial \theta_n$, and either $\partial \langle P_{ka} \rangle / \partial \theta_n$ or $i(\partial \langle Q_{ka} \rangle / \partial \theta_n)$) via the above parameter-shift rule. We determine all derivatives with respect to a fixed parameter θ_n by estimating Pauli strings at parameters $\underline{\theta} + \underline{v}_n \pi / 2$ as well as at parameters $\underline{\theta} - \underline{v}_n \pi / 2$ by two applications of the classical shadow procedure each with $M \leq r + N_c + rN_c$. Thus, determining all derivatives requires 2ν applications of the classical shadow. ■

3. Computing the variance gradient from the Jacobian

The gradient of the variance can be computed via Lemma 1 when the operator pool is $\mathcal{P} = \{\mathcal{H}_a\}_{a=1}^r$ as

$$\partial_n \sigma^2 = \partial_n \left[\sum_{a=1}^r h_a f_a \right] = \sum_{k=1}^r h_a \partial_n f_a(\psi) = [\mathbf{J}]_{an} h_a = \mathbf{J}^\top \underline{h}.$$

This confirms that the gradient of the variance is determined by our Jacobian.

We can also compute the gradient of the vector norm $\|f\|^2 = \sum_{k=1}^{r_p} |f_k|^2$. When the operator pool is larger than the problem Hamiltonian terms, then we compute the gradient of the norm of the covariance vector

$$\begin{aligned} \partial_n \|f\|^2 &= \partial_n \sum_{k=1}^{r_p} f_k f_k^* = \sum_{k=1}^{r_p} [(\partial_n f_k) f_k^* + f_k (\partial_n f_k^*)] \\ &= \sum_{k=1}^{r_p} [\mathbf{J}_{kn} f_k^* + f_k \mathbf{J}_{kn}^*] = 2\text{Re}[\mathbf{J}^\top \underline{f}^*] = 2\tilde{\mathbf{J}}^\top \tilde{\mathbf{f}}, \end{aligned}$$

where in $\tilde{\mathbf{J}}$ and in $\tilde{\mathbf{f}}$ we stack real and imaginary parts on top of each other.

4. Computing the update rule classically

We need to first stack real and imaginary parts on top to solve for only real parameter updates. In particular, we want to solve the linear system of equations $\mathbf{J}\Delta\theta = \underline{f}$. Here, we estimate both \mathbf{J} and \underline{f} with a quantum computer, and we want to compute the parameter update $\Delta\theta$. If we merely compute the pseudoinverse of \mathbf{J} and apply it as $\mathbf{J}^{-1}\underline{f}$, then the resulting solution vector $\Delta\theta$ is generally complex. However, we require that our parameter update be real as $\Delta\theta \in \mathbb{R}$, which we enforce via the following set of linear equations:

$$\text{Re}[\mathbf{J}]\Delta\theta = \text{Re}[\underline{f}] \quad \text{and} \quad \text{Im}[\mathbf{J}]\Delta\theta = \text{Im}[\underline{f}].$$

We can simultaneously solve these systems of equations by stacking real and imaginary parts $\tilde{\mathbf{J}} = (\text{Re}[\mathbf{J}], \text{Im}[\mathbf{J}])^\top$ and $\tilde{\underline{f}} = (\text{Re}[\underline{f}], \text{Im}[\underline{f}])^\top$ on top of each other, and this guarantees that the solution is real $\tilde{\mathbf{J}}^{-1}\tilde{\underline{f}} \in \mathbb{R}$, since $\tilde{\mathbf{J}}$ and $\tilde{\underline{f}}$ are real matrices and vectors, respectively.

We invert the Jacobian via the usual damped, regularized inverse

$$\tilde{\mathbf{J}}^{-1} := [\tilde{\mathbf{J}}^\top \tilde{\mathbf{J}} + \lambda \text{Id}]^{-1} \tilde{\mathbf{J}}^\top.$$

Here, λ is a regularization parameter that we dynamically set by choosing $\lambda = 0.0001 \times 2^i$, where i is incremented from 0 until the condition $\|\underline{f}(\underline{\theta}_i)\| < \|\underline{f}(\underline{\theta}_{i-1})\|$ is met; i.e., the norm of the covariance vector decreases from the previous iteration. If the step $\Delta\theta$ to be taken for any individual parameter is too large $|\Delta\theta_i| > 1$, the update step is rescaled to normalize this value to 1. This prevents our algorithm from taking overly large steps. For the selection of constraints, N_c constraints are selected randomly from the chosen operator pool at every iteration.

We implement a line search algorithm whereby we compute the update rule $\Delta\theta$ and compute the value of the vector norm $\|\underline{f}(\underline{\theta})\|$ at parameter values $\underline{\theta} = \underline{\theta}_t + \kappa \Delta\theta$ in small increments in κ . However, it is observed empirically that a step close to the canonical choice $\kappa = -1$ is almost always chosen, so line search is not used in the numerics in this work.

5. Existence of a shot-noise floor

We extend our example in Sec. III A and consider noisy entries of the Jacobian and the covariance vector as $J_k + \epsilon_k$ and $f_k(\theta) + \eta_k$ for random variables ϵ_k and η_k due to shot noise (and possibly other sources of random errors). The solution to our linear equation in Eq. (11) is, thus, modified as

$$\Delta\theta \approx -\frac{\sum_k J_k f_k(\theta)}{\sum_k J_k^2} - \frac{\sum_k J_k \eta_k + f_k(\theta) \epsilon_k}{\sum_k J_k^2},$$

where we drop all higher-order terms as products $\epsilon_k \eta_k$ as well as consider the approximation $[\sum_k J_k^2 + \mathcal{O}(\epsilon_k)]^{-1} = [\sum_k J_k^2]^{-1} + \mathcal{O}(\epsilon_k / [\sum_k J_k^2]^2)$, where we also drop the term $\mathcal{O}(\epsilon_k / [\sum_k J_k^2]^2)$.

Let us now consider the error propagation to the solution $\Delta\theta$ by considering the linear error propagation formula

$$\begin{aligned} \text{Var}[\Delta\theta] &\approx \frac{\sum_k J_k^2 \text{Var}[\eta_k] + f_k(\theta)^2 \text{Var}[\epsilon_k]}{\sum_k J_k^2} \\ &\leq \mathcal{E}^2 \left[1 + \frac{|f_k(\theta)|^2}{\sum_k J_k^2} \right], \end{aligned}$$

where we simplify the formula by bounding the variance of each $\text{Var}[\eta_k] \leq \mathcal{E}^2$ and $\text{Var}[e_k] \leq \mathcal{E}^2$; i.e., this is indeed the case in practice where we determine each covariance and Jacobian entry to the same guaranteed precision \mathcal{E} using classical shadows. Indeed, we see that, as we approach an eigenstate, $|f_k(\theta)|^2 \rightarrow 0$, and, thus, the error that propagates into our solution is bounded by $\text{Var}[\Delta\theta] \leq \mathcal{E}^2$ by the worst-case error of a *single Jacobian or vector entry*. This is a very powerful averaging of random errors: By increasing the number of constraints, we gain increasingly more information about the root; however, the random error that propagates into our solution does not scale with the number of constraints.

6. Time complexity of classically solving the linear system of equations

Let us now derive the time complexity of computing the regularized inverse as $(\tilde{\mathbf{J}}\tilde{\mathbf{J}} + \lambda R)^{-1}\tilde{\mathbf{J}}\tilde{\mathbf{f}}$ of the Jacobian $\tilde{\mathbf{J}} \in \mathbb{R}^{2N_c \times \nu}$ in which we stack real and imaginary parts on top of each other and R is a regularization matrix. This computation can be broken up into four steps.

- (i) First, compute the square matrix $A := \tilde{\mathbf{J}}\tilde{\mathbf{J}} + \lambda R$ as the product of two nonsquare matrices as $[A]_{mn} = \lambda R_{mn} + \sum_{k=1}^{2N_c} [\tilde{\mathbf{J}}]_{mk} [\tilde{\mathbf{J}}]_{kn}$. Computing all ν^2 entries of A requires $2\nu^2 N_c + \nu^2$ operations.
- (ii) Second, we compute the inverse of A which requires between $\mathcal{O}(\nu^{2.373})$ and $\mathcal{O}(\nu^3)$ operations, depending on the algorithm.
- (iii) Third, compute the matrix-vector product $\mathbf{v} := \tilde{\mathbf{J}}\tilde{\mathbf{f}}$ as $[\tilde{\mathbf{J}}\tilde{\mathbf{f}}]_n = \sum_{k=1}^{2N_c} [\tilde{\mathbf{J}}]_{nk} \tilde{f}_k$, which requires overall $2\nu N_c$ operations.
- (iv) Finally, we compute the matrix-vector product $A\mathbf{v}$, which requires ν^2 operations.

Given $N_c \gg \nu$, the overall computation time is dominated by the first step as computing A , and, thus, the time complexity is $t \in \mathcal{O}(\nu^2 N_c)$, which is merely linear in the dominant dimension N_c . We confirm this theoretical scaling in Fig. 5 and conclude that the absolute times are very reasonable; i.e., we can compute the update rule in a matter of minutes for up to a very large number of covariances $N_c = 10^6$. Of course, the computation can be heavily parallelized and can also be performed in a distributed-memory model with negligible communication between nodes. As such, in principle, one could straightforwardly use a very large number of covariances $N_c \approx 10^8$ that would still require reasonable classical computational resources.

APPENDIX D: FURTHER NUMERICAL SIMULATIONS

1. Comparison with VQE for the spin-chain problem

While in Sec. IV B we focus on practical applications of CoVaR, here we demonstrate a comparison of convergence

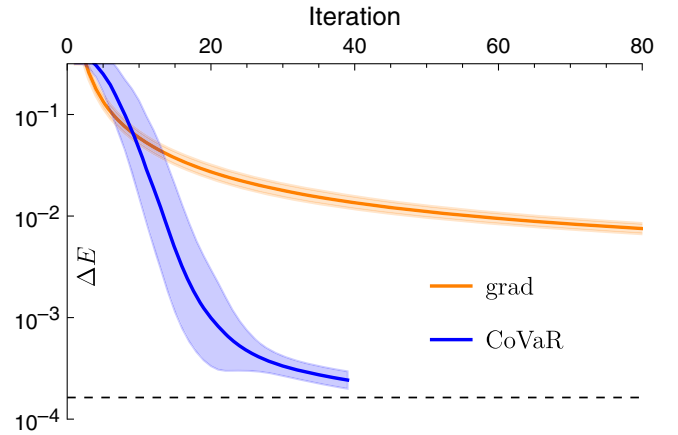


FIG. 12. Comparison of the performance of CoVaR to gradient descent when starting from a point close to the ground state of the ten-qubit spin chain, analogous to Fig. 7(c) for recompilation. This initialization is done by randomly perturbing the parameters of a state close to the ground state. Average initial fidelity to the ground state is $\mathcal{F} = 80 \pm 4\%$. Mean and standard deviation are shown for 20 runs of gradient descent and the 16/20 runs of CoVaR that converge to the ground state rather than one of the excited states. The *Ansatz* can only approximate the ground state of the spin model; the grid line at $\Delta E \approx 1.6 \times 10^{-4}$ marks the minimum achieved energy as the limit of precision. One can further improve ΔE by only increasing the *Ansatz* depth.

speed between gradient descent and CoVaR. In particular, in Fig. 12, we compare the speed of convergence to the ground state of the spin chain, using an identical *Ansatz* and hyperparameters as those used for runs of CoVaR in Fig. 8.

2. Demonstration of escaping local traps

As we note in the main text, even when gradient descent is stuck in a local trap due to the gradient of the energy surface vanishing, CoVaR can be used to navigate out from such a trap, given CoVaR is not an energy minimizer and may yield a nonzero step—especially that constraints that determine a CoVaR step are generated randomly at every iteration. We numerically demonstrate this in Fig. 13 using our spin-chain problem, with all parameters the same as those used for Fig. 8. The local traps are found using 1000 iterations of gradient descent with an adaptive step size and achieve a mean energy difference with the ground state of $\Delta E = 0.019$. CoVaR is then run for 100 further iterations and is able to achieve a mean energy difference of $\Delta E = 1.8 \times 10^{-4}$.

3. Convergence of root finding to eigenstates

In this section, we further analyze the property of CoVaR that it converges to eigenstates that have a dominant contribution to the initial state as reported in Fig. 9. For this reason, in Fig. 14, we perform simulations of CoVaR optimizations with initial states of increasing expected energy starting near the ground state.

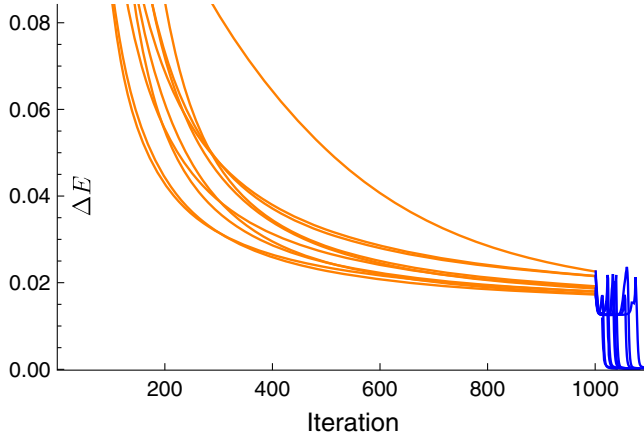


FIG. 13. Ten examples of CoVaR escaping from practical local traps demonstrated on the same ten-qubit spin-chain problem as in Fig. 8. Gradient descent (orange) for 1000 adaptive size steps to reach local minima, followed by 100 iterations of CoVaR (blue). Several steps of CoVaR are sometimes required before finding a set of operators that produce a step that escapes the minimum. Mean energy differences from the ground state are $\Delta E_{\text{localmin}} = 0.019$ and $\Delta E_{\text{CoVaR}} = 1.8 \times 10^{-4}$. The last step of each run of gradient descent has an energy improvement below 2×10^{-5} , indicating being stuck in a local trap. Note that a single iteration of CoVaR does not necessarily decrease the energy, given CoVaR is not an energy minimizer as is demonstrated here but is also visible in Fig. 8. Indeed, escaping from a local trap of the *energy* surface with CoVaR may result in an intermediate increase of the energy.

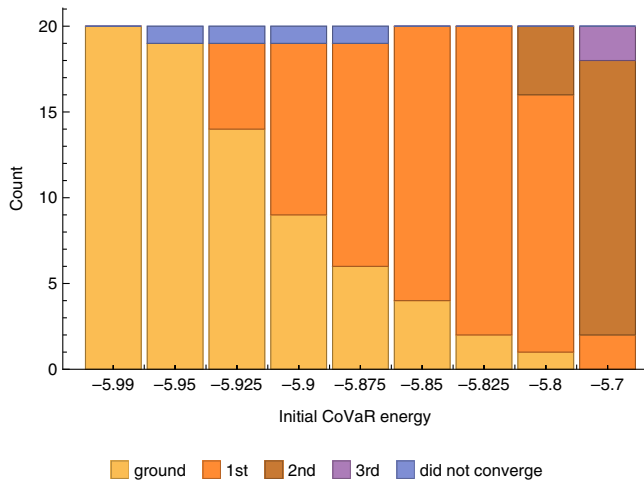


FIG. 14. Bar chart showing the distributions of states which CoVaR converges to for the runs in Fig. 9 showing the probability of converging to the ground state with initial overlap (convergence being an energy difference of less than 10^{-3} in this case) as a function of its initialization energy. Each bar corresponds to one point in Fig. 9, but there are two additional points on the right of the figure.

Recall that the expected value of the energy can be written as $E = \langle \psi | \mathcal{H} | \psi \rangle = \sum_k E_k p_k$, where E_k are eigenenergies (eigenvalues) of \mathcal{H} and p_k are the probabilities (fidelity) that $|\psi\rangle$ is in the k th eigenstate of \mathcal{H} . Given a gapped Hamiltonian with energies $E_0 < E_1 < E_2 \dots$, quantum states with a low energy must necessarily have a high probability (fidelity) to be in the lowest-lying eigenstates. As such, an initial state that has expected energy $E = E_0 p_0 + E_1 p_1 + \dots$ close to the ground-state energy as $E \approx E_0$ guarantees the high probability (fidelity) $p_0 \approx 1$. Indeed, CoVaR nearly always converges to the ground state in Fig. 14 (bars on the left) when $E \approx E_0$ and, thus, $p_0 \approx 1$.

On the other hand, starting CoVaR from initial states that have higher expected energies than E_0 causes CoVaR to not always converge to the ground state due to the necessarily increased populations of excited states: Fig. 14 (bars in the middle) features instances where CoVaR does not converge to the ground state; however, it then nearly always converges to the first excited state. For example, when no higher eigenstates are populated, the probability of the first excited state is $p_1 = 1 - p_0$. Interestingly, we find a nearly linear relationship between the probability of the eigenstate, such as p_0 , and the percentage when CoVaR converges to that state in Fig. 9—and note that this relationship would be exactly linear for the case of fault-tolerant phase-estimation protocols. As we keep increasing the energy, higher excited states start to contribute as in Fig. 14 (bars on the right), and, thus, CoVaR may also converge to those eigenstates.

4. Scaling of performance

We perform simulations to assess the scaling of the performance of CoVaR, for both recompilation and spin-chain problems, and plot the results in Fig. 15. For the recompilation problem, we know by construction the parameters $\underline{\theta}^*$ of the ground state and the circuit depth is constant due to a fixed number (two) of *Ansatz* layers for all qubit counts N . For the spin-chain problem, we first search for the solution at each qubit number N using natural gradient descent followed by CoVaR. The number of *Ansatz* layers is then increased until a desired precision with respect to the ground state is achieved (here, an infidelity of 10^{-5}), thus obtaining a series of solution parameters $\underline{\theta}^*$ at every qubit count.

These solution parameters are perturbed to obtain initial states for CoVaR with a desired initial overlap with the ground state (50% for recompilation and 80% for the spin chain to suppress convergence to excited states). Of course, such states could equally well have been created through alternative initialization methods, including performing an initial period of gradient or natural gradient descent. Finally, CoVaR is run for a fixed number 20 of iterations and statistics of the final achieved fidelity are plotted in Fig. 15.

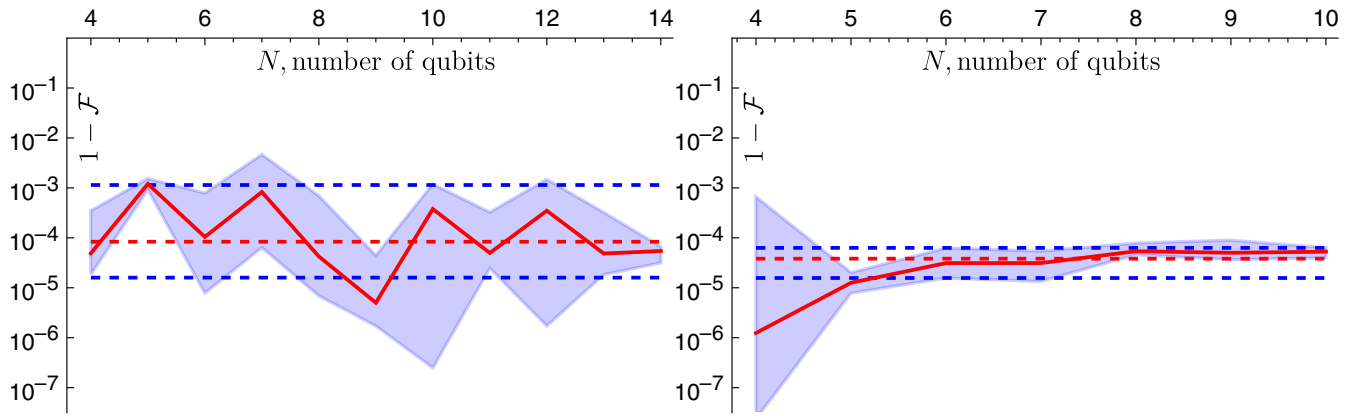


FIG. 15. Scaling results showing the infidelity with respect to the ground state of the state achieved after 20 steps of CoVaR from an initial state of $\mathcal{F}_{\text{init}}$ overlap with the ground state. The red curve and blue shaded region show the median and quartiles of ten runs of CoVaR for each N , and the dotted lines show the median and quartiles of the data as a whole. Left: simulations for our recompilation problem from Sec. IV A using a constant initial fidelity $\mathcal{F}_{\text{init}} = 50\%$ and a constant *Ansatz* depth of two layers. Right: simulations for the spin-chain problem using an increasing *Ansatz* depth that can estimate the ground state to at least an infidelity of 10^{-5} . An initial overlap of $\mathcal{F}_{\text{init}} = 80\%$ is used to suppress convergence to excited states, and only the runs that do converge to the ground state (and not to an excited state) are included in the present statistics. Compared to our recompilation simulations (left) using a constant *Ansatz* depth, here the performance may appear to be decreasing slightly as we increase the number of qubits—which would be explained by the increasing depth (polylogarithmically in Sec. III A) of the *Ansatz* circuit.

APPENDIX E: DETAILS OF NUMERICAL SIMULATIONS

We perform all state vector simulations using the open-source tools QuEST [90] and its high-level *Mathematica*-based interface QuESTlink [85]. Shot noise is simulated by adding Gaussian noise of standard deviation $1/\sqrt{N_s}$ to computed matrix and vector elements. Numerics for recompilation and the spin chain are both performed using a hardware-efficient *Ansatz* of the form in Fig. 11.

1. Effect of constraint number on performance

Data for Figs. 3 and 6 are obtained by simulating a 14-qubit parameter rediscovery problem using two layers of the *Ansatz* in Fig. 11. The initial states are initialized close to the solution by randomly perturbing the solution parameters resulting in an initial average fidelity of $\mathcal{F} = 46 \pm 7\%$. Fits in these figures are of the form $1 - \mathcal{F}_{\text{min}} = a(N_c/\nu)^{-b} + c$, and we report fitted parameters in Table I. The orange line in Fig. 10 is identical to the orange line in Fig. 6.

The noisy simulations are performed assuming the following simplified noise model. Recall that global depolarizing noise is a relatively good approximation in complex quantum circuits and becomes near exact for random circuits; refer for rigorous bounds to Ref. [91]. This error channel acts on any density matrix via the Kraus map $D(\rho) = F\rho + (1-F)\rho_{\text{max}}$, where ρ_{max} is the maximally mixed state, i.e., white noise, and F is the fidelity. The expected value of any traceless Hermitian operator O , such as Pauli strings as relevant in the present work, merely gets

attenuated as $\langle O \rangle = \text{Tr}[O\rho] = F\langle O \rangle_{\text{id}}$ where $\langle O \rangle_{\text{id}}$ is the ideal, noiseless expected value.

In practice, this error model does not capture more subtle physical processes that corrupt the expected value measurement. Nevertheless, it is shown in Ref. [48] that nearly all typical error models used in practice admit the decomposition $F\rho + (1-F)\rho_{\text{err}}$, where $\rho_{\text{err}} \approx \rho_{\text{max}}$ is an error density matrix that we do not expect to be exactly the maximally mixed state, albeit in practice it is relatively close to the maximally mixed state via its vanishing commutator norm from Ref. [48]. In order to go beyond global depolarization, but without resorting to computationally infeasible explicit noise simulations, we compute

TABLE I. In our numerical simulations, we investigate the effect of increasing the number N_c of covariances. We fit a function to the fidelity achieved by CoVaR of the form $1 - \mathcal{F} = a(N_c/\nu)^{-b} + c$. Fits to the worst (Max) and best (Min) of three runs of CoVaR are reported.

Fitted data	Type	a	b	c
Fig. 3 noise-free	Max	15.6	3.00	7×10^{-4}
	Min	5.62	3.23	0.0
Fig. 6 shot noise	Max	11.3	3.76	3×10^{-4}
	Min	0.0259	1.68	1×10^{-4}
Fig. 6 circuit noise	Max	7.80	3.09	2×10^{-4}
	Min	0.35	2.24	0.0
Fig. 10 orth. pool	Max	15.6	2.99	7×10^{-4}
	Min	0.15	2.51	0.0

the noisy expected value as $\langle O_k \rangle = \text{Tr}[O_k \rho] = F \langle O_k \rangle_{\text{id}} + (1 - F) \text{Tr}[O_k \rho_{\text{err}}]$ by approximating the term $\text{Tr}[O_k \rho_{\text{err}}] \sim \mathcal{N}(0, \sigma^2)$ using random Gaussian numbers. Here, we set $\sigma = 0.01$, which is determined by the distance of ρ_{err} from the maximally mixed state which we simulate with random Gaussian numbers that are unique to each observable indexed by k . Furthermore, we choose the fidelity $F = 0.9 \approx (1 - \epsilon_1)^{\nu_1} (1 - \epsilon_2)^{\nu_2}$, such that it approximates the performance of a typical, state-of-the-art experimental device with two-qubit error rates $\epsilon_2 = 0.001$ and single-qubit error rates 4 times smaller $\epsilon_1 = 0.25\epsilon_2$, given in our circuit we have $\nu_1 = 196$ and $\nu_2 = 52$ single- and two-qubit gates, respectively.

2. Recompilation problem in Fig. 7

The numerics for recompilation in Fig. 7(c) are done on a ten-qubit parameter rediscovery problem for two layers of hardware-efficient Ansätze ($\nu = 88$). Gradient descent is performed with a learning rate $\eta = 0.1$ for both VQE and V-VQE.

3. Spin-chain simulations

The Hamiltonian in Eq. (13) is used with parameters $J = 0.1$ and c_i chosen randomly between -1 and 1 . For the spin-chain simulations in Fig. 8, imaginary time evolution is used from a random initialisation until an energy of $E = -5.9$ is reached, with parameters θ_{imag} . These parameters are then disturbed by $|\Delta\theta_k| \leq 0.05$ to produce seven low-energy states. CoVaR is then run from these initial states for 40 iterations. Imaginary time evolution is also continued from θ_{imag} until convergence and reaches an energy difference to the ground state of $\Delta E = 0.012$ compared to the 4×10^{-4} of CoVaR.

[1] F. Arute *et al.*, *Quantum Supremacy Using a Programmable Superconducting Processor*, *Nature (London)* **574**, 505 (2019).
 [2] H.-S. Zhong *et al.*, *Phase-Programmable Gaussian Boson Sampling Using Stimulated Squeezed Light*, *Phys. Rev. Lett.* **127**, 180502 (2021).
 [3] Y. Wu *et al.*, *Strong Quantum Computational Advantage Using a Superconducting Quantum Processor*, *Phys. Rev. Lett.* **127**, 180501 (2021).
 [4] S. Ebadi, T. T. Wang, H. Levine, A. Keesling, G. Semeghini, A. Omran, D. Bluvstein, R. Samajdar, H. Pichler, W. W. Ho, S. Choi, S. Sachdev, M. Greiner, V. Vuletić, and M. D. Lukin, *Quantum Phases of Matter on a 256-Atom Programmable Quantum Simulator*, *Nature (London)* **595**, 227 (2021).
 [5] M. Gong *et al.*, *Quantum Walks on a Programmable Two-Dimensional 62-Qubit Superconducting Processor*, *Science* **372**, 948 (2021).
 [6] J. Preskill, *Quantum Computing in the NISQ Era and beyond*, *Quantum* **2**, 79 (2018).

[7] E. Farhi, J. Goldstone, and S. Gutmann, *A Quantum Approximate Optimization Algorithm*, [arXiv:1411.4028](https://arxiv.org/abs/1411.4028).
 [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] S. Endo, Z. Cai, S. C. Benjamin, and X. Yuan, *Hybrid Quantum-Classical Algorithms and Quantum Error Mitigation*, *J. Phys. Soc. Jpn.* **90**, 032001 (2021).
 [10] 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).
 [11] 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 Algorithms*, *Rev. Mod. Phys.* **94**, 015004 (2022).
 [12] L. Bittel and M. Kliesch, *Training Variational Quantum Algorithms Is NP-Hard*, *Phys. Rev. Lett.* **127**, 120502 (2021).
 [13] 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).
 [14] M. Larocca, N. Ju, D. García-Martín, P. J. Coles, and M. Cerezo, *Theory of Overparametrization in Quantum Neural Networks*, [arXiv:2109.11676](https://arxiv.org/abs/2109.11676).
 [15] B. van Straaten and B. Koczor, *Measurement Cost of Metric-Aware Variational Quantum Algorithms*, *PRX Quantum* **2**, 030324 (2021).
 [16] H.-Y. Huang, R. Kueng, and J. Preskill, *Predicting Many Properties of a Quantum System from Very Few Measurements*, *Nat. Phys.* **16**, 1050 (2020).
 [17] E. R. Anschuetz and B. T. Kiani, *Beyond Barren Plateaus: Quantum Variational Algorithms Are Swamped with Traps*, [arXiv:2205.05786](https://arxiv.org/abs/2205.05786).
 [18] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles, *Cost Function Dependent Barren Plateaus in Shallow Parametrized Quantum Circuits*, *Nat. Commun.* **12**, 1791 (2021).
 [19] A. Ferraro, S. Olivares, and M. G. Paris, *Gaussian States in Continuous Variable Quantum Information*, [arXiv:quant-ph/0503237](https://arxiv.org/abs/quant-ph/0503237) (2005).
 [20] A. Carmi and E. Cohen, *On the Significance of the Quantum Mechanical Covariance Matrix*, *Entropy* **20**, 500 (2018).
 [21] V. Tripathi, C. Radhakrishnan, and T. Byrnes, *Covariance Matrix Entanglement Criterion for an Arbitrary Set of Operators*, *New J. Phys.* **22**, 073055 (2020).
 [22] M. Hagan and M. Menhaj, *Training Feedforward Networks with the Marquardt Algorithm*, *IEEE Trans. Neural Networks* **5**, 989 (1994).
 [23] H. B. Demuth, M. H. Beale, O. De Jess, and M. T. Hagan, *Neural Network Design* (Oklahoma State University, Stillwater, 2014).
 [24] M. H. Beale, M. T. Hagan, and H. B. Demuth, *Neural Network Toolbox, Users Guide*, *MathWorks* **2**, 77 (2010).
 [25] H. Yu and B. M. Wilamowski, *Levenberg-Marquardt Training*, in *Intelligent Systems* (CRC Press, Boca Raton, FL, 2018).
 [26] 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).
- [27] A. G. Taube and R. J. Bartlett, *New Perspectives on Unitary Coupled-Cluster Theory*, *Int. J. Quantum Chem.* **106**, 3393 (2006).
- [28] J. Tilly, H. Chen, S. Cao, D. Picozzi, K. Setia, Y. Li, E. Grant, L. Wossnig, I. Rungger, G. H. Booth *et al.*, *The Variational Quantum Eigensolver: A Review of Methods and Best Practices*, *Phys. Rep.* **986**, 1 (2022).
- [29] A. Cuzzocrea, A. Scemama, W. J. Briels, S. Moroni, and C. Filippi, *Variational Principles in Quantum Monte Carlo: The Troubled Story of Variance Minimization*, *J. Chem. Theory Comput.* **16**, 4203 (2020), pMID: 32419451.
- [30] B. Koczor and S. C. Benjamin, *Quantum Natural Gradient Generalized to Nonunitary Circuits*, [arXiv:1912.08660](https://arxiv.org/abs/1912.08660).
- [31] B. Koczor and S. C. Benjamin, *Quantum Analytic Descent*, *Phys. Rev. Res.* **4**, 023017 (2022).
- [32] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, *Elucidating Reaction Mechanisms on Quantum Computers*, *Proc. Natl. Acad. Sci. U.S.A.* **114**, 7555 (2017).
- [33] S. Khatri, R. LaRose, A. Poremba, L. Cincio, A. T. Sornborger, and P. J. Coles, *Quantum-Assisted Quantum Compiling*, *Quantum* **3**, 140 (2019).
- [34] P. D. Johnson, J. Romero, J. Olson, Y. Cao, and A. Aspuru-Guzik, *QVECTOR: An Algorithm for Device-Tailored Quantum Error Correction*, [arXiv:1711.02249](https://arxiv.org/abs/1711.02249).
- [35] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th ed. (Cambridge University Press, Cambridge, England, 2011).
- [36] T. Jones and S. C. Benjamin, *Robust Quantum Compilation and Circuit Optimisation via Energy Minimisation*, *Quantum* **6**, 628 (2022).
- [37] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations* (Society for Industrial and Applied Mathematics, Philadelphia, 1996).
- [38] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *The Art of Scientific Computing*, Numerical Recipes 3rd ed. (Cambridge University Press, Cambridge, England, 2007).
- [39] H. Okawa, K. Fujisawa, Y. Yamamoto, R. Hirai, N. Yasutake, H. Nagakura, and S. Yamada, *The W4 Method: A New Multi-dimensional Root-Finding Scheme for Non-linear Systems of Equations*, [arXiv:1809.04495](https://arxiv.org/abs/1809.04495).
- [40] L. Pasquini and D. Trigianta, *A Globally Convergent Method for Simultaneously Finding Polynomial Roots*, *Math. Comput.* **44**, 135 (1985).
- [41] E. H. Bergou, Y. Diouane, V. Kungurtsev, and C. W. Royer, *A Stochastic Levenberg-Marquardt Method Using Random Models with Complexity Results*, *SIAM/ASA J. Uncertainty Quantif.* **10**, 507 (2022).
- [42] S. S. Liew, M. Khalil-Hani, and R. Bakhteri, *An Optimized Second Order Stochastic Learning Algorithm for Neural Network Training*, *Neurocomputing* **186**, 74 (2016).
- [43] S. Ruder, *An Overview of Gradient Descent Optimization Algorithms*, [arXiv:1609.04747](https://arxiv.org/abs/1609.04747).
- [44] R. Sweke, F. Wilde, J. Meyer, M. Schuld, P. K. Fahrmann, B. Meynard-Piganeau, and J. Eisert, *Stochastic Gradient Descent for Hybrid Quantum-Classical Optimization*, *Quantum* **4**, 314 (2020).
- [45] Stochastic gradient descent for VQE has been termed for instances when shot noise on estimated gradients is significant [44]. In contrast, the present approach is stochastic due to the random selection of constraints.
- [46] Z. Cai, R. Babbush, S. C. Benjamin, S. Endo, W. J. Huggins, Y. Li, J. R. McClean, and T. E. O'Brien, *Quantum Error Mitigation*, [arXiv:2210.00921](https://arxiv.org/abs/2210.00921).
- [47] B. Koczor, *Exponential Error Suppression for Near-Term Quantum Devices*, *Phys. Rev. X* **11**, 031057 (2021).
- [48] B. Koczor, *The Dominant Eigenvector of a Noisy Quantum State*, *New J. Phys.* **23**, 123047 (2021).
- [49] W. J. Huggins, S. McArdle, T. E. O'Brien, J. Lee, N. C. Rubin, S. Boixo, K. B. Whaley, R. Babbush, and J. R. McClean, *Virtual Distillation for Quantum Error Mitigation*, *Phys. Rev. X* **11**, 041036 (2021).
- [50] K. Sharma, S. Khatri, M. Cerezo, and P. J. Coles, *Noise Resilience of Variational Quantum Compiling*, *New J. Phys.* **22**, 043006 (2020).
- [51] C. Derby, J. Klassen, J. Bausch, and T. Cubitt, *Compact Fermion to Qubit Mappings*, *Phys. Rev. B* **104**, 035118 (2021).
- [52] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio *et al.*, *Variational Quantum Algorithms*, *Nat. Rev. Phys.* **3**, 625 (2021).
- [53] G. Pagano, A. Bapat, P. Becker, K. S. Collins, A. De, P. W. Hess, H. B. Kaplan, A. Kyriianidis, W. L. Tan, C. Baldwin *et al.*, *Quantum Approximate Optimization of the Long-Range Ising Model with a Trapped-Ion Quantum Simulator*, *Proc. Natl. Acad. Sci. U.S.A.* **117**, 25396 (2020).
- [54] M. P. Harrigan, K. J. Sung, M. Neeley, K. J. Satzinger, F. Arute, K. Arya, J. Atalaya, J. C. Bardin, R. Barends, S. Boixo *et al.*, *Quantum Approximate Optimization of Non-planar Graph Problems on a Planar Superconducting Processor*, *Nat. Phys.* **17**, 332 (2021).
- [55] M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, and N. Killoran, *Evaluating Analytic Gradients on Quantum Hardware*, *Phys. Rev. A* **99**, 032331 (2019).
- [56] H.-Y. Huang, R. Kueng, and J. Preskill, *Efficient Estimation of Pauli Observables by Derandomization*, *Phys. Rev. Lett.* **127**, 030503 (2021).
- [57] R. E. Moore, R. B. Kearfott, and M. J. Cloud, *Introduction to Interval Analysis* (Society for Industrial and Applied Mathematics, Philadelphia, 2009).
- [58] N. Schuch and J. Siewert, *Programmable Networks for Quantum Algorithms*, *Phys. Rev. Lett.* **91**, 027902 (2003).
- [59] L. Moro, M. G. A. Paris, M. Restelli, and E. Prati, *Quantum Compiling by Deep Reinforcement Learning*, *Commun. Phys.* **4**, 1 (2021).
- [60] A. Lucas, *Ising Formulations of Many NP Problems*, *Front. Phys.* **2** (2014).
- [61] C. Kokail, C. Maier, R. van Bijnen, T. Brydges, M. K. Joshi, P. Jurcevic, C. A. Muschik, P. Silvi, R. Blatt, C. F. Roos, and P. Zoller, *Self-Verifying Variational Quantum Simulation of Lattice Models*, *Nature (London)* **569**, 355 (2019).
- [62] D. J. Luitz, N. Laflorencie, and F. Alet, *Many-Body Localization Edge in the Random-Field Heisenberg Chain*, *Phys. Rev. B* **91**, 081103(R) (2015).

- [63] A. M. Childs, D. Maslov, Y. Nam, N. J. Ross, and Y. Su, *Toward the First Quantum Simulation with Quantum Speedup*, *Proc. Natl. Acad. Sci. U.S.A.* **115**, 9456 (2018).
- [64] R. Nandkishore and D. A. Huse, *Many-Body Localization and Thermalization in Quantum Statistical Mechanics*, *Annu. Rev. Condens. Matter Phys.* **6**, 15 (2015).
- [65] S. McArdle, T. Jones, S. Endo, Y. Li, S. C. Benjamin, and X. Yuan, *Variational Ansatz-Based Quantum Simulation of Imaginary Time Evolution*, *npj Quantum Inf.* **5**, 75 (2019).
- [66] J. Stokes, J. Izaac, N. Killoran, and G. Carleo, Quantum Natural Gradient, *Quantum* **4**, 269 (2020).
- [67] N. Yamamoto, *On the Natural Gradient for Variational Quantum Eigensolver*, [arXiv:1909.05074](https://arxiv.org/abs/1909.05074).
- [68] O. Higgott, D. Wang, and S. Brierley, *Variational Quantum Computation of Excited States*, *Quantum* **3**, 156 (2019).
- [69] L. Bittel and M. Kliesch, *Training Variational Quantum Algorithms Is NP-Hard*, *Phys. Rev. Lett.* **127**, 120502 (2021).
- [70] D.-B. Zhang, Z.-H. Yuan, and T. Yin, *Variational Quantum Eigensolvers by Variance Minimization*, [arXiv:2006.15781](https://arxiv.org/abs/2006.15781).
- [71] J. R. McClean, M. E. Kimchi-Schwartz, J. Carter, and W. A. de Jong, *Hybrid Quantum-Classical Hierarchy for Mitigation of Decoherence and Determination of Excited States*, *Phys. Rev. A* **95**, 042308 (2017).
- [72] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization* (Academic, London, 1981).
- [73] A. Mari, T. R. Bromley, and N. Killoran, *Estimating the Gradient and Higher-Order Derivatives on Quantum Hardware*, *Phys. Rev. A* **103**, 012405 (2021).
- [74] E. Chertkov and B. K. Clark, *Computational Inverse Method for Constructing Spaces of Quantum Models from Wave Functions*, *Phys. Rev. X* **8**, 031029 (2018).
- [75] X.-L. Qi and D. Ranard, *Determining a Local Hamiltonian from a Single Eigenstate*, *Quantum* **3**, 159 (2019).
- [76] A. D. Bookatz, *QMA-Complete Problems*, *Quantum Inf. Comput.* **14**, 361 (2014).
- [77] A. A. Akhtar, H.-Y. Hu, and Y.-Z. You, *Scalable and Flexible Classical Shadow Tomography with Tensor Networks*, [arXiv:2209.02093](https://arxiv.org/abs/2209.02093).
- [78] C. Bertoni, J. Haferkamp, M. Hinsche, M. Ioannou, J. Eisert, and H. Pashayan, *Shallow Shadows: Expectation Estimation Using Low-Depth Random Clifford Circuits*, [arXiv:2209.12924](https://arxiv.org/abs/2209.12924).
- [79] O. Crawford, B. van Straaten, D. Wang, T. Parks, E. Campbell, and S. Brierley, *Efficient Quantum Measurement of Pauli Operators in the Presence of Finite Sampling Error*, *Quantum* **5**, 385 (2021).
- [80] T.-C. Yen, V. Verteletskyi, and A. F. Izmaylov, *Measuring All Compatible Operators in One Series of Single-Qubit Measurements Using Unitary Transformations*, *J. Chem. Theory Comput.* **16**, 2400 (2020).
- [81] A. Jena, S. Genin, and M. Mosca, *Pauli Partitioning with Respect to Gate Sets*, [arXiv:1907.07859](https://arxiv.org/abs/1907.07859).
- [82] P. Gokhale, O. Angiuli, Y. Ding, K. Gui, T. Tomesh, M. Suchara, M. Martonosi, and F. T. Chong, $O(N^3)$ Measurement Cost for Variational Quantum Eigensolver on Molecular Hamiltonians, *IEEE Trans. Quantum Eng.* **1**, 1 (2020).
- [83] K. Wan, W. J. Huggins, J. Lee, and R. Babbush, *Matchgate Shadows for Fermionic Quantum Simulation*, [arXiv:2207.13723](https://arxiv.org/abs/2207.13723).
- [84] A. Zhao, N. C. Rubin, and A. Miyake, *Fermionic Partial Tomography via Classical Shadows*, *Phys. Rev. Lett.* **127**, 110504 (2021).
- [85] T. Jones and S. Benjamin, *Questlinkmathematica Embiggened by a Hardware-Optimised Quantum Emulator*, *Quantum Sci. Technol.* **5**, 034012 (2020).
- [86] Y. Li and S. C. Benjamin, *Efficient Variational Quantum Simulator Incorporating Active Error Minimization*, *Phys. Rev. X* **7**, 021050 (2017).
- [87] Here, the product of sets produces a set that contains all possible products of the elements.
- [88] O. Kyriienko and V. E. Elfving, *Generalized Quantum Circuit Differentiation Rules*, *Phys. Rev. A* **104**, 052417 (2021).
- [89] D. Wierichs, J. Izaac, C. Wang, and C. Y.-Y. Lin, *General Parameter-Shift Rules for Quantum Gradients*, *Quantum* **6**, 677 (2022).
- [90] T. Jones, A. Brown, I. Bush, and S. C. Benjamin, *QuEST and High Performance Simulation of Quantum Computers*, *Sci. Rep.* **9**, 10736 (2019).
- [91] A. M. Dalzell, N. Hunter-Jones, and F. G. Brandão, *Random Quantum Circuits Transform Local Noise into Global White Noise*, [arXiv:2111.14907](https://arxiv.org/abs/2111.14907).