Hamiltonian-based data loading with shallow quantum circuits

Bojia Duan [*]
School of Computer and Electronic Information/School of Artificial Intelligence, Nanjing Normal University, Nanjing 210023, China

Chang-Yu Hsieh [†]
Tencent Quantum Laboratory, Shenzhen 518000, China

(Received 10 July 2022; accepted 3 November 2022; published 18 November 2022)

Data loading with shallow quantum circuits is a highly desirable ingredient for efficient execution of many quantum algorithms before large-scale quantum error corrections with full fault tolerance become readily available. The need for efficient data loading is especially urgent for the study of quantum machine learning. In this work, we propose a protocol that only uses a parameterized shallow quantum circuit without ancilla qubits for loading data into the amplitude of a quantum state with high fidelity. We term this data-loading method Hamiltonian-based data loading (HDL), which comprises two stages. First, the HDL algorithm identifies a Hamiltonian $\hat{H}$ whose unique ground state $|\psi\rangle$ represents the normalized data $\vec{x}$ in the form of amplitude encoding. Next, the target state is reconstructed with a parameterized quantum circuit by utilizing methods like variational quantum eigensolver to minimize energy. In this work, we provide three convincing examples to demonstrate the effectiveness of HDL for loading an N-dimensional classical data with minimal quantum resources ($O[\text{poly}(\log_2 N)]$-depth quantum circuit without ancilla qubits). Our approach is particularly useful for quantum hardware without large-scale error corrections, and shall benefit the development of quantum machine learning algorithms.

DOI: 10.1103/PhysRevA.106.052422

I. INTRODUCTION

Loading classical data into the amplitudes of a quantum state with high efficiency and high fidelity is a prerequisite for effective execution of many quantum machine learning algorithms—for instance, quantum classifier [1,2], quantum dimensionality reduction [3–5], quantum clustering [6], and quantum recommendation systems [7]. A well-known data-loading method is to use quantum random access memory (QRAM), which facilitates many essential quantum algorithms via accessing classical data in superposition [8–15], but it is extremely challenging to build scalable QRAM hardware. Hence, researchers have been investigating alternative technology to accomplish this task, such as a purely circuit-based protocols for data loading [16–20].

To implement nontrivial quantum algorithms in the near term, it is crucial to optimize the usage of quantum resources, such as minimizing the circuit depth of quantum algorithms and protocols. One of the most pressing tasks is to reduce the quantum resources for the aforementioned data loading or, equivalently, quantum state preparation (assuming we have properly formulated the data to be loaded as a properly normalized target state $|\psi_T\rangle$). The goal is to design a low-depth quantum circuit for loading an N-dimensional state vector into an n-qubit poly(n)-depth circuit, where $n = O(\lceil \log_2 N \rceil)$ is highly desirable. Quantum state preparation is an actively researched topic with many exemplary proposals attempting to solve this formidable problem [16–27]. Recently, Sun *et al.* showed that with m ancillary qubits, a quantum state can be prepared in circuit depth $\tilde{O}([2^n/(m+n)+n]$ [16]. And Iordanis *et al.* proposed a forge data-loading method by using a two-qubit parameterized gate termed Reconfigurable BeamSplitter (RBS), with a circuit depth $O(n)$ and N qubits [17]. That means the total number of quantum gates cannot break through the linear complexity of $O(N)$, which is exponential in n. In order to simultaneously confine the circuit complexity to $O(n)$ qubits and $O[\text{poly}(n)]$ circuit depth in the noisy intermediate-scale quantum (NISQ) era [28] and early fault-tolerant period, we propose an approximate data-loading method which resorts to the framework of the variational quantum algorithm (VQA) [29] for exploiting the power of a noisy and shallow quantum circuit for applications. Although VQA requires classical training loops to optimize parameterized quantum circuits and may only achieve approximate state preparation, it does make data loading with shallow circuits possible, as convincingly proved in Ref. [20]. Our work is similarly motivated but follows a completely different technical path, as further explained later.

We first review the state-of-the-art methods (both exact and approximate) on quantum state preparations. An exact quantum state preparation method typically represents the state as a binary tree data structure, with the tree nodes storing rotational angles that could be used to reconstruct the state exactly in a quantum circuit. Some examples are summarized here. In Ref. [21], Möttönen *et al.* introduce a method to efficiently implement an amplitude loading of data using uniformly controlled rotations, which costs $2^{n+2} - 4n - 4$ controlled NOT (CNOT) gates and $2^{n+2} - 5$ one-qubit elementary rotations to reconstruct an n-qubit quantum state.

[*]duanbojia@njnu.edu.cn
[†]Corresponding author: kimhsieh2@gmail.com

Inspired by Ref. [21], Araujo *et al.* introduced a divide-and-conquer algorithm for loading classical data [22]. By exchanging space for time, this algorithm may load an 2^n-dimensional vector with an $O_q(n^2)$-depth quantum circuit but using $O(2^n)$ ancillary qubits to facilitate the state preparation. Later on, Zhang *et al.* improved the method in Ref. [22] by increasing the ancilla qubits to $O(4^n)$ and reducing entanglement to at most $O(n)$ qubits.

On the other hand, approximate quantum state preparation methods simply use a shallow parameterized quantum circuit to encode classical data by finding gate parameters that minimizes a loss function [20,23–27].

References [25,26] use quantum generative adversarial networks (qGAN) to encode nonnegative classical data in terms of a fixed prior distribution, and Ref. [20] further expands the application in real-valued data by adding an extra measurement in the Hadamard-transformed basis to properly introduce the sign, which provides a head start for solving data-loading problems with quantum variational algorithms.

Different from all the methods mentioned above, we propose an approach named Hamiltonian-based data loading (HDL) protocol for quantum state preparation. Inspired by the idea that, for many target quantum states, one can efficiently reconstruct a local Hamiltonian (containing $O[\text{poly}(n)]$ Pauli strings) that has the target state as one of its eigenstates [30] on a classical computer, then one can resort to the variational quantum eigensolver (VQE) [31] to train a chosen parameterized quantum circuit ansatz by minimizing the energy-dependent loss function until the target state is recovered by the circuit. In this case, we trade off the scarce quantum resources with relatively affordable classical computations. Further comparisons and discussions on our method and other variational methods are deferred to Sec. IV.

To illustrate the practicality of our method, we perform and report numerical simulations involving random probability distributions and realistic data such as the compressed Modified National Institute of Standards and Technology (MNIST) handwritten data to validate our claims; a low-depth parameterized quantum circuit can faithfully represent a variety of useful data in the context of data loading for quantum algorithms, especially machine learning applications. In particular, we show how to build a circuit-based QRAM with the HDL method.

The remainder of the paper is organized as follows: Sec. II proposes our algorithm for loading classical data. Section III reports the numerical results and corresponding remarks. We compare our method against alternative approaches for data loading with quantum circuits and give a short conclusion in Secs. IV and V, respectively. In Appendix A, we provide a brief summary of HDL protocol. In Appendix B, we provide additional justification on why the HDL protocol could be highly relevant for the development of quantum machine learning.

II. METHOD

In this section, we present a variational protocol for quantum state preparation with a low-depth parameterized quantum circuit. We term this method HDL. Before we introduce this method in detail, we first define the problem and provide a rationale that motivates this work.

For a given normalized vector of classical data $\vec{x} = (x_1, x_2, \ldots, x_N) \in \mathbb{C}^N$, our goal is to construct a low-depth circuit which outputs the quantum state $|\psi_\theta\rangle$ that approximates $|\psi_T\rangle = \sum_{i=1}^N x_i|i\rangle$. A straightforward idea is to use the absolute fidelity $|\mathcal{F}_\theta| = |\langle\psi_\theta|\psi_T\rangle|$ as the loss function for training the parameters θ with a classical computer.

While this approach sounds reasonable, when the number of qubits n is sufficiently large it still becomes undesirable to classically evaluate $|\mathcal{F}_\theta|$, which naively scales as $O(2^n)$, at every training step. The fidelity $\mathcal{F}_\theta$ cannot be efficiently evaluated on a quantum computer when one does not know how to prepare $|\psi_T\rangle$ in a quantum circuit. However, if the loss function is the expected value of a Hermitian observable $\hat{H}$, then one can utilize a quantum computer to efficiently estimate $E_\theta = \langle\psi_\theta|\hat{H}|\psi_\theta\rangle$. Ideally, $\hat{H}$ should be a linear combination of at most $\text{poly}(n)$ number of k-local Pauli strings.

Hence, the first part of our proposed protocol is to adopt the correlation-matrix method to derive a k-local Hamiltonian [with $\text{poly}(n)$ number of Pauli strings] whose ground state is sufficiently close to the target $|\psi_T\rangle$. Once an appropriate Hamiltonian is chosen, the task is then reduced to the standard variational quantum simulation, which one can solve with methods like VQE [31] or the variational implementation of the imaginary evolution [32]. The whole protocol is succinctly summarized as a schematic in Fig. 1. The rest of this section discusses how to construct the appropriate $\hat{H}$ and the complexity of the HDL protocol.

A. The correlation-matrix method

In order to construct a Hamiltonian $\hat{H}$ whose unique ground state is sufficiently close to $|\psi_T\rangle$, we first generate a k-local Hamiltonian $\hat{H}_0$, comprising a $\text{poly}(n)$ number of Pauli strings, by following the basic idea outlined in Ref. [30]. Once a $\hat{H}_0$ is found, then we may define $\hat{H} = (\hat{H}_0 - e\hat{I})^2$, with $e = \langle\psi_T|\hat{H}_0|\psi_T\rangle$, to ensure that a standard VQE and similar algorithms can recover $|\psi_T\rangle$ with a parameterized quantum circuit.

The correlation-matrix method begins with selecting a set of mutually independent Hermitian operators $\{L_i\}$ acting in the n-qubit Hilbert space. The maximum set of $\{L_i\}$ comprises all 4^n Pauli strings. For our purposes, it is desirable to control the cardinality of this set. To this end, we impose a circular topology (i.e., linear chain with closed boundary) on the qubits, and only consider k-local Pauli strings that are made up of k qubits chosen from the circle with a fixed interval s apart,

$$P_i^{\alpha_i} P_{i+1+s}^{\alpha_{i+1+s}} \cdots P_{i+k-2+s}^{\alpha_{i+k-2+s}} P_{i+k-1+s}^{\alpha_{i+k-1+s}}, \tag{1}$$

that satisfy this topology. By only considering up to all possible k-local Paulis defined above, we then calculate correlation matrix elements M_{ij} with respect to $|\psi_T\rangle$,

$$M_{ij} = \tfrac{1}{2}\langle\psi_T|\{L_i, L_j\}|\psi_T\rangle - \langle\psi_T|L_i|\psi_T\rangle\langle\psi_T|L_j|\psi_T\rangle, \tag{2}$$

where $\{\cdot,\cdot\}$ denotes the anticommutator. One then diagonalizes the correlation matrix M, which is positive semidefinite, and inspects its lowest eigenvalue. If it is greater than zero, then we have to repeat the process with an enlarged basis

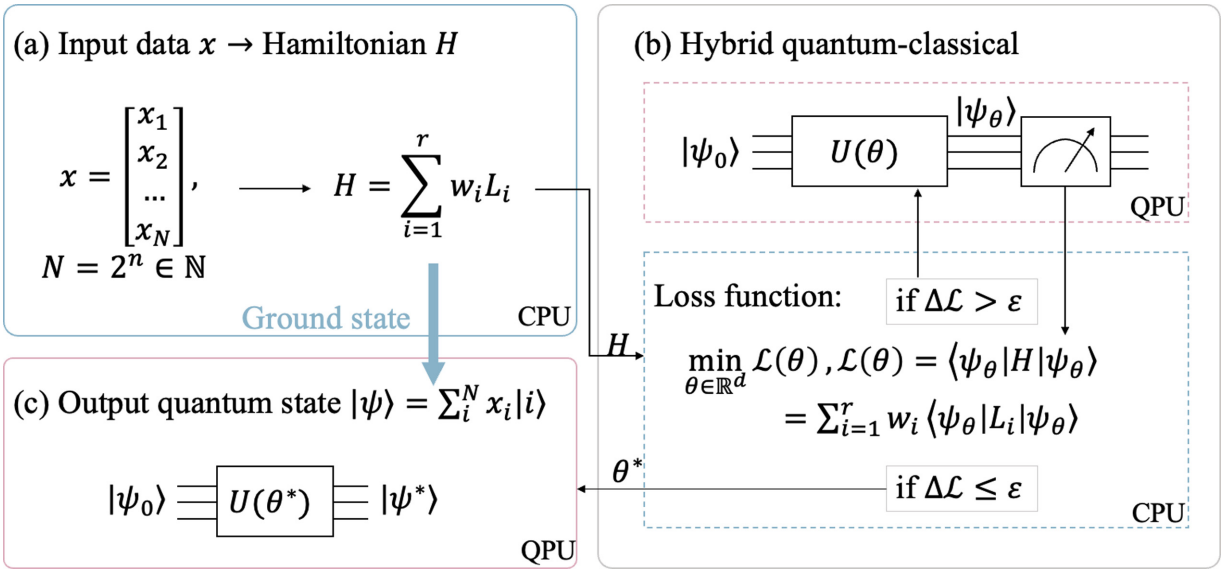

FIG. 1. The schematic of the HDL protocol. (a) At first, the input data x is provided to construct a Hamiltonian $H = \sum_i \omega_i L_i$ with a set of observables L_i and parameters ω_i by using the correlation-matrix method illustrated in Sec. II A. (b) Then the quantum processing unit (QPU) will be invoked to estimate the energy of H. With these estimations, a classical central processing unit (CPU) will evaluate the loss function and provide updates to the gate parameters. These hybrid quantum-classical loops will continue until the loss function gets below the tolerance ε. (c) Once the optimization stops, ideally, the learned parameterized quantum circuit $U(\theta^*)$ will output a state that closely resembles the desired quantum state $|\psi\rangle$ with high fidelity.

$\{L_i\}$ with a larger k to extend the locality or increase the fixed interval parameter s defined in Eq. (1). If the lowest eigenvalue is zero then we get the Hamiltonian $\hat{H}_0 = \sum_i w_i L_i$ with $\vec{w}$ the lowest-energy eigenvector of M satisfying $M\vec{w} = 0$.

B. Computational complexity of the HDL protocol

The computational complexity of the HDL protocol is determined by two stages of computations: finding $\hat{H}$ with the correlation matrix method and finding $|\psi_T\rangle$ with VQE. The cost of the first stage as shown in Algorithm 1 (in Appendix A) mainly depends on the construction and diagonalization of the correlation matrix M. The complexity for getting $\hat{H}_0$ is $O(NL_M^2) + O(L_M^3)$. Computing each M_{ij} scales as $O(N)$ and there are L_M^2 matrix elements in M with the matrix dimension given by $L_M = O[\log_2(N)3^k]$ for our current setup of considering only linear circular topology for the local $\hat{H}_0$. Diagonalization of M generally costs $O(L_M^3)$.

For the second part of the HDL protocol, it is not easy to rigorously determine the computational complexity of the training part of VQE and similar hybrid quantum-classical algorithms. Mainly, we cannot confidently determine the number of iterative gradient descents to perform. Overall, the complexity is given by $O(N_t N_m)$, where N_t is the number of training steps to get a converged VQE simulation and N_m is the number of measurements on the Pauli strings making up $\hat{H}$. Currently, many proposals have been put forward to drastically reduce the total number of measurements to get an accurate estimation of Pauli strings, such as the classical shadow [33–35], machine learning methods [36–38], grouping maximum number of commutative Pauli strings (e.g., tensor product basis) [39–44], and entanglement-assisted measurements [45–47].

It is worth pointing out that the fact that we impose a low-depth quantum circuit with the depth scales at most as poly(n) without using any ancillary qubits already stands out as an advantage against more traditional methods if we succeed at reconstructing the target state in the NISQ era. Further details on comparisons between methods are given in Sec. IV.

III. NUMERICAL SIMULATIONS

In this section, we numerically demonstrate data loading with the HDL protocol using shallow quantum circuits and without any ancilla qubits. The VQE simulations are performed with PENNYLANE [48]. In the first stage, the Hamiltonian is obtained by the correlation matrix method. In the second stage, we consider the most common hardware efficient ansatz (HEA) [39] with L layers, and Fig. 2 gives the schematic of one such layer.

The unitary operator $U(\vec{\theta})$ appearing in the HEA in Fig. 2 is defined in Eq. (3), which is a general single-qubit rotation.

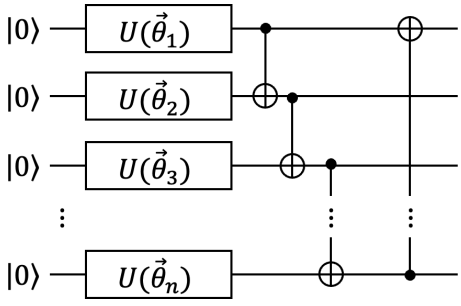

FIG. 2. The ansatz in the variational quantum circuit, where n represents the number of qubits.

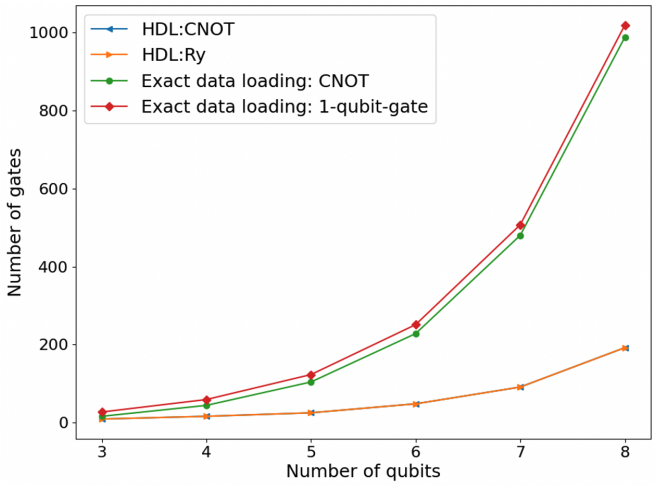

FIG. 3. Comparison of the number of quantum gates for data loading in the HDL and Ref. [21].

However, in this work, we only consider real-valued data, and we will consistently adopt $U(\vec{\theta}) = R_y(\theta_2) = e^{-i\theta_2\sigma_y/2}$ in order to minimize the number of unnecessary parameters. In other words, we always consider $\theta_3 = \theta_1 = 0$ in Eq. (3):

$$U(\vec{\theta}) = R_z(\theta_3)R_y(\theta_2)R_z(\theta_1) = \begin{pmatrix} e^{-i(\theta_1+\theta_3)/2}\cos(\theta_2/2) & -e^{i(\theta_1-\theta_3)/2}\sin(\theta_2/2) \\ e^{-i(\theta_1-\theta_3)/2}\sin(\theta_2/2) & e^{i(\theta_1+\theta_3)/2}\cos(\theta_2/2) \end{pmatrix}. \tag{3}$$

A. Real-valued random distribution

We begin with the investigation of loading randomly sampled real-valued data into the HEA circuit introduced earlier. More precisely, we consider from $N = 8$ up to $N = 256$ dimensional data vector (i.e., equivalent to $n = 3 \sim 8$ qubits). Each input data is uniformly sampled from standard normal distribution, and the entire data vector is properly normalized for the quantum state preparation. Loading random distributions is a common test for state-preparation protocols [25,26].

As clearly shown in Fig. 2 with the single-qubit gates restricted to R_y rotations, the number of CNOT gates and one-qubit rotational gates is equal to nL, where $n = \log_2 N$ is the number of qubits and L is the number of ansatz layers. To more fairly assess the feasibility, we consider 50 random sampled data vectors for each number of qubits n for data loading. We average the number of quantum gates required to successfully prepare the target quantum state (with fidelity greater than 99%) with the HEA using the HDL protocol. In each case, we try to minimize the number of layers in the n-qubit HEA to reach the targeted fidelity. Note that the fidelity obtained in the numerical study refers to the inner product between the input data and the output of the learned variational quantum circuit.

We compare the HDL with an exact data-loading quantum circuit introduced in Ref. [21]. Both the number of CNOT gates and one-qubit elementary rotations needed for the two state-preparation protocols are summarized in Fig. 3. For the exact data-loading protocol considered here, the number of gates is always the same for any n-qubit state unless one performs circuit optimization. The result clearly manifests that the number of quantum gates in the HDL only scales polynomially with the number of qubits; however, the exact method scales exponentially as given in Ref. [21]. The advantage of the variational method in the NISQ era is certainly very clear.

B. Structured data: MNIST handwritten digits

Now we consider loading structured data that is more relevant for practical tasks. To this end, we consider loading MNIST handwritten digits, which are commonly used for training and benchmarking machine learning models. For our illustrative investigations, we decide to trim the original 28×28 pixels of MNIST figures down to 16×16 pixels by simply removing peripheral pixels around the handwritten digits in each figure. After the compression, the handwritten data can be amplitude encoded with only $\log_2(16 \times 16) = 8$ qubits.

The highly structured pattern in the MNIST image files is manifested with the observation that a universal form of Hamiltonians could be used to encode all the MNIST image files we have tested in this study. In other words, we can use an identical set of $\{L_i\}$ to construct the M matrix for each MNIST image, and get $\hat{H} = \sum_i \omega_i L_i$. This is a nice property, as the determination of the most compact basis for spanning the M matrix may often take some trial and error.

Now, we look at how well the HDL performs in loading some realistic data into the shallow parameterized quantum circuits. Again, we still use the HEA for the VQE simulations. The results of ten different digits (one example per digit) are summarized in Fig. 4. We compare the original image and the approximated image, as outputted by the VQE method. Figure 4(c) presents output data given by an $L = 16$-layer HEA (composed of 128 CNOT gates and 128 R_y rotations) with a fidelity of at least 95%, and Fig. 4(d) presents output data given by an $L = 24$-layer ansatz (composed of 192 CNOT gates and 192 R_y rotations) with a fidelity of at least 99%. If these image files are meant for training machine learning models, it seems that the reconstructed figures in Fig. 4(c) are sufficient. On average, increasing the fidelity by 4% requires a nontrivial addition of eight layers for the given HEA circuit architecture with our predefined VQE simulation parameters for the MNIST data set. In principle, with increasing n (the number of qubits), one may want to add more ansatz layers to enhance the expressive power of the quantum circuit in order to achieve higher fidelity for the state preparation. However, in real hardware plagued by environmental noises, an optimal number for the ansatz layer may exist such that further increasing the circuit depth results in a dip of fidelity. A simple solution is certainly to start with a modest number of circuit layers and gradually increase the circuit depth to improve the fidelity for state preparation. Because we know the actual ground-state energy of the Hamiltonian and the target state, it is easy to diagnose whether further increasing the circuit depth helps us or not.

The trade-off between high fidelity and circuit depth is highly flexible in the variational framework; one can adeptly decide this fidelity-versus-circuit-depth depending on the

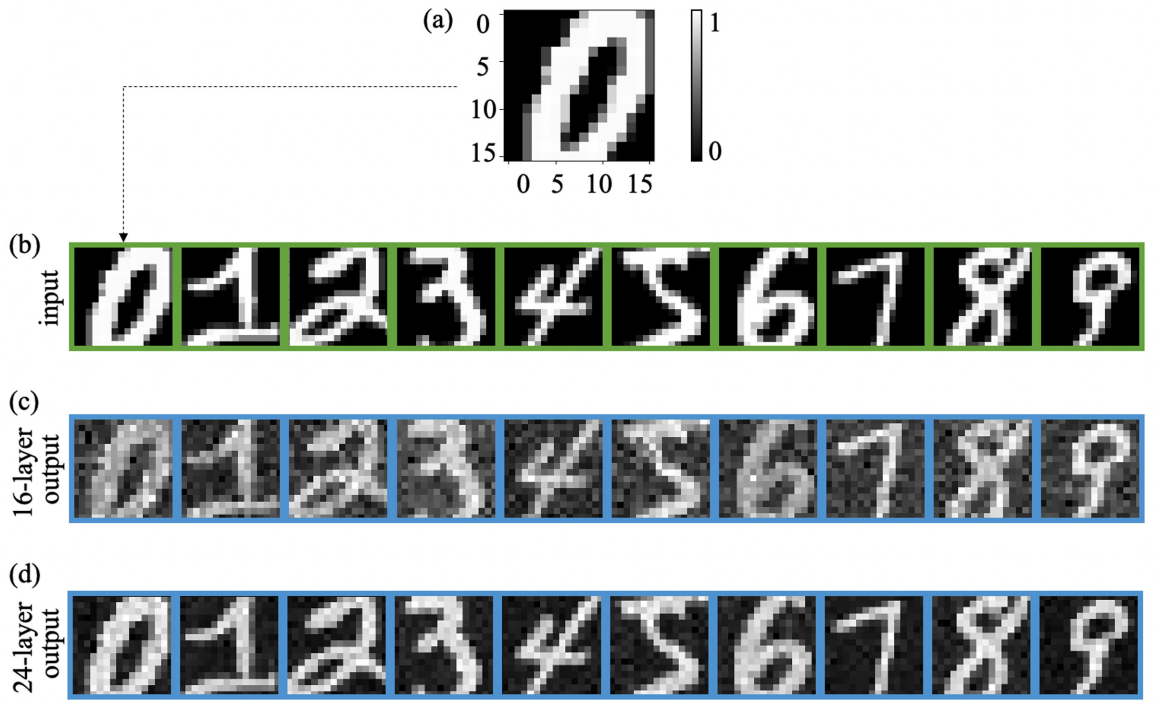

FIG. 4. Experimental results of quantum state representations of ten different grayscale handwritten digits. (a) An example of a grayscale picture of handwritten digits with 16×16 pixels. (b) The grayscale pictures represent ten different input handwritten digits. (c) The grayscale pictures represent the output with a 16-layer quantum circuit, with fidelity of each of at least 95%. (d) The grayscale pictures represent the output with a 24-layer quantum circuit, with fidelity of each of at least 99%.

quality of the quantum hardware and the quality of data fidelity for the specific task at hand.

C. Circuit-based QRAM: Controlled quantum state preparation

Our proposed HDL protocol can be easily generalized to load multiple data, each associated with a corresponding address index. This composite data structure implements a circuit-based QRAM [8] as defined in Ref. [49]. If we may afford the classical computations, the HDL does not train and load each data individually; instead, we can define this QRAM wave function $|\psi_{\text{qram}}\rangle$, construct the corresponding $\hat{H}$ in the extended Hilbert space of the address and data qubits, and execute parameterized circuit learning with VQE only once. Again, the target quantum state of the learned shallow parameterized quantum circuits is the superposition of all the input data (each with a unique address). Therefore, HDL can serve as a useful subroutine in data preprocessing for quantum machine learning tasks such as quantum classification and quantum clustering.

Now we illustrate this idea with a concrete illustration. As shown in Fig. 5, the four images with $4 \times (4 \times 8) = 128$ dimension will be loaded into a seven-qubit parameterized quantum circuit. The same HDL protocol is executed on a seven-qubit parameterized circuit to prepare the state $|\psi_{\text{qram}}\rangle = \sum_i |i\rangle|\psi_T^i\rangle$, where $|\psi_T^i\rangle$ corresponds to the ith data point. The top circuit in the blue square acts as the address qubits for data, and the lower part of the circuit in the orange square acts as each encoded data point (conditioned on the state of the address qubits). It is also straightforward to optimize the gate parameters in this case. We show the output data (fidelity 99.69%) by the $L = 11$-layer HEA circuit in the right part of Fig. 5. As the ansatz depth scales as $O[\text{poly}(n)]$ by design (where n is the total number of address and data qubits), the depth of our method is still log-polynomial with both the number and dimension of the data in this case.

IV. DISCUSSION

In this section, we briefly discuss why variational methods could be competitive against more traditional and exact data-loading methods. We then further compare the HDL against other variational methods to highlight the HDL's potential advantages. Further discussions about the relevance of small qubit systems to classical data loading are shared in Appendix B.

A. Variational data loading versus exact methods

Table I summarizes and compares the circuit depth and the number of qubits required by different state-preparation protocols. As shown, the HDL demands much fewer quantum resources than other exact data-loading methods [16,17,21]. At first thought, we could attribute the quantum-resource frugality of the variational methods to the exchange for heavy classical optimization loops to minimize the loss function, which is argued to be NP-hard [50]. However, we caution that all these exact loading methods also incur a steep computational complexity for classical processing, because these methods need to calculate an exponential number $[O(2^n)]$ of controlled-rotation angles [16,17,21]. Hence, these exact data-loading methods might not necessarily have an advantage

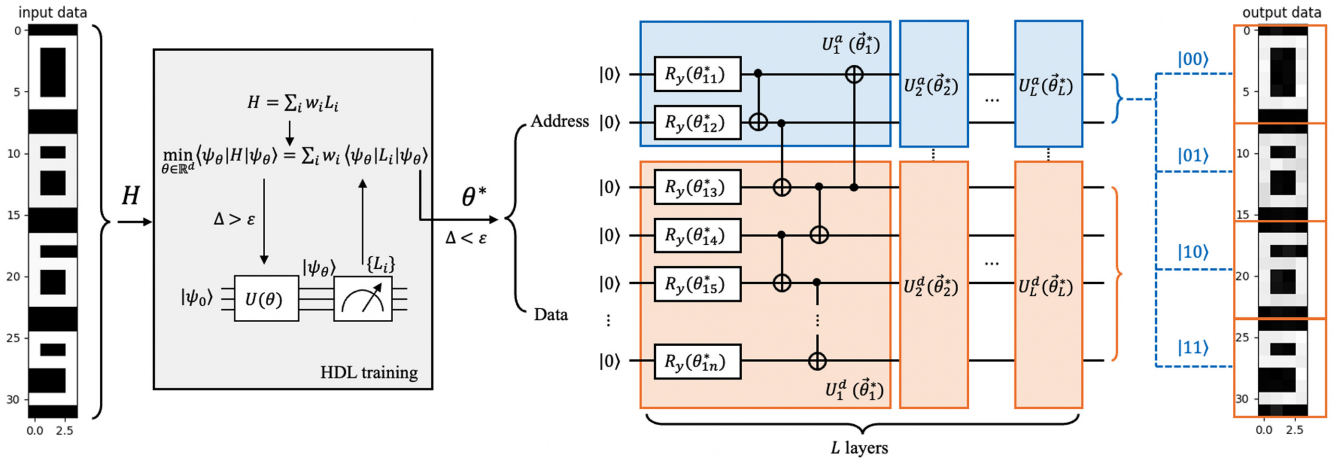

FIG. 5. The schematic of loading multiple data via the HDL protocol. Four images with 128 pixels will be loaded into a seven-qubit quantum circuit. Then the same HDL protocol is executed on a seven-qubit parameterized circuit to prepare the state $|\psi_{\text{qram}}\rangle = \sum_i |i\rangle|\psi_T^i\rangle$, where $|\psi_T^i\rangle$ corresponds to the ith data point. The qubits in the top circuit in the blue square act as the address qubits, and the lower part of the circuit in the orange square as each encoded data point (conditioned on the state of the address qubits).

over classical processing either (at least in the way they are currently formulated).

A clear disadvantage of the variational methods is the range of applicability, i.e., if certain data (or the corresponding quantum state) cannot be accurately approximated by a low-depth parameterized quantum circuit with any ansatz layout. This is a very interesting open question deserving further investigation. We currently do not have an answer, but we have three comments. Firstly, attempting to map the target quantum state to a compact and local Hamiltonian is one way to leverage physicists' insights with local Hamiltonians to assess the difficulty of reconstructing the state in a low-depth quantum circuit. Secondly, many practical data comes with some inherent structure that restrains the complexity, such as the MNIST images discussed in Sec. III. A $O[\text{poly}(n)]$-depth circuit indeed can approximate the state sufficiently accurately for the downstream task. Finally, we note that many advanced techniques have been proposed to perform variational calculations with adaptable ansatz-adjustment algorithms or quantum architecture search [51–54] to identify a highly customized ansatz layout for the problem at hand. These advanced methods should substantially mitigate the challenge of approximating quantum states with ansatz circuits. Therefore, we believe that the variational method for state preparation is a very competitive option in the NISQ era and the early fault-tolerant period (when the full fault tolerance is not sufficiently robust). Its adoption should greatly facilitate the development and testing of many quantum machine learning algorithms and beyond.

B. Comparing HDL and other variational state preparations

Next, we compare the HDL against other variational data-loading methods. Three major aspects distinguish the HDL from existing algorithms. Firstly, some previous variational data-loading methods, inspired by the deep-learning generative models to learn the probability distribution from training data, can only deal with positive input data, such as Refs. [25,26], by design. Reference [20] generalized the quantum generative model for handling any real-valued data. Different from their approaches, our method naturally supports loading fully complex-valued quantum states, the most general scenario.

Secondly, our HDL has an extra computational burden in comparison to other variational methods, and it is to construct and diagonalize the correlation matrix M. However, we argue there could be situations in which it is advantageous to do so. Once we manage to successfully derive an appropriate $\hat{H}$ for the HDL protocol, the VQE procedure requires only polynomial scaling of independent measurements on Pauli strings in $\hat{H} = \sum_i \omega_i L_i$ at every learning step. On the other hand, other methods that tend to use a loss function to evaluate the differences between probability distributions such as the maximum mean discrepancy (MMD) in Ref. [20]. In princi-

TABLE I. Comparison between the HDL (or variational methods in general) and exact data-loading methods in Refs. [16,17,21] to encode a $N = 2^n$ dimensional real positive vector into a quantum computer.

Methods	HDL and alike	Ref. [16]	Ref. [17]	Ref. [21]
Depth	$O[\text{poly}(n)]$	$\tilde{O}\left(\frac{2^n}{m+n}+n\right)$[a]	$O(n)$	$O(2^n)$
No. of gates	$O[\text{poly}(n)]$	$O(2^n)$	$O(2^n)$	$O(2^n)$
No. of qubits	n	$m+n$	2^n	n
Classical	Yes	Yes	Yes	Yes

[a] m represents the ancilla qubits needed in Ref. [16].

ple, this kind of loss function requires an exponential number of sampling as shown in Eq. (4),

$$L_{\mathrm{MMD}}(q_\theta, p) = \left| \sum_{j=0}^{N-1} q_\theta(j)\Phi(j) - \sum_{j=0}^{N-1} p(j)\Phi(j) \right|^2, \qquad (4)$$

where $q_\theta(j) = |\langle j|\psi_\theta\rangle|^2$, $p(j) = |\langle j|\psi_T\rangle|^2$, and $\Phi(i)$ is a function that maps j to a feature space. Since MMD has to be evaluated at every learning step, the efforts invested in deriving $\hat{H}$ prior to the classical optimization loop certainly gives HDL a huge computational advantages if the MMD has to be rigorously computed according to Eq. (4). In practice, though, one often notices that the MMD could be accurately estimated with a manageable number of projective measurement shots ($N_{\mathrm{shot}} \ll N$) and having an error scaling of $O(1/\sqrt{N_{\mathrm{shot}}})$. This sampling strategy completely hides the exponential scaling from the formal complexity arguments at each training step of these variational methods. However, we argue that this kind of trick can be easily adapted for our protocol. Whenever such an efficient sampling procedure for MMD allows the preparation of a highly accurate quantum state for the downstream tasks, one can argue that the same sampling technique can be used to estimate the matrix elements M_{ij}. Whenever the required N_{shot} becomes substantial, such as the underlying probability amplitudes for the quantum state are not concentrated enough, the HDL could be the better choice.

Thirdly, we discuss the last distinction separating the HDL from other variational methods. Variational ground-state preparation, popularized by the introduction of VQE [31], is definitely the most actively researched topic for the NISQ algorithmic developments. Once we have turned the state preparation problem to the ground-state preparation, we can adopt many advanced techniques that have already been proposed. For instance, in the case of ansatz selection, we can consider Hamiltonian variational ansatz [39,55,56], which has been shown to assuage the barren plateau in the loss-function landscape. One can also adopt Hamiltonian-generated imaginary-time evolution and the imaginary-time control method [57] to significantly accelerate the optimization for many cases. Whenever the Hamiltonian is sparse, Ref. [58] provides a decomposition of a general d-sparse Hamiltonian into just $O(d^2)$ Hermitian terms, which provides support when the constructed Hamiltonians are most efficiently described in terms of sparse matrices. To further restrict the circuit depth, methods to adaptively design a problem-specific quantum circuit architecture have also been investigated [59,60]. Moreover, many error-mitigation techniques have been developed [61–65] to fight against decoherence. Lastly, as already mentioned in Sec. II, there also exists a wide array of strategies to minimize the number of measurements for estimating $\hat{H}$. In short, all these advanced techniques are readily available to enhance the variational part of the HDL protocol, and once we have $\hat{H}$ for the state-preparation problem we may decide which techniques to use.

V. CONCLUSIONS

In this work, we propose the HDL protocol that approximately loads a given classical data into a shallow parameterized quantum circuit. The key idea is to first construct a Hamiltonian whose ground state is exactly the normalized version of the input data, and to solve the subsequent state preparation with the variational quantum eigensolver or other methods. By shifting the majority of computational complexity to classical computing, the proposed HDL protocol gives a potentially viable quantum-classical approach in the near-term NISQ era and the early fault-tolerant period.

To demonstrate that the HDL protocol is practical for approximately loading classical data, we report numerical simulations on different kinds of input data. As clearly shown in Sec. III, a poly(n)-depth HEA circuit without ancilla qubits can efficiently amplitude encode a variety of data. Moreover, we find that many similar data, such as the MNIST handwritten images, can be efficiently encoded by a family of structurally similar Hamiltonians, i.e., comprising the same set of Pauli strings. This observation saves us a lot of time from trying to find the most compact set of $\{L_i\}$ to embed the relevant data in Hamiltonians. In this study, we also briefly investigate the trade-off between the fidelity and the circuit depth. For many tasks, especially involving machine learning, it is acceptable (and in some cases, even desirable) to lower the fidelity in exchange for low-depth circuits. At the end of Sec. III, we also explore the possibility of constructing a circuit-based QRAM to load multiple data, such as the entire data set, with the HDL protocol. We intend to develop this idea further in a future work.

While it is easy to see the benefits of a variational-based methods for quantum state preparation in the NISQ era, it is less clear which particular variational approaches would be the most ideal for state preparations. The truth could very well be that it is case dependent. However, in Sec. IV, we discuss the potential benefits of the HDL against other variational methods. Particularly, we argue that the HDL could possibly be a better approach (against methods that rely on estimating differences between probability distributions) when the quantum states are less concentrated in the Hilbert space.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (Grant No. 62101270) and the Natural Science Foundation of the Jiangsu Higher Education Institutions, China (Grant No. 21KJB520010).

APPENDIX A: HDL ALGORITHM

Complete details of the HDL protocol are elucidated in Algorithm 1.

APPENDIX B: FURTHER DISCUSSION

The relevance of small qubit systems to classical data loading is discussed here for a reference of practical applications even beyond the NISQ era.

For many practical problems such as machine learning tasks, 30 qubits or less can help us faithfully load an entire data set into a quantum computer and help us to study the impact of quantum machine learning on some conventional tasks, such as image classification. Note that the fact that the data loading takes 30 or fewer qubits does not imply that a

Algorithm 1: Hamiltonian-based data loading algorithm.

Input: a normalized classical data $\vec{x} = (x_1, x_2, \ldots, x_N) \in \mathbb{R}^N$, parameterized circuits $U(\vec{\theta})$ with initial parameters $\vec{\theta} = \vec{\theta}_0$, an initial quantum state $|\psi_0\rangle = |0^{\otimes n}\rangle$.
Output: A quantum state $|\psi\rangle = \frac{1}{|\vec{x}|}|x\rangle|i\rangle$ which is proportional to $\vec{x}$, where $|\vec{x}|$ is the normalization of the input $\vec{x}$.
Stage 1:
1. Choose a set of Hermitian $\{L_i\}$, construct the matrix $M \in \mathbb{C}^{m\times m}$ with elements $M_{ij} = \frac{1}{2}\langle\psi|\{L_i, L_j\}|\psi\rangle - \langle\psi|L_i|\psi\rangle\langle\psi|L_j|\psi\rangle$. Diagonalize the Matrix M to find at least one zero eigenvalue.
2. If the j^{th} eigenvalue of M equals to zero, the j^{th} corresponding eigenstate is assigned to the coefficients ω_i, then construct Hamiltonian $\hat{H}_0$ according to $\hat{H}_0 = \sum_i w_i L_i$.
3. If $\vec{x}$ is not the eigenstate with lowest eigenvalue of $\hat{H}_0$, reconstruct $\hat{H} = (\hat{H}_0 - eI)^2$ to make sure $\vec{x}$ is the ground state of $\hat{H}$, where $e = \langle\psi|\hat{H}_0|\psi\rangle$ and I is the identity matrix. Update $\{L_i\}$.
Stage 2:
4. Apply $U(\vec{\theta}_0)$ to state $|\psi_0\rangle$ to obtain $|\psi\rangle$, and compute energy $E_i = \langle\psi|L_i|\psi\rangle$ via L_i tests for $i = 1, ..., m$, where m is the size of $\{L_i\}$. Perform optimization to minimize the loss function $\mathcal{L}(\vec{\theta}_j) = \sum_{i=1}^m w_i E_i$ for j-th iteration.
5. Repeat 4 until loss function $\mathcal{L}(\vec{\theta}_j)$ converges with tolerance ϵ, record the learned parameters $\vec{\theta}^*$.
6. Apply $U(\vec{\theta}^*)$ to state $|\psi_0\rangle$ and then obtain the target quantum state $|\psi^*\rangle$.

quantum machine learning model is limited to this number of qubits.

The following are some concrete examples which claim that many scenarios for classical data loading can be capped at 30 qubits or so. For instance, we can load (1) $\sim 10^3$ data points of grayscale CIFAR-10 images (32×32 pixels) using 20 qubits, (2) $\sim 10^6$ data points of CIFAR-10-like (32×32 pixels) images using 30 qubits, and (3) $\sim 10^3$ data points of grayscale ImageNet images (256×256 pixels) using 26 qubits.

For example, the average resolution of a grayscale ImageNet image is 469×387 pixels, and each image can be exactly amplitude-encoded by 18 qubits if using our proposed HDL scheme. Moreover, in practice, the images are often cropped to 256×256 or 224×224 pixels, which implies amplitude encoding with 16 qubits. Another equally important image data set in the history of the development of classical machine learning is CIFAR-10, which collects images of size 32×32 (implying ten-qubit encoding). A reasonably large data set for research purposes can be taken on the order of $10^5 \sim 10^6$ (which implies $16 \sim 20$ qubits for address), but for most contemporary quantum machine learning studies, the number of training data points is often orders of magnitudes less, say around $1000 \sim 8000$ data points (which implies $10 \sim 13$ qubits). In short, our data-loading scheme can facilitate state-of-the-art quantum machine learning research with just $20 \sim 30$ qubits in the foreseeable future. The range of data size and data quality (i.e., retaining the full resolutions of image data) we discuss here should provide a realistic test bed to study the nuance of whether or not quantum machine learning models may potentially outperform sophisticated deep-learning models for some specific tasks. It is not hard to anticipate that this kind of experimental investigation will continue from the NISQ era through the early fault-tolerant period, if not further along the time line of quantum computing development.

[1] P. Rebentrost, M. Mohseni, and S. Lloyd, Quantum Support Vector Machine for Big Data Classification, Phys. Rev. Lett. **113**, 130503 (2014).

[2] B. Duan, J. Yuan, Y. Liu, and D. Li, Quantum algorithm for support matrix machines, Phys. Rev. A **96**, 032301 (2017).

[3] S. Lloyd, M. Mohseni, and P. Rebentrost, Quantum principal component analysis, Nat. Phys. **10**, 631 (2014).

[4] I. Cong and L. Duan, Quantum discriminant analysis for dimensionality reduction and classification, New J. Phys. **18**, 073011 (2016).

[5] B. Duan, J. Yuan, J. Xu, and D. Li, Quantum algorithm and quantum circuit for a-optimal projection: Dimensionality reduction, Phys. Rev. A **99**, 032311 (2019).

[6] I. Kerenidis, J. Landman, A. Luongo, and A. Prakash, q-Means: A quantum algorithm for unsupervised machine learning, in *Advances in Neural Information Processing Systems*, Vol. 32 (Curran Associates, Inc., 2019).

[7] I. Kerenidis and A. Prakash, Quantum recommendation systems, in *8th Innovations in Theoretical Computer Science Conference (ITCS 2017), Vol. 67 of Leibniz International Proceedings in Informatics (LIPIcs)* (2017), pp. 49:1–49:21.

[8] V. Giovannetti, S. Lloyd, and L. Maccone, Quantum Random Access Memory, Phys. Rev. Lett. **100**, 160501 (2008).

[9] V. Giovannetti, S. Lloyd, and L. Maccone, Architectures for a quantum random access memory, Phys. Rev. A **78**, 052310 (2008).

[10] F.-Y. Hong, Y. Xiang, Z.-Y. Zhu, L.-Z. Jiang, and L.-N. Wu, Robust quantum random access memory, Phys. Rev. A **86**, 010306 (2012).

[11] S. Arunachalam, V. Gheorghiu, T. Jochym-O'Connor, M. Mosca, and P. V. Srinivasan, On the robustness of bucket brigade quantum ram, New J. Phys. **17**, 123010 (2015).

[12] O. Di Matteo, V. Gheorghiu, and M. Mosca, Fault tolerant resource estimation of quantum random-access memories, IEEE Trans. Quantum Eng. **1**, 1 (2020).

[13] D. K. Park, F. Petruccione, and J.-K. K. Rhee, Circuit-based quantum random access memory for classical data, Sci. Rep. **9**, 3949 (2019).

[14] J. Bang, A. Dutta, S.-W. Lee, and J. Kim, Optimal usage of quantum random access memory in quantum machine learning, Phys. Rev. A **99**, 012326 (2019).

[15] C. T. Hann, C.-L. Zou, Y. Zhang, Y. Chu, R. J. Schoelkopf, S. M. Girvin, and L. Jiang, Hardware-Efficient Quantum Random Access Memory with Hybrid Quantum Acoustic Systems, Phys. Rev. Lett. **123**, 250501 (2019).

[16] X. Sun, G. Tian, S. Yang, P. Yuan, and S. Zhang, Asymptotically optimal circuit depth for quantum state preparation and general unitary synthesis, arXiv:2108.06150.

[17] S. Johri, S. Debnath, A. Mocherla, A. Singk, A. Prakash, J. Kim, and I. Kerenidis, Nearest centroid classification on a trapped ion quantum computer, npj Quantum Inf. **7**, 122 (2021).

[18] X.-M. Zhang, T. Li, and X. Yuan, Quantum state preparation with optimal circuit depth: Implementations and applications, arXiv:2201.11495.

[19] X.-M. Zhang, M.-H. Yung, and X. Yuan, Low-depth quantum state preparation, Phys. Rev. Res. **3**, 043200 (2021).

[20] K. Nakaji, S. Uno, Y. Suzuki, R. Raymond, T. Onodera, T. Tanaka, H. Tezuka, N. Mitsuda, and N. Yamamoto, Approximate amplitude encoding in shallow parameterized quantum circuits and its application to financial market indicators, Phys. Rev. Res. **4**, 023136 (2022).

[21] M. Möttönen, J. Vartiainen, V. Bergholm, and M. Salomaa, Transformation of quantum states using uniformly controlled rotations, Quant. Inf. Comput. **5** (2005).

[22] I. F. Araujo, D. K. Park, F. Petruccione, and A. J. da Silva, A divide-and-conquer algorithm for quantum state preparation, Sci. Rep. **11**, 1 (2021).

[23] T. Shirakawa, H. Ueda, and S. Yunoki, Automatic quantum circuit encoding of a given arbitrary quantum state, arXiv:2112.14524.

[24] G. Marin-Sanchez, J. Gonzalez-Conde, and M. Sanz, Quantum algorithms for approximate function loading, arXiv:2111.07933.

[25] J. Romero and A. Aspuru-Guzik, Variational quantum generators: Generative adversarial quantum machine learning for continuous distributions, Adv. Quantum Technol. **4**, 2000003 (2021).

[26] C. Zoufal, A. Lucchi, and S. Woerner, Quantum generative adversarial networks for learning and loading random distributions, npj Quantum Inf. **5**, 103 (2019).

[27] A. M. Gomez, S. F. Yelin, and K. Najafi, Reconstructing quantum states using basis-enhanced born machines, arXiv:2206.01273.

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

[29] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, Parameterized quantum circuits as machine learning models, Quantum Sci. Technol. **4**, 043001 (2019).

[30] X.-L. Qi and D. Ranard, Determining a local Hamiltonian from a single eigenstate, Quantum **3**, 159 (2019).

[31] 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).

[32] J. Stokes, J. Izaac, N. Killoran, and G. Carleo, Quantum natural gradient, Quantum **4**, 269 (2020).

[33] H.-Y. Huang, R. Kueng, and J. Preskill, Predicting many properties of a quantum system from very few measurements, Nat. Phys. **16**, 1050 (2020).

[34] S. Chen, W. Yu, P. Zeng, and S. T. Flammia, Robust shadow estimation, PRX Quantum **2**, 030348 (2021).

[35] C. Hadfield, S. Bravyi, R. Raymond, and A. Mezzacapo, Measurements of quantum Hamiltonians with locally-biased classical shadows, Commun. Math. Phys. **391**, 951 (2022).

[36] R. G. Melko, G. Carleo, J. Carrasquilla, and J. I. Cirac, Restricted Boltzmann machines in quantum physics, Nat. Phys. **15**, 887 (2019).

[37] G. Torlai, G. Mazzola, G. Carleo, and A. Mezzacapo, Precise measurement of quantum observables with neural-network estimators, Phys. Rev. Res. **2**, 022060 (2020).

[38] A. Shlosberg, A. J. Jena, P. Mukhopadhyay, J. F. Haase, F. Leditzky, and L. Dellantonio, Adaptive estimation of quantum observables, arXiv:2110.15339.

[39] 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).

[40] C. Hempel, C. Maier, J. Romero, J. McClean, T. Monz, H. Shen, P. Jurcevic, B. P. Lanyon, P. Love, R. Babbush *et al.*, Quantum Chemistry Calculations on a Trapped-Ion Quantum Simulator, Phys. Rev. X **8**, 031022 (2018).

[41] Y. Nam, J.-S. Chen, N. C. Pisenti, K. Wright, C. Delaney, D. Maslov, K. R. Brown, S. Allen, J. M. Amini, J. Apisdorf *et al.*, Ground-state energy estimation of the water molecule on a trapped-ion quantum computer, npj Quantum Inf. **6**, 1 (2020).

[42] 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).

[43] W. J. Huggins, J. R. McClean, N. C. Rubin, Z. Jiang, N. Wiebe, K. B. Whaley, and R. Babbush, Efficient and noise resilient measurements for quantum chemistry on near-term quantum computers, npj Quantum Inf. **7**, 23 (2021).

[44] P. Gokhale, O. Angiuli, Y. Ding, K. Gui, T. Tomesh, M. Suchara, M. Martonosi, and F. T. Chong, Minimizing state preparations in variational quantum eigensolver by partitioning into commuting families, arXiv:1907.13623.

[45] V. Verteletskyi, T.-C. Yen, and A. F. Izmaylov, Measurement optimization in the variational quantum eigensolver using a minimum clique cover, J. Chem. Phys. **152**, 124114 (2020).

[46] R. Kondo, Y. Sato, S. Koide, S. Kajita, and H. Takamatsu, Computationally efficient quantum expectation with extended bell measurements, Quantum **6**, 688 (2022).

[47] A. F. Izmaylov, T.-C. Yen, R. A. Lang, and V. Verteletskyi, Unitary partitioning approach to the measurement problem in the variational quantum eigensolver method, J. Chem. Theory Comput. **16**, 190 (2020).

[48] V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, M. S. Alam, S. Ahmed, J. M. Arrazola, C. Blank, A. Delgado, S. Jahangiri *et al.*, Pennylane: Automatic differentiation of hybrid quantum-classical computations, arXiv:1811.04968.

[49] P. Yuan and S. Zhang, Optimal qram and improved unitary synthesis by quantum circuits with any number of ancillary qubits, arXiv:2202.11302.

[50] L. Bittel and M. Kliesch, Training Variational Quantum Algorithms Is NP-Hard, Phys. Rev. Lett. **127**, 120502 (2021).

[51] S.-X. Zhang, C.-Y. Hsieh, S. Zhang, and H. Yao, Differentiable quantum architecture search, Quantum Sci. Tech. **7**, 045023 (2022).

[52] S.-X. Zhang, C.-Y. Hsieh, S. Zhang, and H. Yao, Neural predictor based quantum architecture search, Mach. Learn.: Sci. Technol. **2**, 045027 (2021).

[53] H. L. Tang, V. Shkolnikov, G. S. Barron, H. R. Grimsley, N. J. Mayhall, E. Barnes, and S. E. Economou, Qubit-adapt-VQE: An adaptive algorithm for constructing hardware-efficient ansätze on a quantum processor, PRX Quantum **2**, 020310 (2021).

[54] M. Bilkis, M. Cerezo, G. Verdon, P. J. Coles, and L. Cincio, A semi-agnostic ansatz with variable structure for quantum machine learning, arXiv:2103.06712.

[55] R. Wiersema, C. Zhou, Y. de Sereville, J. F. Carrasquilla, Y. B. Kim, and H. Yuen, Exploring entanglement and optimization within the Hamiltonian variational ansatz, PRX Quantum **1**, 020319 (2020).

[56] A. A. Mele, G. B. Mbeng, G. E. Santoro, M. Collura, and P. Torta, Avoiding barren plateaus via transferability of smooth solutions in Hamiltonian variational ansatz, arXiv:2206.01982.

[57] Y.-C. Chen, Y.-Q. Chen, A. Hu, C.-Y. Hsieh, and S. Zhang, Variational quantum simulation of the imaginary-time Lyapunov control for accelerating the ground-state preparation, arXiv:2112.11782.

[58] W. M. Kirby and P. J. Love, Variational Quantum Eigensolvers for Sparse Hamiltonians, Phys. Rev. Lett. **127**, 110503 (2021).

[59] T. Fösel, M. Y. Niu, F. Marquardt, and L. Li, Quantum circuit optimization with deep reinforcement learning, arXiv:2103.07585.

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

[61] 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).

[62] E. R. Bennewitz, F. Hopfmueller, B. Kulchytskyy, J. Carrasquilla, and P. Ronagh, Neural error mitigation of near-term quantum simulations, Nat. Mach. Intell. **4**, 618 (2022).

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

[64] G. S. Ravi, K. N. Smith, P. Gokhale, A. Mari, N. Earnest, A. Javadi-Abhari, and F. T. Chong, Vaqem: A variational approach to quantum error mitigation, in *2022 IEEE International Symposium on High- Performance Computer Architecture (HPCA)* (IEEE, New York, 2022), pp. 288–303.

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