Efficient Variational Quantum Simulator Incorporating Active Error Minimization

Ying Li and Simon C. Benjamin

Department of Materials, University of Oxford, Parks Road, Oxford OX1 3PH, United Kingdom
(Received 14 December 2016; revised manuscript received 17 April 2017; published 29 June 2017)

One of the key applications for quantum computers will be the simulation of other quantum systems that arise in chemistry, materials science, etc., in order to accelerate the process of discovery. It is important to ask the following question: Can this simulation be achieved using near-future quantum processors, of modest size and under imperfect control, or must it await the more distant era of large-scale fault-tolerant quantum computing? Here, we propose a variational method involving closely integrated classical and quantum coprocessors. We presume that all operations in the quantum coprocessor are prone to error. The impact of such errors is minimized by *boosting* them artificially and then extrapolating to the zero-error case. In comparison to a more conventional optimized Trotterization technique, we find that our protocol is efficient and appears to be fundamentally more robust against error accumulation.

I. INTRODUCTION

Many quantum algorithms have been developed under the presumption that the hardware upon which they will run is effectively error free: The error rate is so low that the entire algorithm can be executed successfully without a single error. It is now known that such hardware can, in principle, be created using components that have far higher error rates. Fault-tolerant quantum computing can be achieved by encoding qubits in non-Abelian anyons in topological materials [1] or using the quantum error correction codes [2]. While the former is still in the early stages of its development, for the latter approach, sub-threshold quantum operations have been demonstrated in ion-trap and superconducting systems [3–6]. However, quantum error correction involves a substantial multiplication of resources; the number of physical qubits required may be orders of magnitude greater than the number of error-free logical qubits seen by the algorithm. A recent study audited the cost of implementing Shor's algorithm to solve a classically infeasible task and found that, even with state-of-the-art techniques for magic-state distillation, the machine would need over six million of today's highest-quality qubits [7].

The need for millions of qubits contrasts starkly with the fact that only 50 qubits are needed to achieve so-called "quantum supremacy," i.e., to create a quantum processor that is so complex that conventional supercomputers [8] cannot predict its behavior. Machines involving this many qubits, under good but imperfect control, are expected to emerge in the next few years. The challenge for researchers is to identify useful functions for such devices, in order to motivate further investment and continue the evolution toward the longer-term goal of fully fault-tolerant systems.

Recently, some hybrid quantum/classical algorithms have been developed which are promising for near future quantum applications [9–17]. A common feature of these algorithms is that the quantum computer is only in charge of carrying out a subroutine, acting as a "coprocessor" while the larger-scale algorithm is governed by a classical computer. The task of the quantum computer is thus simplified and may be accomplished with relatively few quantum operations. A higher error rate per operation is therefore tolerable; in a fault-tolerant machine, this would imply a more modest resource overhead for the code, but it may even be possible to implement such quantum algorithms without quantum error correction.

Hybrid approaches are very relevant to quantum simulation, i.e., Feynman's vision [18,19] of using a controlled quantum processor to model another quantum system. Such a technology would be highly advantageous for the investigation of various large quantum systems, e.g., simulating quantum chemistry systems [20–23], or novel materials and other condensed-matter systems [24–26]. A powerful tool that has been exploited in several hybrid protocols is the variational method [9–12]. Typically, the state of the target system can be found by writing a trial quantum state with a large but tractable number of parameters, and then discovering the optimal value of these parameters. Implicitly, this requires the scientists to use their understanding of the target system (the novel molecule, or material) to select a set of parameters that, while large, is far smaller than the total number of parameters needed to specify an arbitrary quantum state. The latter is of course exponential in the number of particles composing the target system.

Our focus here is on finding the dynamics of interesting quantum systems, and we briefly remark on the considerable significance of such a capability. Dynamics must be studied when properties cannot be determined from static features. This has motivated dynamical versions of many well-known techniques, e.g., nonequilibrium dynamical mean-field theory [27], the time-dependent variational quantum Monte Carlo method [28], time-dependent tensor network methods [29,30], and, of course, time-dependent density functional theory [31]. However, there are still many problems that cannot be solved using these powerful classical methods, so it is hoped that quantum computers can extend their reach [18,19,32].

We therefore propose a hybrid quantum algorithm for simulating the dynamics of a quantum system. The conventional approach for simulating quantum dynamics employs Trotterization [19,33–36], which usually requires many quantum operations; therefore, it seems likely to necessitate the full machinery of fault-tolerant quantum computing [20–22]. Our approach is based on the variational method, and our hope is that it could be implemented using small-size quantum circuits, i.e., quantum circuits with a small number of quantum operations that suffer significant noise compared with fault-tolerant quantum computers. A novel feature of our algorithm is that it compensates for errors through classical inference without encoding: If the noise in the quantum computer mainly results in stochastic errors and the rate of errors can be amplified in a controllable way, then we find that errors can be approximately corrected. The condition that noise is stochastic can be met by engineering for many systems: If, for example, single qubit gates have relatively high fidelity [3–6,37] and their noise is stochastic, then arbitrary two-qubit gate errors can be made stochastic through a technique akin to twirling [38–40], which we presently discuss. Moreover, the severity of such errors can be deliberately *increased* artificially, allowing one to create curves that the classical algorithm can extrapolate to estimate the zero-error limit. We performed numerical emulations of the process on small systems, finding that this technique does indeed lead to robustness: The impact of physical errors on the simulator's performance is far lower than in an (optimized) Trotterization protocol, and moreover, this impact does not worsen with the duration of the simulation.

The remainder of this paper is organized as follows. In Sec. II, we review the Trotterization algorithm and variational methods. In Sec. III, our hybrid algorithm is introduced. In Sec. IV, the variational theory is discussed. In Sec. V, the task for the quantum computer and the overall program are described in detail. Errors in our algorithm are analyzed in Sec. VI. The method for reducing errors is given in Sec. VII, in which we also discuss how to convert nonstochastic errors into stochastic errors and how to tune the rate of errors. Numerical results are presented in Sec. VIII. A summary is given in Sec. IX.

II. TROTTERIZATION AND VARIATIONAL METHOD

The Trotterization approach to simulation, which we use as a basis for comparison with our protocol, exploits the fact that time evolution under a general Hamiltonian $H = \sum_j H_j$ can be approximated according to the Trotter-Suzuki decomposition [33]

$$e^{-iHT} \simeq R = \prod_{n=1}^{N_t} \left(\prod_j e^{-iH_j \tau_{n,j}} \right). \tag{1}$$

Here, each term $e^{-iH_j \tau_{n,j}}$ corresponds to the evolution driven by the term H_j for a short time $\tau_{n,j}$, which can be realized by a quantum gate or a combination of quantum gates. Usually, the short time is taken uniformly as $\tau_{n,j} = T/N_t$, where T is the time of the simulated evolution. When N_t is larger, the approximation is better, and errors in the approximation scale with the simulated time and the number of quantum gates as T^2/N_t [35].

Our approach is based on a variational technique. Variational methods have numerous applications in the numerical study of many-body quantum systems: for example, density functional theory [41], the matrix product state method [42], and simulating molecular dynamics using the variational principle [43]. In these methods, typically a trial state is used to approximate the true state of the system. The trial state must, of course, be specified by some tractable number of parameters. But since existing realizations are entirely classical, there is a stronger condition on the trial function: It must be possible to efficiently evaluate its fit to the true quantum state using only a classical algorithm. This requirement limits the application of variational methods. Sometimes, it may be impossible to evaluate a trial state that provides a good approximation to the true state in a classical computer. In such a case, a quantum computer could be helpful because we may be able to complete tasks that are difficult for a classical computer using a quantum computer. An example is the unitary coupled cluster method [10,44], in which the energy of the trial state can be evaluated using a quantum computer when it is hard for a classical computer to do so. The protocol we describe here is another example.

III. HYBRID QUANTUM SIMULATION OF DYNAMICS

The purpose of our hybrid algorithm is to solve the Schrödinger equation $i[\partial/(\partial t)]|\Phi(t)\rangle = H|\Phi(t)\rangle$ ($\hbar = 1$), assuming that the state $|\Phi(t)\rangle$ can be approximated by a trial state $|\Psi(t)\rangle \equiv |\Psi(\lambda_1, \lambda_2, \ldots)\rangle$, where $\{\lambda_k(t)\}$ are variational parameters. As shown in Fig. 1(a), the hybrid algorithm is implemented on both a quantum computer and a classical computer. The task of the classical computer is to determine variational parameters according to the
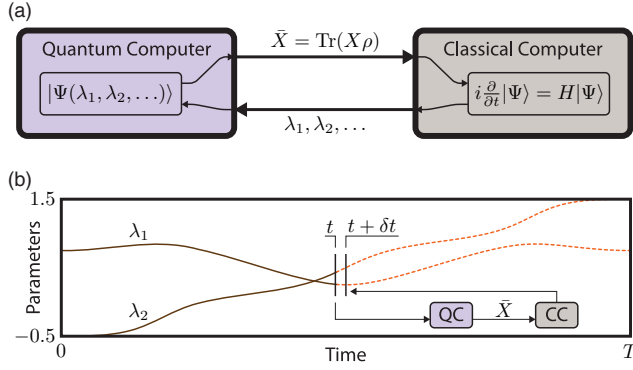
FIG. 1. Hybrid solver of quantum dynamics. (a) Both a quantum computer and a classical computer are used in solving the time evolution of a quantum system. The quantum state is approximated by a trial state $|\Psi(\lambda_1, \lambda_2, \ldots)\rangle$. Variational parameters are determined by the classical computer according to the Schrödinger equation. The quantum computer is used to implement a subroutine: Inputs are parameters $\{\lambda_k(t)\}$, and outputs are values of certain derivatives required by the main program in the classical computer. (b) Variational parameters are determined iteratively given their initial values. Parameters at the time t are sent to the quantum computer, which evaluates the quantities required by the classical computer. Then, the classical computer can work out parameters at the time $t + \delta t$, where δt is a short time. Note that the curves here represent the actual evolution of parameters in the example described in Appendix A.

Schrödinger equation, and this procedure requires certain derivatives associated with the state $|\Psi(t)\rangle$, which the quantum computer provides.

The hybrid algorithm works out variational parameters iteratively, as shown in Fig. 1(b). Parameters at the time t ($\{\lambda_k(t)\}$) are sent to the quantum computer, with which the quantum computer finds the values required by the classical computer. Based on results from the quantum computer, the classical computer can determine parameters at the time $t + \delta t$ ($\{\lambda_k(t + \delta t)\}$), where δt is a short time. Then, these new parameters are sent back to the quantum computer. In this way, given parameters of the initial state ($\{\lambda_k(0)\}$), parameters at the time T ($\{\lambda_k(T)\}$) are systematically inferred by iterating the process carried out by two computers. The simulation is successful if the state $|\Psi(T)\rangle$ is a good approximation of the state $|\Phi(T)\rangle$.

Using the variational method, the degrees of freedom provided by variational parameters allow us to use quantum circuits with a size much smaller than the circuit of the Trotterization algorithm to simulate the time evolution of a quantum system. Note that this is an "apples to oranges" comparison because our algorithm only simulates the time evolution of a given initial state while the Trotterization algorithm simulates the time evolution of arbitrary initial states, i.e., the time evolution operator. Thus, our algorithm aims at an easier problem than the Trotterization algorithm.

Tracking the evolution from a specific initial state is the main goal in many simulations, and other more general

tasks can also be reexpressed this way. The approach we describe can be relevant to the specific problem of designing and calibrating quantum gates, thus allowing early quantum computers to aid in the design of their successors. Moreover, there are also interesting connections between dynamical simulation and the determination of static properties: One could find a ground state by simulating an adiabatic time evolution [45]; thus, our algorithm is relevant to that task. In other hybrid algorithms for determining the ground state of a quantum system [9–12], one may need to find the global minimum of the energy in the parameter space to maximize the fidelity. However, finding the global minimum in a high-dimensional parameter space is usually a nontrivial computing task. In our algorithm, parameters are worked out iteratively; therefore, the global minimization is not required. We remark that Trotterization is used in some hybrid algorithms [10,12–15]. In principle, our algorithm can be used to replace the Trotterization method in these instances, to further simplify the task of the quantum computer.

IV. VARIATIONAL THEORY OF QUANTUM TIME EVOLUTION

The time-dependent variational principle corresponding to the Schrödinger equation reads $\delta \int_{t_i}^{t_f} dt L = 0$, where the Lagrangian is [46,47]

$$L = \langle \Psi(t) | \left(i \frac{\partial}{\partial t} - H \right) | \Psi(t) \rangle. \tag{2}$$

Assuming that the state $|\Psi(t)\rangle$ is determined by a set of real parameters $\{\lambda_k(t)\}$, i.e., $|\Psi(t)\rangle \equiv |\Psi(\lambda_1, \lambda_2, \ldots)\rangle$, the Lagrangian can be rewritten as

$$L = i \sum_k \langle \Psi | \frac{\partial |\Psi\rangle}{\partial \lambda_k} \dot{\lambda}_k - \langle \Psi | H | \Psi \rangle, \tag{3}$$

which is a function of parameters $\{\lambda_k\}$ and their time derivatives $\{\dot{\lambda}_k = [(d\lambda_k)/dt]\}$. According to L, the Euler-Lagrange equation describing the evolution of parameters (hence, the state $|\Psi\rangle$) is

$$\sum_q M_{k,q} \dot{\lambda}_q = V_k, \tag{4}$$

where

$$M_{k,q} = i\eta \frac{\partial \langle \Psi |}{\partial \lambda_k} \frac{\partial |\Psi\rangle}{\partial \lambda_q} + \text{H.c.}, \tag{5}$$

$$V_k = \eta \frac{\partial \langle \Psi |}{\partial \lambda_k} H | \Psi \rangle + \text{H.c.} \tag{6}$$

Here, $\eta = 1$, both M and V are real, and M is antisymmetric. There are other variational principles for the quantum time evolution [48]. For example, McLachlan's variational principle reads $\delta\|\{i[\partial/(\partial t)] - H\}|\Psi(t)\rangle\| = 0$ [49], which leads to the same equation as Eq. (4) but $\eta = -i$. Here, the norm is $\|\psi\| = \sqrt{\langle\psi|\psi\rangle}$. In the following, we focus on the time-dependent variational principle, but the hybrid algorithm can be adapted to McLachlan's variational principle.

Recall that we can always express a state as $|\Psi\rangle = \sum_n (\alpha_n + i\beta_n)|n\rangle$, where α_n and β_n are real, and $|n\rangle$ are orthonormal basis states. Taking parameters $\{\lambda_k\} = \{\alpha_n, \beta_n\}$, Eq. (4) leads to the Schrödinger equation; but, of course, we require a parametrization such that the number of parameters remains tractable for the sizes of target systems that we are interested in. Thus, our variational approach, like others, is relevant when the scientist can make an educated guess as to the general form of the quantum state—there can be a large number of free parameters $\{\lambda_k\}$ but typically far fewer than would be needed to specify an arbitrary state.

V. VARIATIONAL ALGORITHM ON A HYBRID COMPUTER

We consider trial states that can be directly prepared in the quantum computer; i.e., states can be expressed as $|\Psi\rangle = R|\bar{0}\rangle$, where $|\bar{0}\rangle$ is an initial state of the quantum computer, and R is a sequence of quantum gates determined by parameters $\{\lambda_k\}$, i.e.,

$$R = R_{N_v}(\lambda_{N_v})R_{N_v-1}(\lambda_{N_v-1})\ldots R_2(\lambda_2)R_1(\lambda_1). \quad (7)$$

Here, R_k is a unitary operator describing a quantum gate, and the total number of gates (i.e., parameters) is N_v. If N_v is smaller than the dimension of the Hilbert space ($2 \times \dim -2$ to be exact), trial states $|\Psi\rangle$ only span a submanifold of the Hilbert space. In this restricted trial-state space, Eq. (4) approximates the exact evolution if the exact state is close to the trial-state space.

In the following analysis, we describe each R_k gate as dependent on only one parameter λ_k. However, it is worth noting that the trial state can be generalized to the case where each gate R_k depends on multiple parameters, including parameters that vary in a predefined way with time, and that both the Trotter-Suzuki decomposition [33] and the unitary coupled cluster ansatz [44] can be expressed in this form. As a generalization to the case in which the number of gates N_v is fixed, one can even vary N_v depending on the simulated time, providing that we understand how to reexpress the trial state using the new gates (adding gates to the set is, of course, trivially possible). Our point is that the set of gates with which we create our trial state can itself evolve over the simulated time.

We are interested in the case where evaluating coefficients M and V in Eq. (4) is intractable in a classical computer; therefore, these coefficients are obtained using the quantum computer. Each parameter is determined, in turn, by appropriately configuring a quantum circuit involving about N_v gates and a single measurement outcome; this fixed circuit is run repeatedly until the expected measurement outcome is known to a given precision. Note that this implies the overall protocol is trivially parallelizable over a large number of quantum processors with no quantum link between them.

We express the Hamiltonian in the form

$$H = \sum_i h_i \sigma_i, \quad (8)$$

where σ_i are unitary operators. In many quantum systems, the number of terms in this expression scales with the size of the system polynomially. Similarly, we write

$$\frac{dR_k}{d\lambda_k} = \sum_i f_{k,i} R_k \sigma_{k,i}, \quad (9)$$

where $\sigma_{k,i}$ are also unitary operators. For many frequently used single-qubit gates and two-qubit gates (R_k), there is only one term in this expression, and $\sigma_{k,i}$ is also a one-qubit or two-qubit gate. Because any operator can be expressed using Pauli operators, we can choose unitary operators σ_i and $\sigma_{k,i}$ as (single-qubit and multiqubit) Pauli operators.

Using the expression (9), we rewrite the derivative of the state as

$$\frac{\partial|\Psi\rangle}{\partial\lambda_k} = \sum_i f_{k,i} R_{k,i}|\bar{0}\rangle, \quad (10)$$

where

$$R_{k,i} = R_{N_v} R_{N_v-1}\ldots R_{k+1} R_k \sigma_{k,i}\ldots R_2 R_1. \quad (11)$$

Then, differential equation coefficients can be expressed as

$$M_{k,q} = \sum_{i,j}(if^*_{k,i}f_{q,j}\langle\bar{0}|R^\dagger_{k,i}R_{q,j}|\bar{0}\rangle + \text{H.c.}) \quad (12)$$

and

$$V_k = \sum_{i,j}(f^*_{k,i}h_j\langle\bar{0}|R^\dagger_{k,i}\sigma_j R|\bar{0}\rangle + \text{H.c.}), \quad (13)$$

where we have used the expression (8).

In Eqs. (12) and (13), each term is in the form

$$a\text{Re}(e^{i\theta}\langle\bar{0}|U|\bar{0}\rangle),$$

where the amplitude a and phase θ are determined by either $if^*_{k,i}f_{q,j}$ or $f^*_{k,i}h_j$ [50], and U is a unitary operator equal to either $R^\dagger_{k,i}R_{q,j}$ or $R^\dagger_{k,i}\sigma_j R$. Such a term can be evaluated using the quantum circuit shown in Fig. 2. This circuit needs an ancillary qubit initialized in the state $(|0\rangle + e^{i\theta}|1\rangle)/\sqrt{2}$ and a register initialized in the state $|\bar{0}\rangle$.
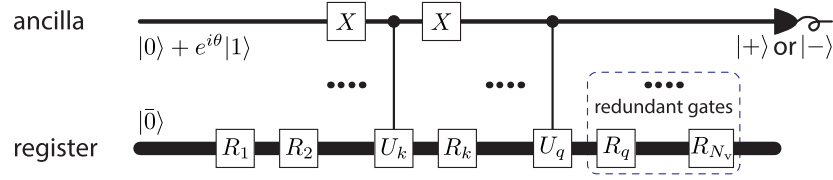
FIG. 2.   Quantum circuit for the evaluation of certain coefficients required by the classical main program, as specified in the text. To evaluate $\mathrm{Re}(e^{i\theta}\langle\bar{0}|U|\bar{0}\rangle)$, where $U = R_1^\dagger \ldots U_k^\dagger R_k^\dagger \ldots R_{N_v}^\dagger R_{N_v} \ldots R_q U_q \ldots R_1$, the ancillary qubit is initialized in the state $(|0\rangle + e^{i\theta}|1\rangle)/\sqrt{2}$ and measured in the $|\pm\rangle = (|0\rangle \pm |1\rangle)/\sqrt{2}$ basis. Here, U_k is one of $\sigma_{k,i}$, and U_q is one of $\sigma_{q,j}$ or σ_j (by taking $q = N_v + 1$, σ_j is put on the left side of R_{N_v} in the product). In the figure, we have assumed that $k < q$. Gates on the register after the second controlled unitary gate can be omitted. This circuit is actually a variant of the circuit proposed in 2002 by Ekert *et al.* [52,53]. It involves N_v gates on the register, two flip gates (X) on the ancillary qubit, and two controlled unitary gates on the ancillary qubit and the register.

The ancillary qubit is measured in the $\{|+\rangle, |-\rangle\}$ basis after a sequence of quantum gates on the register and two controlled gates, in which the ancillary qubit is the control qubit. The value of each term is given by $\mathrm{Re}(e^{i\theta}\langle\bar{0}|U|\bar{0}\rangle) = \langle X \rangle = \mathrm{Tr}(X\rho)$, where ρ is the final state of the quantum computer, and X is the x-direction Pauli operator of the ancillary qubit. In the following, we consider the case in which the value of $\langle X \rangle$ is estimated by repeating this relatively shallow circuit and calculating the mean value of measurement outcomes. Note that the value of $\langle X \rangle$ could be estimated more efficiently using quantum amplitude estimation [51] if error rates could be made low enough to allow a circuit of sufficient depth to function.

A. Main program

The overall flow of the algorithm is as follows: First, we select initial parameters $\{\lambda_k(0)\}$. Second, we solve the differential equation (4) numerically using the classical computer, in which the matrix M and the vector V in the equation are evaluated using the quantum coprocessor. The solution permits us to project our parameters forward by a small time increment and repeat the second step. Eventually, we reach the parameters $\{\lambda_k(T)\}$, which allow us to prepare the final state in the quantum computer.

There are many different numerical methods for solving a differential equation, and the choice of specific numerical method determines the details of the information exchange loop between quantum and classical processors. In the following, we take the Euler method as an example, but the algorithm can be adapted to other numerical methods, e.g., Runge-Kutta methods [54].

Time is discretized as $t_n = n\delta t$, where $t_0 = 0$ is the initial time, and $t_N = N\delta t = T$ is the simulated evolution time. First, M and V corresponding to parameters $\{\lambda_k(t_0)\}$ are evaluated using the quantum computer. Then, the following process is repeated. Given M and V corresponding to the time t_n, Eq. (4) is solved numerically on the classical computer to obtain values of $\{\dot{\lambda}_k(t_n)\}$. As parameters $\{\lambda_k(t_n)\}$ have been obtained from previous calculations, one can approximately calculate parameters of the time t_{n+1} using $\lambda_k(t_{n+1}) = \lambda_k(t_n) + \dot{\lambda}_k\delta t$. Repeating the process until $t_{n+1} = T$, we can work out the parameters $\{\lambda_k(T)\}$ of the final state.

VI. ERROR ANALYSIS

There are four types of errors that can result in infidelity in the variational quantum simulation: (i) errors due to limited generality of the trial wave function, which may only be able to describe the simulated system approximately; (ii) errors in the numerical integration obtained by solving Eq. (4), which is always approximate because of the finite discretization of time; (iii) shot noise in measuring equation coefficients M and V; and (iv) errors due to noise in the quantum machine, e.g., decoherence and quantum gate infidelity.

A good trial wave function allows us to not only reduce trial wave-function errors but also minimize the difficulty of implementing the algorithm, e.g., only use small-size quantum circuits. Whether a good trial wave function can be found depends on the simulated system and our understanding of the physics in that system. However, a trial wave function that contains a polynomial number of parameters and is a high-fidelity approximation to the true wave function always exists. For example, one can set the trial wave function in the Trotter-Suzuki form, i.e., take the sequence of gate operations R in the form of Eq. (1), and then take the evolution time of each term as a variational parameter ($\{\lambda_k\} = \{\tau_{n,j}\}$) rather than a fixed value as in the Trotterization algorithm [11]. Using the Trotter-Suzuki-form trial wave function, we know that the severity of trial wave-function errors decreases with the number of Trotterization slices N_t as "Error $\propto 1/N_t$" [35] in the worst case.

Integration errors depend on the numerical method for solving the differential equation (4). We take the Euler method as an example. In the Euler method, the severity of error is proportional to the size of each step δt [54]. Therefore, by choosing a small step size, integration errors can be suppressed.

Shot noise and machine noise occur in the implementation of the algorithm, while trial wave-function errors and integration errors are due to the imperfection of the

algorithm itself. The effect of implementation errors in the integration process is that coefficients M and V evaluated using the quantum computer are inaccurate; i.e., their values are different from their true values $M^{(0)}$ and $V^{(0)}$ given by Eqs. (12) and (13) (that is, the values given by a quantum computer without shot noise and machine noise). If coefficients M and V are inaccurate, time derivatives $\{\dot{\lambda}_k\}$ obtained from M and V [see Eq. (4)] are different from their true values $\{\dot{\lambda}_k^{(0)}\}$ obtained from $M^{(0)}$ and $V^{(0)}$. Using $\delta M = M - M^{(0)}$, $\delta V = V - V^{(0)}$, and $\delta\dot{\lambda} = \dot{\lambda} - \dot{\lambda}^{(0)}$ to denote deviations from true values and using $\dot{\lambda} = g(M, V)$ to denote the solution of Eq. (4), we have $\delta\dot{\lambda}_k \simeq \sum_{p,q}[\partial g_k/(\partial M_{p,q})]\delta M_{p,q} + \sum_q[\partial g_k/(\partial V_q)]\delta V_q$. When the matrix is invertible in the vicinity of $M^{(0)}$, $\delta\dot{\lambda} \simeq -(1/M^{(0)})\delta M \dot{\lambda}^{(0)} + (1/M^{(0)})\delta V$.

A. Trace distance

To analyze errors in the hybrid algorithm, we use the trace distance $D(\rho, \rho') = \frac{1}{2}\mathrm{Tr}|\rho - \rho'|$ [55] as the measure of error severity. The degree of error in the overall process is given by $D(|\Phi(t_N)\rangle, \rho_N)$. Here, $|\Phi_n\rangle \equiv |\Phi(t_n)\rangle$ denotes the true wave function, $|\Psi_n\rangle \equiv |\Psi(t_n)\rangle$ denotes the trial wave function, and ρ_N is the state prepared in the quantum computer according to the state $|\Psi_N\rangle$. The two states ρ_N and $|\Psi_N\rangle$ are different because of the machine noise. The triangle inequality holds for the trace distance, i.e., $D(\rho, \rho') \le D(\rho, \rho'') + D(\rho'', \rho')$. Therefore, an upper bound of $D(|\Phi_N\rangle, \rho_N)$ is given by (see Fig. 3)
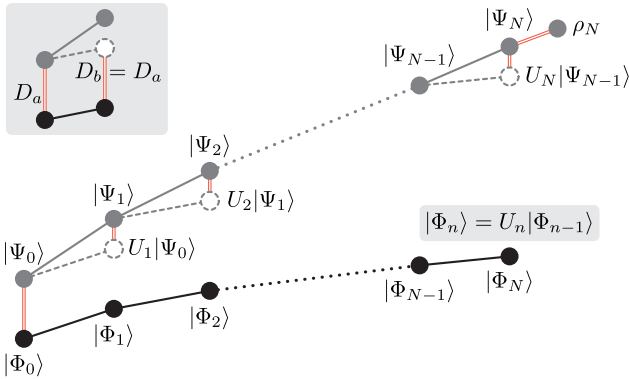

FIG. 3.   The distance between the true wave function and the wave function obtained from the hybrid algorithm. Black circles denote true wave functions given by the exact time evolution at the time $t_0, t_1, \ldots, t_N$, respectively. U_n denotes the exact time evolution during the time from t_{n-1} to t_n. Gray circles denote trial wave functions, and each empty circle with a dashed edge is the wave function at the time t_n given by the exact time evolution U_n and taking the trial wave function at the time t_{n-1} as the initial state. Distances are marked by red double lines. Note that $D_a = D(|\Phi_{n-1}\rangle, |\Psi_{n-1}\rangle)$ and $D_b = D(|\Phi_n\rangle, U_n|\Psi_{n-1}\rangle)$. The distance between $|\Phi_N\rangle$ and ρ_N is not larger than the sum of all of the red double lines in the main figure.

$$D(|\Phi_N\rangle, \rho_N) \le D(|\Phi_0\rangle, |\Psi_0\rangle) + D(|\Psi_N\rangle, \rho_N)$$
$$+ \sum_{n=1}^{N} D(U_n|\Psi_{n-1}\rangle, |\Psi_n\rangle). \qquad (14)$$

Here, U_n is the exact evolution during the time from t_{n-1} to t_n ($U_n = e^{-iH\delta t}$ if the Hamiltonian is time independent); therefore, $|\Phi_n\rangle = U_n|\Phi_{n-1}\rangle$. To obtain the upper bound, we have used $D(|\Phi_{n-1}\rangle, |\Psi_{n-1}\rangle) = D(U_n|\Phi_{n-1}\rangle, U_n|\Psi_{n-1}\rangle) = D(|\Phi_n\rangle, U_n|\Psi_{n-1}\rangle)$.

To distinguish algorithm errors and implementation errors, we use the inequality $D(U_n|\Psi_{n-1}\rangle, |\Psi_n\rangle) \le D(U_n|\Psi_{n-1}\rangle, |\Psi_n^{(0)}\rangle) + D(|\Psi_n^{(0)}\rangle, |\Psi_n\rangle)$. Here, the state $|\Psi_n^{(0)}\rangle$ is a trial state corresponding to parameters $\{\lambda_k^{(0)} = \lambda_k(t_{n-1}) + \dot{\lambda}_k^{(0)}\delta t\}$, i.e., assuming that the quantum computer reports accurate values of M and V at the time t_{n-1}. Then, the upper bound can be rewritten as

$$D(|\Phi_N\rangle, \rho_N) \le D_A + D_I, \qquad (15)$$

where

$$D_A = D(|\Phi_0\rangle, |\Psi_0\rangle) + \sum_{n=1}^{N} D(U_n|\Psi_{n-1}\rangle, |\Psi_n^{(0)}\rangle), \quad (16)$$

$$D_I = \sum_{n=1}^{N} D(|\Psi_n^{(0)}\rangle, |\Psi_n\rangle) + D(|\Psi_N\rangle, \rho_N). \qquad (17)$$

Here, D_A corresponds to algorithm errors, and D_I corresponds to implementation errors.

Algorithm errors in each time step can be expressed as

$$D(U_n|\Psi_{n-1}\rangle, |\Psi_n^{(0)}\rangle) = \sqrt{\Delta^{(2)}\delta t^2 + E}, \qquad (18)$$

where

$$\Delta^{(2)} = \langle\delta\Psi_n|\delta\Psi_n\rangle - |\langle\delta\Psi_n|\Psi_{n-1}\rangle|^2, \qquad (19)$$

$$|\delta\Psi_n\rangle = -iH|\Psi_{n-1}\rangle - \sum_k \dot{\lambda}_k^{(0)}\frac{\partial|\Psi_{n-1}\rangle}{\partial\lambda_k}, \qquad (20)$$

and

$$|E| \le \left(\sum_{m=0}^{\infty}\frac{\|H^m\|\delta t^m}{m!}\right)^2\left(\sum_{m=0}^{\infty}\frac{\|\frac{d^m R}{dt^m}\|\delta t^m}{m!}\right)^2$$
$$- \left(1 + \|H\|\delta t + \|\frac{dR}{dt}\|\delta t\right)^2$$
$$- \left(\|H^2\| + 2\|H\|\|\frac{dR}{dt}\| + \|\frac{d^2 R}{dt^2}\|\right)\delta t^2$$
$$= \Delta^{(3)}\delta t^3 + \mathcal{O}(\delta t^4). \qquad (21)$$

Here, we have used Taylor expansions of $U_n|\Psi_{n-1}\rangle$ and $|\Psi_n^{(0)}\rangle$, i.e., $U_n|\Psi_{n-1}\rangle = \sum_{m=0}^{\infty}(\delta t^m/m!)(-iH)^m|\Psi_{n-1}\rangle$ and

$|\Psi_n^{(0)}\rangle = \sum_{m=0}^{\infty}(\delta t^m/m!)(d^m R/dt^m)|\bar{0}\rangle$, where $[d/(dt)] = \sum_k \dot{\lambda}_k^{(0)}[\partial/(\partial\lambda_k)]$. See Appendix B for details. The matrix norm is induced by the vector norm; therefore, $|\langle\Psi_{n-1}|H^{m'}[d^m R/(dt^m)]|\bar{0}\rangle| \le \|H^{m'}\|\|[d^m R/(dt^m)]\|$.

Implementation errors in each time step are due to the difference between time derivatives obtained from the real quantum computer and their true values, i.e., $\delta\dot{\lambda}_k = \dot{\lambda}_k - \dot{\lambda}_k^{(0)}$, which can be expressed as

$$D(|\Psi_n^{(0)}\rangle, |\Psi_n\rangle) = \sqrt{\delta\dot{\lambda}^{\mathrm{T}} A \delta\dot{\lambda}\delta t^2} + \mathcal{O}(\delta t^3), \quad (22)$$

where A is a positive semidefinite matrix and

$$A_{k,q} = \frac{\partial\langle\Psi_{n-1}|}{\partial\lambda_q}\frac{\partial|\Psi_{n-1}\rangle}{\partial\lambda_k}$$
$$- \frac{\partial\langle\Psi_{n-1}|}{\partial\lambda_q}|\Psi_{n-1}\rangle\langle\Psi_{n-1}|\frac{\partial|\Psi_{n-1}\rangle}{\partial\lambda_k}. \quad (23)$$

Here, we have used the Taylor expansion of $|\Psi_n\rangle$, which is the same as $|\Psi_n^{(0)}\rangle$ but with $\{\dot{\lambda}_k^{(0)}\}$ replaced by $\{\dot{\lambda}_k\}$.

B. Cost of the hybrid algorithm

Using $Q_{\max}$ to denote the maximum value of the quantity Q for all t_n, we have

$$D_{\mathrm{A}} \lesssim D(|\Phi_0\rangle, |\Psi_0\rangle) + \sqrt{\Delta_{\max}^{(2)}}T + \sqrt{\Delta_{\max}^{(3)}\delta t}T, \quad (24)$$

$$D_{\mathrm{I}} \lesssim \sqrt{\|A\|_{\max}}\|\delta\dot{\lambda}\|_{\max}T + D(|\Psi_N\rangle, \rho_N). \quad (25)$$

The first term of D_{A} and the second term of D_{I} are due to imperfections in approximating the initial state using the trial wave function and preparing the final state in the quantum computer with machine noise, respectively. Note that these two terms are not T dependent; for simulations over a substantial time T, we may expect them to make relatively small contributions. In the following, we analyze the other three terms one by one.

The second term of D_{A} is the accumulation of trial wave-function imperfections in each time step. Using the circuit in Fig. 2, every term in $\Delta^{(2)}$ can be measured by a method analogous to that used for obtaining M and V. The accuracy is again limited by the shot noise and machine noise. Therefore, algorithm errors due to the trial wave function can be continually estimated during the execution of the hybrid quantum computation.

The third term of D_{A} is caused by the finite integration step size, which can be reduced by decreasing δt. In order to limit this term to ε, we need to choose a step size $\delta t \sim \varepsilon^2/(\Delta^{(3)}T^2)$, i.e., the number of time steps $N \sim \Delta^{(3)}T^3/\varepsilon^2$. When the trial wave function is a good approximation to the exact state, we can expect that $[(d^m R)/dt^m]|\Psi\rangle \simeq H^m|\Psi\rangle$, which implies $\|[(d^m R)/dt^m]\| \sim \|H^m\|$ in the subspace of the problem, and in this case, $\Delta^{(3)} \sim \|H\|^3$.

The first term of D_{I} is due to the difference between $\{\dot{\lambda}_k\}$ and their true values $\{\dot{\lambda}_k^{(0)}\}$. The difference is $\|\delta\dot{\lambda}\| \le \|1/M^{(0)}\|^2\|V^{(0)}\|\|\delta M\| + \|1/M^{(0)}\|\|\delta V\|$. Here, M and V are evaluated as required by the algorithm. Similar to $\Delta^{(2)}$, each element of A can also be measured using the circuit in Fig. 2. Therefore, the susceptibility to shot noise and machine noise in the integration process can be estimated during the execution of the hybrid quantum computation. The algorithm is susceptible to implementation errors when $M^{(0)}$ is singular, which should be avoided when choosing the trial wave function. As a worst-case scenario, implementation errors accumulate linearly with the simulated time. However, it can be far less severe: In Sec. VIII, we will explore an example in which the accumulation of errors due to the machine noise is almost negligible compared with errors in a quantum simulation based on the conventional Trotter-Suzuki decomposition.

Shot noise can be suppressed by repeating quantum circuits for measuring M and V many times. To measure the quantity $\langle X \rangle$, the deviation due to the shot noise decreases with the number of repetitions N_{r} as $\delta\langle X\rangle \propto 1/\sqrt{N_{\mathrm{r}}}$. Therefore, if there is only shot noise, we have $\|\delta\dot{\lambda}\| \sim \Delta/\sqrt{N_{\mathrm{r}}}$, where $\Delta = \|1/M^{(0)}\|^2\|V^{(0)}\|\Theta_M + \|1/M^{(0)}\|\Theta_V$, $\Theta_M = 2[\sum_{k,q}(\sum_{i,j}|if_{k,i}^*f_{q,j}|)^2]^{1/2}$, and $\Theta_V = 2[\sum_k(\sum_{i,j}|f_{k,i}^*h_j|)^2]^{1/2}$. In order to limit the overall effect of shot noise to ε', we need to choose $N_{\mathrm{r}} \sim \|A\|\Delta^2 T^2/\varepsilon'^2$.

The number of distinct circuits N_{c} required for finding the M and V parameters depends on N_{H}, the number of terms in the Hamiltonian H [see Eq. (8)]; N_{d}, the number of terms in each time derivative of R_k [see Eq. (9)]; and N_{v}, the number of parameters in the trial wave function. Note that $N_{\mathrm{c}} = N_{\mathrm{v}}^2 N_{\mathrm{d}}^2 + N_{\mathrm{v}}N_{\mathrm{d}}N_{\mathrm{H}}$, where the first term corresponds to M, and the second term corresponds to V. If each R_k is realized by N_R gates, and each term σ_i or $\sigma_{k,i}$ in the Hamiltonian or the time derivative of R_k is a Pauli operator of less than K qubits, each circuit includes at most $N_{\mathrm{g}} = N_{\mathrm{v}}N_R + 2(K+1)$ gates. Here, we have assumed that each controlled-U gate in the circuit (Fig. 2) is realized by K two-qubit controlled-σ gates, where σ is a single-qubit Pauli operator. To complete the circuit, other operations include preparing initial states of the ancillary qubit and the register and measuring the ancillary qubit.

The overall computation includes N times steps. In each time step, N_{c} circuits are implemented. Each circuit contains N_{g} gates and is repeated N_{r} times. Therefore, the overall number of gates is

$$NN_{\mathrm{c}}N_{\mathrm{g}}N_{\mathrm{r}} \sim \frac{\|A\|\Delta^2\Delta^{(3)}T^5}{\varepsilon^2\varepsilon'^2}$$
$$\times N_{\mathrm{v}}N_{\mathrm{d}}(N_{\mathrm{v}}N_{\mathrm{d}} + N_{\mathrm{H}})$$
$$\times [N_{\mathrm{v}}N_R + 2(K+1)]. \quad (26)$$

From this expression, we see that the cost is a polynomial function with respect to the integration error ε, the shot-noise error ε', and the simulated time T. Factors $\|A\|$, Δ, and $\Delta^{(3)}$ depend on the form of the trial wave function. However, during the actual execution of the hybrid computation, it will be possible to estimate $\|A\|$ and Δ. Moreover, $\Delta^{(3)} \sim \|H\|^3$, when the trial wave function is a good approximation to the true state.

It is important to remember that while the overall gate count will be a large (albeit polynomially scaling) total, the complete calculation is formed of many small quantum calculations of depth N_g. Each small computation is isolated from the others; i.e., there is no shared or persistent quantum resource, and indeed, $\sim N_c N_r$ such circuits could be performed in parallel using many separate small quantum computers.

VII. EFFECT OF MACHINE NOISE AND ERROR REDUCTION

Of the implementation errors, machine noise is the more problematic. Shot noise can be suppressed merely by repeating each quantum circuit many times, and the number of repetitions is a polynomial function with respect to the accuracy. Machine noise is less easily dismissed, and it is the focus of this section.

Machine noise need not necessarily result in computing errors. The task of the quantum computer is to evaluate coefficients M and V so that the classical computer can solve time derivatives $\{\dot{\lambda}_k\}$ according to Eq. (4). Machine noise (as well as shot noise) can potentially cause computing errors by changing these time derivatives from their true values $\{\dot{\lambda}_k^{(0)}\}$. However, sometimes machine noise does not change values of $\{\dot{\lambda}_k\}$. For example, consider the case in which the effect of machine noise is to depolarize the ancillary qubit at a fixed level; i.e., the output of the quantum computer becomes $\langle X \rangle = \eta \langle X \rangle^{(0)}$, where η is a constant independent of the quantum circuit. In this case, all equation coefficients are scaled as $M = \eta M^{(0)}$ and $V = \eta V^{(0)}$. As long as η is nonzero, the solution $\{\dot{\lambda}_k\}$ of Eq. (4) is the same for any value of η. Therefore, only an inhomogeneous scaling of quantum outputs $\langle X \rangle$ results in computing errors.

An example of the homogeneous scaling is the case of balanced measurement errors. Errors in the measurement on the ancillary qubit (see Fig. 2) can be modeled as follows: If the state of the qubit is $|0\rangle$ ($|1\rangle$), the measurement outcome is correct, i.e., 0 (1), with the probability $1 - p_0$ ($1 - p_1$), and the outcome is incorrect, i.e., 1 (0), with the probability p_0 (p_1). If there are no other implementation errors, quantum outputs are changed from $\langle X \rangle^{(0)}$ to $\langle X \rangle = (p_1 - p_0) + (1 - p_0 - p_1)\langle X \rangle^{(0)}$ (the measurement in the $\{|+\rangle, |-\rangle\}$ basis is done by performing a Hadamard gate before measuring the qubit in the $\{|0\rangle, |1\rangle\}$ basis). If measurement errors are balanced, i.e., $p_0 = p_1$, the effect of measurement errors is a fixed scaling factor $\eta = 1 - p_0 - p_1$, which does not result in computing errors. Therefore, our hybrid algorithm is inherently insensitive to measurement errors on the ancillary qubit if these errors are balanced. We remark that if single-qubit gates are reliable, one can flip the qubit before the measurement so that measurement errors are effectively balanced.

Measurement errors can be corrected even if they are not balanced. If p_0 and p_1 can be evaluated by benchmarking measurement operations, one can easily work out the true value $\langle X \rangle^{(0)}$ using the value obtained from the real machine: $\langle X \rangle^{(0)} = [\langle X \rangle - (p_1 - p_0)]/(1 - p_0 - p_1)$. We remark that when error probabilities are higher, the denominator is smaller, which means that we need to evaluate $\langle X \rangle$ with a higher accuracy in order to achieve the same accuracy of $\langle X \rangle^{(0)}$. Next, we show that a similar procedure can be applied to any machine noise if errors due to the machine noise are stochastic with tunable probabilities. We also show how to convert errors in two-qubit entangling gates, which are expected to be the main sources of errors, into stochastic errors if they are not stochastic, and how to simulate stochastic errors to tune error probabilities.

A. Error reduction

Errors in an operation are stochastic if the operation is described by a superoperator $\mathcal{N}\mathcal{U}$ and $\mathcal{N}$ has the form $\mathcal{N} = (1 - \epsilon)\mathcal{I} + \epsilon\mathcal{E}$. Here, $\mathcal{U}$ is the ideal operation without errors, $\mathcal{N}$ is the superoperator describing the effect of the noise, $\mathcal{I}$ is an identity operation, and errors $\mathcal{E}$ occur with the probability ϵ. Here, $\mathcal{E}$ is a valid quantum operation, i.e., trace-preserving completely positive map.

Given an initial state $|\bar{0}\rangle$, after a sequence of operations, the final state of the quantum computer is

$$\rho = \mathcal{N}_L \mathcal{U}_L ... \mathcal{N}_l \mathcal{U}_l ... \mathcal{N}_1 \mathcal{U}_1 (|\bar{0}\rangle\langle\bar{0}|), \qquad (27)$$

where $\mathcal{N}_l \mathcal{U}_l$ denotes the lth operation. Taking into account the fact that errors are stochastic, the quantum outcome can be rewritten in the form

$$\langle X \rangle = \left(1 - r\sum_l \epsilon_l\right)\langle X \rangle^{(0)} + r\langle X \rangle^{(1)} + \mathcal{O}(r^2). \qquad (28)$$

Here, $\langle X \rangle = \mathrm{Tr}(X\rho)$, $\langle X \rangle^{(0)} = \mathrm{Tr}(X\rho^{(0)})$ is the value without errors,

$$\rho^{(0)} = \mathcal{U}_L \cdots \mathcal{U}_l \cdots \mathcal{U}_1(|\bar{0}\rangle\langle\bar{0}|), \qquad (29)$$

$\langle X \rangle^{(1)} = \mathrm{Tr}(X\rho^{(1)})$, and

$$\rho^{(1)} = \sum_l \epsilon_l \mathcal{U}_L ... \mathcal{E}_l \mathcal{U}_l ... \mathcal{U}_1(|\bar{0}\rangle\langle\bar{0}|). \qquad (30)$$
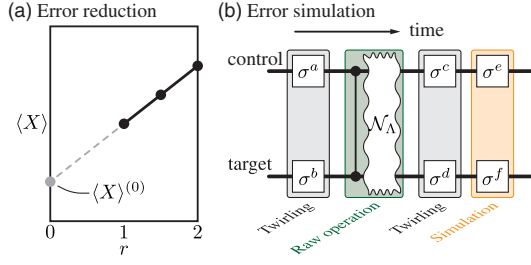
FIG. 4. (a) A schematic diagram of the error reduction. The true value $\langle X \rangle^{(0)}$ is inferred by measuring $\langle X \rangle$ for a set of error factors r and fitting data using the function $\langle X \rangle = \langle X \rangle^{(0)} + \chi r$. (b) Error twirling and simulation. Nonstochastic errors in a controlled-phase gate can be converted into stochastic errors by performing Pauli gates before and after the gate, and error probabilities can be tuned by applying Pauli gates after the gate.

The lth term of $\rho^{(1)}$ corresponds to the case in which only the lth operation causes errors and all other operations are ideal. Note that in these equations, we have replaced error probabilities ϵ_l with $r\epsilon_l$, where r is a convenient scale factor, allowing us to write $\langle X \rangle(r)$ as a function of r, and $\langle X \rangle(0) = \langle X \rangle^{(0)}$.

If probabilities of errors are tunable, we can infer the value of $\langle X \rangle(0)$ by measuring values of $\langle X \rangle(r)$ of a set of different factors r [see Fig. 4(a)]. Clearly, r can never be zero for our machine, as this would imply that we have the ability to fully switch off the machine noise, making it perfect. If $\{\epsilon_l\}$ are the minimum error probabilities allowed by the machine, the minimum value of r achievable by the machine is 1. To infer the value of $\langle X \rangle(0)$, first, we take N_X values of r as $r_1, r_2, ..., r_{N_X}$ and measure $\langle X \rangle(r_1)$, $\langle X \rangle(r_2), ..., \langle X \rangle(r_{N_X})$ using the quantum computer, where we can take $r_1 = 1$. For example, in Fig. 4(a), we have taken $r_1 = 1$, $r_2 = 1.5$, and $r_3 = 2$. Second, we fit quantum outputs using the function $\langle X \rangle(r) = \langle X \rangle^{(0)} + \chi r$, where $\chi = -\langle X \rangle^{(0)} \sum_l \epsilon_l + \langle X \rangle^{(1)}$. As a result of the fitting, we obtain the value of $\langle X \rangle^{(0)}$, represented by the gray circle in Fig. 4(a). In this way, the first-order contribution of machine noise can be corrected. Similarly, by considering second-order terms in the expansion (28), one can fit data using a function with second-order terms (i.e., r^2 terms) to correct the second-order contribution of machine noise. Using the extrapolation, we can reduce the effect of the machine noise. However, the final estimation of $\langle X \rangle^{(0)}$ may still be different from its actual value, and the error in the extrapolation depends on the shot noise in estimating each $\langle X \rangle(r)$.

The error-reduction protocol only works for small-size circuits, which are used in the hybrid algorithm, while the Trotterization algorithm usually needs large-size circuits. The true value $\langle X \rangle^{(0)}$ can be inferred because the contribution of high-order terms is much smaller than the contribution of lower-order terms, i.e., $|\mathcal{O}(r^2)| \ll |r\langle X \rangle^{(1)}|$. The total rate of errors in the quantum circuit

with N_g gates is $\sim 1 - (1 - \epsilon)^{N_g} = N_g\epsilon + N_g^2\epsilon^2/2 + \cdots$. The first term in the expansion corresponds to the first-order contribution $|r\langle X \rangle^{(1)}|$, and so on. Therefore, high-order terms cannot be neglected if $N_g\epsilon \gtrsim 1$. When there are too many gates in the circuit or the error rate is too high, the quantum state will be populated with errors, and one cannot retrieve the true value $\langle X \rangle^{(0)}$ even if we consider high-order terms in the interpolation. The best experiments to date [5,6] have reduced two-qubit gate infidelity to the range 10^{-3} to 10^{-4}. With hardware of that kind, our protocol could support hundreds of gates. Thus, a simulation of a quantum system using a trial wave function with hundred(s) of parameters may be feasible.

Implementing the error-reduction protocol requires knowledge of the inherent machine noise. Therefore, quantum operations need to be benchmarked, e.g., using quantum process tomography [55,56], before the quantum coprocessor is used, and the nature of the machine noise should not vary significantly during the simulation. Alternatively, one can monitor the machine noise in the process by stopping the protocol and benchmarking operations because, in the hybrid algorithm, the quantum computer only performs small-size circuits and can be stopped at any stage.

B. Error twirling and simulation

Nonstochastic errors can be converted into stochastic Pauli errors using some redundant Pauli gates [38–40]. It is a common feature of quantum computing systems that two-qubit gates are the main source of errors; i.e., probabilities of errors in two-qubit gates are much higher than probabilities of errors in single-qubit gates [3–6,37]. In this case, arbitrary errors in two-qubit gates can be converted into stochastic errors through the use of redundant Pauli gates, without introducing significant additional noise. We consider the controlled-phase gate $\Lambda = (\mathbb{1} + \sigma_c^z)/2 + \sigma_t^z(\mathbb{1} - \sigma_c^z)/2$ as an example, and it is similar for other two-qubit gates, e.g., the controlled-NOT gate. Here, c and t denote the control qubit and the target qubit, respectively.

The controlled-phase gate with noise is $\mathcal{N}_\Lambda \mathcal{U}_\Lambda$, where $\mathcal{U}_\Lambda \rho = \Lambda \rho \Lambda^\dagger$. In general the noise may not be in the stochastic-error form, but we can always express the noise operation in the Kraus form, i.e., $\mathcal{N}_\Lambda \rho = \sum_h E_h \rho E_h^\dagger$, where $\sum_h E_h^\dagger E_h = \mathbb{1}$, and each two-qubit matrix can be written as $E_h = \sum_{a=0}^3 \sum_{b=0}^3 \alpha_{h;a,b}\sigma_c^a\sigma_t^b$ [55]. Here, a, $b = 0, 1, 2, 3$ correspond to Pauli operators $\mathbb{1}$, σ^x, σ^y, and σ^z, respectively.

In order to convert errors, Pauli gates $\{\mathbb{1}, \sigma^x, \sigma^y, \sigma^z\}$ are randomly chosen and applied on each qubit before and after the controlled-phase gate. If the gate before the controlled-phase gate is U, the gate after the controlled-phase gate is restricted to be $\Lambda U \Lambda^\dagger$ in order to let random Pauli gates cancel each other. Here, U and $\Lambda U \Lambda^\dagger$ are both two-qubit Pauli gates. The circuit is shown in Fig. 4(b), in which we take $c = a + b(3 - b)(3 - 2a)/2$ and

$d = b + a(3 − a)(3 − 2b)/2$, so that $\sigma_c^c \sigma_t^d = \Lambda \sigma_c^a \sigma_t^b \Lambda^\dagger$ up to a phase factor. As a result, the overall operation is $\bar{\mathcal{N}}_\Lambda \mathcal{U}_\Lambda$, where $\bar{\mathcal{N}}_\Lambda$ is the superoperator describing the effective noise after applying random Pauli gates. The effective noise superoperator is in the stochastic form [57] (see Appendix C), i.e.,

$$\bar{\mathcal{N}}_\Lambda = F_\Lambda[\mathbb{1}] + \sum_{(a,b)\neq(0,0)} \epsilon_{a,b}[\sigma_c^a \sigma_t^b], \qquad (31)$$

where the fidelity is $F_\Lambda = \sum_h |\alpha_{h;0,0}|^2$, and error probabilities are $\epsilon_{a,b} = \sum_h |\alpha_{h;a,b}|^2$. Here, $[U]$ is a superoperator, and $[U]\rho = U\rho U^\dagger$. We have assumed that Pauli gates are ideal. In the case where Pauli gates are not ideal but their error probabilities are much lower than the controlled-phase gate, noise of Pauli gates will be a perturbation to the effective noise of the controlled-phase gate. Error severity in the controlled-phase gate will effectively increase by four units of the single-qubit Pauli gate error rate, and the effective noise of the controlled-phase gate may not be fully stochastic if Pauli-gate errors are nonstochastic.

To tune probabilities of errors, we can randomly perform Pauli gates after the controlled-phase gate according to desired error probabilities [see Fig. 4(b)]. Assuming that we want to tune error probabilities from $\epsilon_{e,f}$ to $r\epsilon_{e,f}$, we can perform the Pauli gate $\sigma_c^e \sigma_t^f$ [$(e,f) \neq (0,0)$] with the probability $(r − 1)\epsilon_{e,f}$. Because we are only interested in the case where $r\epsilon_{e,f} \ll 1$, overall error probabilities are approximately $\epsilon_{e,f} + (r − 1)\epsilon_{e,f}$, where the first term is due to the raw controlled-phase gate (with noise), and the second term is due to simulated errors using single-qubit Pauli gates.

VIII. NUMERICAL RESULTS—QUANTUM ISING MODEL

To demonstrate the hybrid algorithm, we numerically simulate a small quantum computer using classical computers. We take the quantum Ising model [58] as an example. The Hamiltonian of the model reads $H = H_Z + H_X$, where $H_Z = −J \sum_{j=1}^{n_s} \sigma_j^z \sigma_{j+1}^z$ and $H_X = −B \sum_{j=1}^{n_s} \sigma_j^x$. Here, n_s is the number of spins, and $\sigma_{n_s+1}^z = \sigma_1^z$. In our numerical simulations, we take $J = B = 1/2$ and $n_s = 3$; therefore, we need at least four qubits in the quantum computer to implement the hybrid algorithm. The trial state is chosen to be

$$|\Psi\rangle = e^{i\lambda_2 H_X} e^{i\lambda_1 H_Z} |\Phi(0)\rangle,$$

where the initial state $|\Phi(0)\rangle$ is a one-dimensional cluster state. In the cluster state, qubits are in the eigenstate of stabilizers $S_j = \sigma_{j-1}^z \sigma_j^x \sigma_{j+1}^z$ ($j = 1, 2, ..., n_s$) with the same eigenvalue $+1$, which is prepared by performing the controlled-phase gate on each pair of nearest-neighbor

qubits initialized in the state $|+\rangle$ [59]. The evolution of the true state is $|\Phi(t)\rangle = e^{-iHt}|\Phi(0)\rangle$. In this example, the trial state is capable of exactly matching the true state given the correct values of the parameters.

We consider a quantum computer that can initialize qubits in the state $|0\rangle$, measure a qubit in the $\{|0\rangle, |1\rangle\}$ basis, and perform single-qubit and two-qubit quantum gates. Quantum gates include the Hadamard gate, Pauli gates, phase gates $e^{i\sigma^z \theta}$, flip gates $e^{i\sigma^x \theta}$, two-qubit gates $e^{i\sigma_1^z \sigma_2^z \theta}$, the controlled-phase gate, and the controlled-NOT gate. If we have one of the three types of two-qubit gates, the other two can be efficiently realized; e.g., the gate $e^{i\sigma_1^z \sigma_2^z \theta} = H_1 \Lambda e^{i\sigma_1^x \theta} \Lambda H_1$ can be realized using two controlled-phase gates and three single-qubit gates. Here, H_1 is the Hadamard gate on qubit 1. We assume that all three types of two-qubit gates can be directly implemented, for simplification. The state $|+\rangle$ is prepared by initializing the qubit in the state $|0\rangle$ and performing a Hadamard gate; the measurement in the $\{|+\rangle, |-\rangle\}$ basis is done by performing a Hadamard gate before measuring the qubit in the $\{|0\rangle, |1\rangle\}$ basis.

We model the machine noise in the quantum computer as depolarizing errors. A qubit may be initialized in the incorrect state ($|1\rangle$) with the probability ϵ^I. The measurement outcome is incorrect with the probability $p_0 = p_1 = \epsilon^M$. For single-qubit gates, the noise superoperator is

$$\mathcal{N}^{(1)} = \left(1 - \frac{4}{3}\epsilon^{(1)}\right)[\mathbb{1}] + \frac{\epsilon^{(1)}}{3} \sum_{a=0}^{3} [\sigma^{(a)}].$$

For two-qubit gates, the noise superoperator is

$$\mathcal{N}^{(2)} = \left(1 - \frac{16}{15}\epsilon^{(2)}\right)[\mathbb{1}] + \frac{\epsilon^{(2)}}{15} \sum_{a=0}^{3} \sum_{b=0}^{3} [\sigma_1^{(a)} \sigma_2^{(b)}].$$

Here, $\epsilon^{(1)}$ and $\epsilon^{(2)}$ are rates of errors per gate. We assume that error rates of all single-qubit gates are the same, error rates of all two-qubit gates are the same, and error rates of single-qubit operations are only one-tenth of the error rates of two-qubit gates, i.e., $\epsilon^I = \epsilon^M = \epsilon^{(1)} = \epsilon^{(2)}/10$. Because the size of quantum circuits for implementing the hybrid algorithm is small, we neglect memory errors. In this model of the machine noise, errors in quantum operations are all stochastic, and error rates can be tuned by simulating errors using single-qubit Pauli gates.

Numerical simulations are performed to find the trace distance $D(|\Phi\rangle, \rho)$ between the true state $|\Phi(t)\rangle$ and the state $\rho(t)$ prepared in the quantum computer according to the trial state $|\Psi(t)\rangle$ [see Fig. 5(a)]. Because of the machine noise, $\rho(t)$ is different from $|\Psi(t)\rangle$. The average values of stabilizers can be used to describe the quality of a cluster state. The performance of quantum algorithms in estimating the average values of stabilizers is shown in Fig. 5(b). In our numerical simulations, we have taken $\epsilon^{(2)} = 0.1\%$, which is
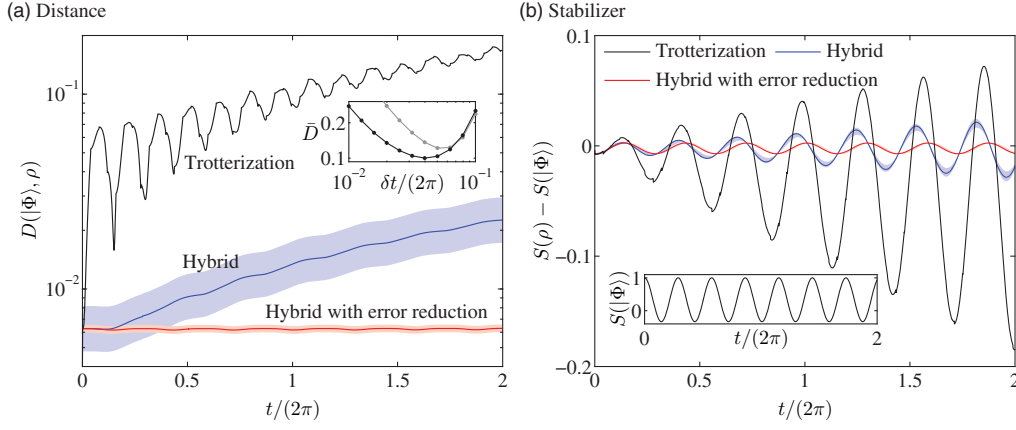
FIG. 5. Numerical estimations of the performance of quantum simulation using the Trotterization algorithm and the hybrid algorithm. The simulated system is a quantum Ising model of three spins initialized in the cluster state. (a) Trace distance $D(|\Phi\rangle, \rho)$ as a function of the simulated time t. Here, $|\Phi(t)\rangle$ is the true state, and $\rho(t)$ is the state prepared in the quantum computer. The Trotterization algorithm (black curves) and the hybrid algorithm are compared. In the Trotterization algorithm, $\rho(t)$ is prepared according to the Trotter-Suzuki decomposition, and in the hybrid algorithm, $\rho(t)$ is prepared according to the trial state $|\Psi(t)\rangle$. We have taken the error rate $\epsilon^{(2)} = 0.1\%$. The hybrid algorithm without the error reduction (blue curves) is already much more reliable than the conventional Trotterization algorithm. In the hybrid algorithm, one can further reduce the distance using the error-reduction protocol (red curve). The residual distance that cannot be eliminated using the error reduction is mainly due to errors in the state preparation (i.e., $D(|\Psi_N\rangle, \rho_N)$ term). Inset: The parameter δt in the Trotterization algorithm is optimized to minimize the average distance, and the gray curve is obtained by using the lowest-order symmetric Trotter-Suzuki decomposition (see Appendix D). (b) The difference between the average value of a stabilizer estimated using the quantum computer and its true value. Here, $S(\rho) = \mathrm{Tr}(S_2\rho)$ and $S(|\Phi\rangle) = \langle\Phi|S_2|\Phi\rangle$. Average values of other stabilizers are only slightly different. The true average value of the stabilizer is plotted in the inset. In both figures (a) and (b), light blue and red bands denote the fluctuation due to shot noise: With 68% chance (in total 100 trials), the corresponding quantity is within the band.

the state-of-the-art error rate [5,6]. See Appendix D for some details about our numerical simulations.

We observe that in this simulation, the hybrid algorithm proves to be far more reliable than the Trotterization algorithm. The distance in the hybrid algorithm is about 10 times lower than the distance in the Trotterization algorithm, and moreover, the increase of the distance as a function of time can be largely suppressed in the hybrid algorithm by using the error-reduction scheme [Fig. 5(a)]. As a result, the hybrid algorithm can provide a much better estimation of the average values of stabilizers [Fig. 5(b)]. We would like to stress that in order to make the comparison fair, the time interval selected for the Trotterization algorithm has been optimized to minimize errors (in practice, this would be possible only if the distance from the true state can be measured in the quantum computer, which would probably negate the need for simulation). We have also considered the lowest-order symmetric Trotter-Suzuki decomposition [60,61], which can reduce errors due to the Trotterization compared with the conventional Trotterization scheme. However, we find that the total errors are more significant using the symmetric decomposition given the gate error rate $\epsilon^{(2)} = 0.1\%$. In the hybrid algorithm, we have used the fourth-order Runge-Kutta method and taken $\delta t = 2\pi \times 10^{-6}$ to eliminate errors due to the numerical integration. We have neglected the effect of shot noise; therefore, all errors are due to machine noise.

Given a finite time cost of implementing the hybrid algorithm, we need to consider the effect of errors in the numerical integration and shot noise. Taking into account the effect of these two types of imperfections, we find that the distance in the hybrid algorithm may be increased but is still much lower than the distance in the Trotterization algorithm [blue and red bands in Fig. 5(a)]: We take $\delta t = 2\pi \times 10^{-4}$ and assume that each circuit is repeated $N_r = 10^4$ ($N_r = 10^6$) times to measure $\langle X \rangle$ in the hybrid algorithm (with the error reduction). Increasing the distance only slightly changes the estimation of average values of stabilizers [Fig. 5(b)].

IX. SUMMARY

We have proposed a quantum algorithm for simulating the time evolution of a quantum system. In this algorithm, both a classical processor and a quantum coprocessor are tightly integrated. Because of the assistance of the classical computer, the algorithm can be implemented with quantum circuits of much less depth (i.e., fewer quantum operations) compared with the canonical Trotterization algorithm. We discussed the robustness of the algorithm to noise, and we found that the algorithm can automatically correct some errors induced by the noise in the quantum computer. Moreover, by deliberately amplifying stochastic noise in a controllable way, the zero-error limit can be estimated;

consequently, the effect of errors can be significantly suppressed without the need for code-based quantum error correction and its concomitant resource overheads. This quantum algorithm can also be parallelized easily. The task of the quantum coprocessor is to repeatedly implement a set of small-size quantum circuits; therefore, the computing speed can be accelerated by using a cluster of quantum coprocessors, in which each works independently and there are no quantum channels linking them.

In view of these various merits, we believe that our algorithm is a promising candidate for early-stage non-fault-tolerant quantum computers.

ACKNOWLEDGMENTS

Note added.—Recently, a highly relevant paper was also posted by Ref. [63]. In that work, the authors also studied an error mitigation strategy involving deliberate variation of the error severity and subsequent extrapolation to the most-likely zero-error value of their observable. While the authors analyze this as a general technique and investigate higher-order corrections, rather than employing the technique in the specific context of quantum dynamical simulation as we do here, the results of our two papers are consistent and can be compared.

APPENDIX A: SIMULATED MODEL OF FIG. 1(b)

In Fig. 1(b), the time evolution of a qubit is simulated. The time evolution is driven by the Hamiltonian $H = -(\sigma^y + \sigma^z \cos t - \sigma^y \sin t)/2$. The trial state is in the form $|\Psi\rangle = e^{i(\pi/2)\lambda_2 \sigma^z} e^{i(\pi/2)\lambda_1 \sigma^y} |0\rangle$. The initial state is given by $\lambda_1 = 3/4$ and $\lambda_2 = -1/2$, and $T = 2\pi$. To demonstrate this example, we need a quantum computer of two qubits.

APPENDIX B: ALGORITHM ERRORS

Algorithm errors in each time step are expressed as per Eq. (18), where $E = 1 - |\langle\Psi_n^{(0)}|U_n|\Psi_{n-1}\rangle|^2 - \Delta^{(2)}\delta t^2$. Using Taylor expansions of $|\Psi_n^{(0)}\rangle$ and $U_n|\Psi_{n-1}\rangle$ and the expression of $\Delta^{(2)}$ in Eq. (19), we find that

$$|\langle\Psi_n^{(0)}|U_n|\Psi_{n-1}\rangle|^2 = 1 - \Delta^{(2)}\delta t^2 + \mathcal{O}(\delta t^3). \quad \text{(B1)}$$

Here, we used the observation that $\text{Re}\langle\bar{0}|[(dR^\dagger)/dt]|\Psi_{n-1}\rangle = 0$ and $\text{Re}\langle\bar{0}|[(d^2R^\dagger)/dt^2]|\Psi_{n-1}\rangle + \|[(dR)/dt]|\bar{0}\rangle\|^2 = 0$. We remark that $|\Psi_{n-1}\rangle = R|\bar{0}\rangle$. In other words, in the expansion of E, there is no term corresponding to either δt^0, δt^1, or δt^2. Therefore, one can obtain

the inequality (21), where the second and third lines negate the δt^0, δt^1, and δt^2 terms from the first line. We can sum the δt^3 terms from the first line of inequality (21) as follows:

$$\Delta^{(3)} = \|H\|\|H^2\| + \frac{1}{3}\|H^3\|$$
$$+ \|\frac{dR}{dt}\|\|\frac{d^2R}{dt^2}\| + \frac{1}{3}\|\frac{d^3R}{dt^3}\|$$
$$+ \|H\|\left(\|\frac{dR}{dt}\|^2 + \|\frac{d^2R}{dt^2}\|\right)$$
$$+ (\|H\|^2 + \|H^2\|)\|\frac{dR}{dt}\|. \quad \text{(B2)}$$

APPENDIX C: ERROR TWIRLING OF CONTROLLED-PHASE GATES

The twirled controlled-phase gate on a state ρ reads

$$\frac{1}{16}\sum_{a=0}^{3}\sum_{b=0}^{3}[\sigma_c^c\sigma_t^d]\mathcal{N}_\Lambda\mathcal{U}_\Lambda[\sigma_c^a\sigma_t^b]\rho = \bar{\mathcal{N}}_\Lambda\mathcal{U}_\Lambda\rho. \quad \text{(C1)}$$

The effective noise superoperator is

$$\bar{\mathcal{N}}_\Lambda = \frac{1}{16}\sum_{a=0}^{3}\sum_{b=0}^{3}[\sigma_c^a\sigma_t^b]\mathcal{N}_\Lambda[\sigma_c^a\sigma_t^b]. \quad \text{(C2)}$$

Using $\sigma^a\sigma^b\sigma^a = [2\delta_{a,b} - (2\delta_{a,0} - 1)(2\delta_{b,0} - 1)]\sigma^b$, we find that the effective noise is in the stochastic form of Eq. (31).

APPENDIX D: SOME DETAILS ABOUT NUMERICAL SIMULATIONS

In the hybrid algorithm with error reduction, we take $r_1 = 1$ and $r_2 = 2$ and fit the function $\langle X\rangle = \langle X\rangle^{(0)} + \chi r$ to infer the value of $\langle X\rangle^{(0)}$. To simulate shot noise, we use the normal distribution to approximate the binomial distribution; i.e., the value of $\langle X\rangle$ taking into account the shot noise is given by $x = 1 - 2p$, where p is given by a normal distribution with the mean $p_0 = (1 - \langle X\rangle)/2$ and the standard deviation $\sqrt{p_0(1 - p_0)/N_r}$.

In the Trotterization algorithm, the time evolution is simulated using Eq. (1). For the quantum Ising model, the Hamiltonian is decomposed into two terms: $H_1 = H_Z$ and $H_2 = H_X$. In our numerical simulations, we take $N_t = \text{ceiling}(T/\delta t)$, $\tau_{n,j} = \delta t$ if $n < N_t$, and $\tau_{N_t,j} = T - (N_t - 1)\delta t$. Such a method of determining Trotterization parameters coincides with the optimal choice of the number N_t to minimize errors [36]. The state $\rho(t)$ is given by replacing T with t and determining N_t and $\tau_{n,j}$ using the method we just described.

To find the optimal δt, we consider the average trace distance $D(|\bar{\Phi}\rangle, \rho) = T^{-1}\int_0^T dt D(|\Phi(t)\rangle, \rho(t))$, where $|\Phi\rangle$

is the true state and ρ is the state prepared in the quantum computer according to the Trotterization algorithm. The average trace distance is plotted in Fig. 5(a), taking $T = 4\pi$. To obtain other results in Fig. 5, we take $\delta t = 2\pi \times 10^{-1.4}$ in the Trotterization algorithm, which minimizes the average distance.

It is interesting to ask whether second-order techniques, which are known to be helpful in reducing the error in the Trotterization technique, might have a superior performance here. For the lowest-order symmetric Trotter-Suzuki decomposition [60,61], each time step in Eq. (1) is replaced by $\prod_{j=1}^{N_H} e^{-iH_j\delta t/2} \prod_{j=N_H}^{1} e^{-iH_j\delta t/2}$, where N_H is the number of terms in the Hamiltonian. The performance when using this technique is plotted as the grey curve in the inset to Fig. 5(a). We see that the performance is actually inferior to the more basic Trotterization approach; the explanation is that the potential gains are more than negated because the larger number of gates employed introduces a greater degree of error (due to the 0.1% physical gate errors).

---

[1] A. Kitaev, *Fault-Tolerant Quantum Computation by Anyons*, Ann. Phys. (Amsterdam) **303**, 2 (2003).

[2] R. Raussendorf and J. Harrington, *Fault-Tolerant Quantum Computation with High Threshold in Two Dimensions*, Phys. Rev. Lett. **98**, 190504 (2007).

[3] R. Barends, J. Kelly, A. Megrant, A. Veitia, D. Sank, E. Jeffrey, T. C. White, J. Mutus, A. G. Fowler, B. Campbell *et al.*, *Superconducting Quantum Circuits at the Surface Code Threshold for Fault Tolerance*, Nature (London) **508**, 500 (2014).

[4] T. P. Harty, D. T. C. Allcock, C. J. Ballance, L. Guidoni, H. A. Janacek, N. M. Linke, D. N. Stacey, and D. M. Lucas, *High-Fidelity Preparation, Gates, Memory and Readout of a Trapped-Ion Quantum Bit*, Phys. Rev. Lett. **113**, 220501 (2014).

[5] C. J. Ballance, T. P. Harty, N. M. Linke, M. A. Sepiol, and D. M. Lucas, *High-Fidelity Quantum Logic Gates Using Trapped-Ion Hyperfine Qubits*, Phys. Rev. Lett. **117**, 060504 (2016).

[6] J. P. Gaebler, T. R. Tan, Y. Lin, Y. Wan, R. Bowler, A. C. Keith, S. Glancy, K. Coakley, E. Knill, D. Leibfried, and D. J. Wineland, *High-Fidelity Universal Gate Set for $^9Be^+$ Ion Qubits*, Phys. Rev. Lett. **117**, 060505 (2016).

[7] J. O'Gorman and E. T. Campbell, *Quantum Computation with Realistic Magic State Factories*, Phys. Rev. A **95**, 032338 (2017)

[8] M. Smelyanskiy, N. P. D. Sawaya, and A. Aspuru-Guzik, *qHiPSTER: The Quantum High Performance Software Testing Environment*, arXiv:1601.07195.

[9] E. Farhi and J. Goldstone, *A Quantum Approximate Optimization Algorithm*, arXiv:1411.4028.

[10] 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).

[11] D. Wecker, M. B. Hastings, and M. Troyer, *Progress Towards Practical Quantum Variational Algorithms*, Phys. Rev. A **92**, 042303 (2015).

[12] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, *The Theory of Variational Hybrid Quantum-Classical Algorithms*, New J. Phys. **18**, 023023 (2016).

[13] B. Bauer, D. Wecker, A. J. Millis, M. B. Hastings, and M. Troyer, *Hybrid Quantum-Classical Approach to Correlated Materials*, Phys. Rev. X **6**, 031045 (2016).

[14] J. M. Kreula, S. R. Clark, and D. Jaksch, *Non-linear Quantum-Classical Scheme to Simulate Non-equilibrium Strongly Correlated Fermionic Many-Body Dynamics*, Sci. Rep. **6**, 32940 (2016).

[15] J. M. Kreula, L. García-Álvarez, L. Lamata, S. R. Clark, E. Solano, and D. Jaksch, *Few-Qubit Quantum-Classical Simulation of Strongly Correlated Lattice Fermions*, EPJ Quantum Techno. **3**, 11 (2016).

[16] Y. Shen, X. Zhang, S. Zhang, J.-N. Zhang, M.-H. Yung, and K. Kim, *Quantum Implementation of Unitary Coupled Cluster for Simulating Molecular Electronic Structure*, Phys. Rev. A **95**, 020501 (2017).

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

[18] R. Feynman, *Simulating Physics with Computers*, Int. J. Theor. Phys. **21**, 467 (1982).

[19] S. Lloyd, *Universal Quantum Simulators*, Science **273**, 1073 (1996).

[20] D. Wecker, B. Bauer, B. K. Clark, M. B. Hastings, and M. Troyer, *Gate-Count Estimates for Performing Quantum Chemistry on Small Quantum Computers*, Phys. Rev. A **90**, 022305 (2014).

[21] D. Poulin, M. B. Hastings, D. Wecker, N. Wiebe, Andrew C. Doberty, and M. Troyer, *The Trotter Step Size Required for Accurate Quantum Simulation of Quantum Chemistry*, Quantum. Inf. Comput. **15**, 361 (2015).

[22] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, *Elucidating Reaction Mechanisms on Quantum Computers*, arXiv:1605.03590.

[23] I. Kassal, J. D. Whitfield, A. Perdomo-Ortiz, M.-H. Yung, and A. Aspuru-Guzik, *Simulating Chemistry Using Quantum Computers*, Annu. Rev. Phys. Chem. **62**, 185 (2011).

[24] B. P. Lanyon, C. Hempel, D. Nigg, M. Müller, R. Gerritsma, F. Zähringer, P. Schindler, J. T. Barreiro, M. Rambach, G. Kirchmair, M. Hennrich, P. Zoller, R. Blatt, and C. F. Roos, *Universal Digital Quantum Simulation with Trapped Ions*, Science **334**, 57 (2011).

[25] X. Ma, B. Dakic, W. Naylor, A. Zeilinger, and P. Walther, *Quantum Simulation of the Wavefunction to Probe Frustrated Heisenberg Spin Systems*, Nat. Phys. **7**, 399 (2011).

[26] R. Barends, L. Lamata, J. Kelly, L. García-Álvarez, A. G. Fowler, A. Megrant, E. Jeffrey, T. C. White, D. Sank, J. Y. Mutus *et al.*, *Digital Quantum Simulation of Fermionic Models with a Superconducting Circuit*, Nat. Commun. **6**, 7654 (2015).

[27] H. Aoki, N. Tsuji, M. Eckstein, M. Kollar, T. Oka, and P. Werner, *Nonequilibrium Dynamical Mean-Field Theory and Its Applications*, Rev. Mod. Phys. **86**, 779 (2014).

[28] G. Carleo, F. Becca, M. Schiró, and M. Fabrizio, *Localization and Glassy Dynamics of Many-Body Quantum Systems*, Sci. Rep. **2**, 243 (2012).

[29] A. J. Daley, C. Kollath, U. Schollwöck, and G. Vidal, *Time-Dependent Density-Matrix Renormalization-Group Using Adaptive Effective Hilbert Spaces*, J. Stat. Mech. (2004) P04005.

[30] M. C. Bañuls, M. B. Hastings, F. Verstraete, and J. I. Cirac, *Matrix Product States for Dynamical Simulation of Infinite Chains*, Phys. Rev. Lett. **102**, 240603 (2009).

[31] E. Runge and E. K. U. Gross, *Density-Functional Theory for Time-Dependent Systems*, Phys. Rev. Lett. **52**, 997 (1984).

[32] J. I. Cirac and P. Zoller, *Goals and Opportunities in Quantum Simulation*, Nat. Phys. **8**, 264 (2012).

[33] H. F. Trotter, *On the Product of Semi-groups of Operators*, Proc. Am. Math. Soc. **10**, 545 (1959).

[34] G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, *Quantum Algorithms for Fermionic Simulations*, Phys. Rev. A **64**, 022319 (2001).

[35] I. Dhand and B. C Sanders, *Stability of the Trotter-Suzuki Decomposition*, J. Phys. A **47**, 265206 (2014).

[36] G. C. Knee and W. J. Munro, *Optimal Trotterization in Universal Quantum Simulators under Faulty Control*, Phys. Rev. A **91**, 052327 (2015).

[37] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, and J. L. O'Brien, *Quantum Computers*, Nature (London) **464**, 45 (2010).

[38] E. Knill, *Fault-Tolerant Postselected Quantum Computation: Threshold Analysis*, arXiv:quant-ph/0404104.

[39] J. J. Wallman and J. Emerson, *Noise Tailoring for Scalable Quantum Computation via Randomized Compiling*, Phys. Rev. A **94**, 052325 (2016).

[40] J. O'Gorman, N. H. Nickerson, P. Ross, J. J. L. Morton, and S. C. Benjamin, *A Silicon-Based Surface Code Quantum Computer*, npj Quantum Inf. **2**, 15019 (2016).

[41] R. O. Jones, *Density Functional Theory: Its Origins, Rise to Prominence, and Future*, Rev. Mod. Phys. **87**, 897 (2015).

[42] D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, *Matrix Product State Representations*, Quantum Inf. Comput. **7**, 401 (2007).

[43] H. Feldmeier and J. Schnack, *Molecular Dynamics for Fermions*, Rev. Mod. Phys. **72**, 655 (2000).

[44] A. G. Taube and R. J. Bartlett, *New Perspectives on Unitary Coupled-Cluster Theory*, Int. J. Quantum Chem. **106**, 3393 (2006).

[45] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, *Quantum Computation by Adiabatic Evolution*, arXiv:quant-ph/0001106.

[46] P. A. M. Dirac, *The Principles of Quantum Mechanics* (Clarendon, Oxford, 1930).

[47] J. Frenkel, *Wave Mechanics: Advanced General Theory* (Clarendon, Oxford, 1934).

[48] J. Broeckhove, L. Lathouwers, E. Kesteloot, and P. Van Leuven, *On the Equivalence of Time-Dependent Variational Principles*, Chem. Phys. Lett. **149**, 547 (1988).

[49] A. D. McLachlan, *A Variational Solution of the Time-Dependent Schrodinger Equation*, Mol. Phys. **8**, 39 (1964).

[50] For the (i, j) term of $M_{k,q}$, $a = 2|if^*_{k,i}f_{q,j}|$, $\theta = \arg if^*_{k,i}f_{q,j}$, and $U = R^\dagger_{k,i}R_{q,j}$; for the (i, j) term of V_k, $a = 2|f^*_{k,i}h_j|$, $\theta = \arg f^*_{k,i}h_j$ and $U = R^\dagger_{k,i}\sigma_j R$.

[51] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, in *Quantum Amplitude Amplification and Estimation*, edited by S. J. Lomonaco Jr. (AMS Contemporary Mathematics, Providence, 2002), Vol. 305, p. 53; arXiv:quant-ph/0005055.

[52] A. K. Ekert, C. M. Alves, D. K. L. Oi, M. Horodecki, and L. C. Kwek, *Direct Estimations of Linear and Nonlinear Functionals of a Quantum State*, Phys. Rev. Lett. **88**, 217901 (2002).

[53] In the circuit, there are only two controlled gates because of the pattern of R_l terms in $\langle \bar{0}|U|\bar{0}\rangle$. Note that U is a product of unitary operators, and if R_l is in the product, $R^\dagger_l$ is also in the product. The only two exceptions are U_k and U_q. The circuit prepares the state $(|0\rangle \otimes R_{N_v}\ldots R_k U_k\ldots R_1|\bar{0}\rangle + e^{i\theta}|1\rangle \otimes R_{N_v}\ldots R_q U_q\ldots R_1|\bar{0}\rangle)/\sqrt{2}$ before the ancillary qubit is measured. If we take out an overall factor $R_{N_v}\ldots R_k U_k\ldots R_1$, which does not affect the measurement outcome, the state becomes $(|0\rangle\otimes|\bar{0}\rangle + e^{i\theta}|1\rangle\otimes U|\bar{0}\rangle)/\sqrt{2}$, which is the state prepared using the circuit in Ref. [52].

[54] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations* (Wiley, New York, 2008).

[55] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, England, 2010).

[56] R. C. Bialczak, M. Ansmann, M. Hofheinz, E. Lucero, M. Neeley, A. D. O'Connell, D. Sank, H. Wang, J. Wenner, M. Steffen, A. N. Cleland, and J. M. Martinis, *Quantum Process Tomography of a Universal Entangling Gate Implemented with Josephson Phase Qubits*, Nat. Phys. **6**, 409 (2010).

[57] J. Emerson, M. Silva, O. Moussa, C. Ryan, M. Laforest, J. Baugh, D. G. Cory, and R. Laflamme, *Symmetrized Characterization of Noisy Quantum Processes*, Science **317**, 1893 (2007).

[58] S. Sachdev, *Quantum Phase Transitions* (Cambridge University Press, Cambridge, England, 2011).

[59] R. Raussendorf, D. E. Browne, and H. J. Briegel, *Measurement-Based Quantum Computation on Cluster States*, Phys. Rev. A **68**, 022312 (2003).

[60] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, *Efficient Quantum Algorithms for Simulating Sparse Hamiltonians*, Commun. Math. Phys. **270**, 359 (2007).

[61] N. Wiebe, D. Berry, P. Høyer, and B. C. Sanders, *Higher Order Decompositions of Ordered Operator Exponentials*, J. Phys. A **43**, 065203 (2010).

[62] http://dx.doi.org/10.5281/zenodo.22558.

[63] K. Temme, S. Bravyi, and J. M. Gambetta, *Error Mitigation for Short Depth Quantum Circuits*, arXiv:1612.02058.