

Tensor-based phase difference estimation on time-series analysis

Shu Kanno ^{1,2,*}, Kenji Sugisaki ^{3,2,4}, Rei Sakuma^{5,2}, Jumpei Kato ^{6,2,7}, Hajime Nakamura ², and Naoki Yamamoto ^{2,8}

¹*Mitsubishi Chemical Corporation, Science & Innovation Center, Yokohama 227-8502, Japan*

²*Quantum Computing Center, Keio University, 3-14-1 Hiyoshi, Kohoku-ku, Yokohama 223-8522, Japan*

³*Deloitte Tohmatsu LLC, 3-2-3 Marunouchi, Chiyoda-ku, Tokyo 100-8363, Japan*

⁴*Centre for Quantum Engineering, Research and Education, TCG Centres for Research and Education in Science and Technology, Sector V, Salt Lake, Kolkata 700091, India*

⁵*Materials Informatics Initiative, RD Technology & Digital Transformation Center, JSR Corporation, 3-103-9 Tonomachi, Kawasaki-ku, Kawasaki 210-0821, Japan*

⁶*Mitsubishi UFJ Financial Group Inc. and MUFG Bank Ltd., 4-10-2 Nakano, Nakano-ku, Tokyo 164-0001, Japan*

⁷*Graduate School of Science and Technology, Keio University, 3-14-1 Hiyoshi, Kohoku, Yokohama, Kanagawa 223-8522, Japan*

⁸*Department of Applied Physics and Physico-Informatics, Keio University, 3-14-1 Hiyoshi, Kohoku-ku, Yokohama 223-8522, Japan*



(Received 3 March 2026; accepted 20 April 2026; published 15 May 2026)

We propose a phase-difference-estimation algorithm based on tensor-network circuit compression, leveraging time-evolution data to pursue scalability and higher accuracy with a quantum-phase-estimation- (QPE) type algorithm. Using tensor networks, we construct circuits composed solely of nearest-neighbor gates and extract time-evolution data with four-type circuit measurements. In addition, to enhance the accuracy of the time-evolution and state-preparation circuits, we propose techniques based on algorithmic error mitigation and on iterative circuit optimization combined with merging into matrix product states, respectively. Verification using a noiseless simulator for the eight-qubit one-dimensional Hubbard model using an ancilla qubit shows that the proposed algorithm achieves accuracies with 0.4%–4.7% error from a true energy gap with an appropriate time-step size and that accuracy improvements due to the algorithmic error mitigation are observed. We also confirm the enhancement of the overlap with matrix product states through iterative optimization. Finally, the proposed algorithm is demonstrated on IBM Heron devices with Q-CTRL error suppression for 8-, 36-, and 52-qubit models using more than 4000 2-qubit gates. These largest-scale demonstrations for the QPE-type algorithm represent significant progress not only toward practical applications of near-term quantum computing but also toward preparation for the era of error-corrected quantum devices.

DOI: [10.1103/PhysRevA.113.052433](https://doi.org/10.1103/PhysRevA.113.052433)

I. INTRODUCTION

Quantum computers can perform chemical tasks that are intractable for classical algorithms. One of the tasks is calculating spectral quantities, such as energies and energy gaps, in chemical and physical models. The algorithms for the tasks include quantum-phase-estimation- (QPE) type algorithms, which estimate the value Θ from the phase of $\exp(i\Theta)$ constructed in a quantum circuit [1–3]: If Θ corresponds to an eigenvalue for a given Hamiltonian, the algorithm estimates energy. Since QPE-type algorithms can estimate the quantities with desired precision with a polynomial cost for the system size, a clear exponential advantage is expected from a quantum computer in theory; in the classical counterpart, the full configuration interaction (FCI), the computational cost increases exponentially with the system size, and benchmarks have been performed only up to a maximum of 46 qubits [4].

In the history of progress on QPE-type algorithms [5–7], approaches based on time-series-data analysis [8–10] attracted attention in the near-term or early fault-tolerant quantum computer (FTQC) era. Specifically, in this approach, the phase Θ is estimated from time-series data $\{\exp(i\Theta t)\}$ for multiple times t . The approach leverages mature classical processing techniques such as the Fourier transformation, the convolution integral [11,12], and direct data fitting [13–15], leading to shallow circuits. However, on current noisy devices, experimental implementations have been limited to only few-qubit systems [16] since the circuit to run the approach is still too deep for current noisy devices. For end-to-end implementation on the devices, efficient implementation of initial-state preparation, time-evolution operators, and time-series-data extraction is required.

We propose an efficient QPE-type algorithm using time-series analysis, as shown in Fig. 1. Here, we adopt a strategy combining tensor-network circuit compression [17–32] and quantum phase difference estimation (QPDE) [33], which enabled our group to achieve the largest demonstration for the QPE-type algorithm up to a 32-qubit system [34]: QPDE is one QPE-type algorithm focusing on energy-gap calculation (i.e., Θ corresponds to the gap Δ) without controlled operations for time-evolution operators. The circuits for initial-state

*Contact author: shu.kanno@quantum.keio.ac.jp

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

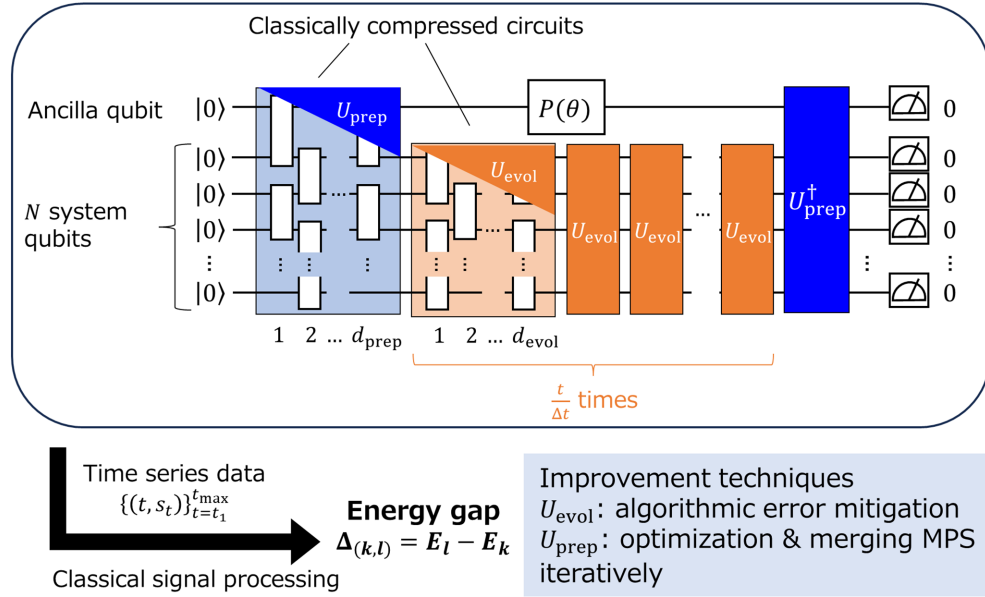


FIG. 1. Overview of our proposal. $P(\theta) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$ represents the phase gate, with the parameter θ taking values of $0, \pi/2, \pi$, and $3\pi/2$. s_t is a signal to construct time-series data $\{(t, s_t)\}_{t=t_1}^{t_{\max}}$, where t_1 and $t_{\max}$ are Δt and a maximum time, respectively.

preparation and time evolution in QPDE are further compressed to sets of brick-wall gates through the matrix product states (MPSs) and matrix product operators (MPOs), represented as U_{prep} and U_{evol} , respectively. We choose the direct data fitting as the classical signal processing [13].

In addition, to improve the accuracy of tensor-based QPDE, we introduce two refinement techniques. The first is the improvement of time-evolution error by using algorithmic error mitigation (AEM) [35]. While AEM is mainly targeted to mitigate Trotter errors, our approach, which uses techniques from Ref. [36], provides a framework for mitigating approximation errors introduced by tensor compression of a time-evolution circuit. The second is the improvement of the state-preparation overlap. The state-preparation circuit is optimized to approximate an MPS, but the classical optimization cost grows exponentially with the circuit depth. In this work, we propose an approach that allows the circuit depth to be increased while suppressing the exponential growth in classical cost.

Finally, we executed time-series QPDE on real quantum hardware for a one-dimensional Hubbard model with up to 52 qubits, exceeding the size limit of the FCI. Although the accuracy of this approach is confined depending on the tensor-network topology, e.g., quasi-one dimension in the MPSs and MPOs, demonstrations of the large-scale QPE-type algorithm on current devices create a clear pathway not only for the discovery of near-term quantum-advantage tasks but also for practical insights into the FTQC era. This point is also clearly distinct from recent large-scale demonstrations of energy estimation based on quantum-classical hybrid approaches [37,38]. Note that in our previous demonstration [34], we adopted an adaptive approach based on Bayesian inference. In addition to the advantages described above, the present nonadaptive time-series approach has the benefit that data can be acquired at all time steps until the signal decays and that the number of shots required per time step is small.

II. METHOD

As shown in Fig. 1, we extract a signal s_t with energy-gap information for a time t from the compressed circuit and then calculate the energy gap from the data using classical signal processing, where the time-series data are denoted as $\{(t, s_t)\}_{t=t_1}^{t_{\max}}$ and t_1 and $t_{\max}$ are Δt and a maximum time, respectively. The circuit applies the state-preparation operator $U_{\text{prep}}|0\rangle^{\otimes N+1}$, a phase gate $P(\theta)$ with phase θ , the time-evolution operator U_{evol} , and the inverse operator $U_{\text{prep}}^\dagger$ and, finally, measures by all zeros. U_{prep} and U_{evol} are composed of brick-wall gates with depths d_{prep} and d_{evol} , respectively.

We first explain the extraction of s_t in the next section. Then we provide a summary of a procedure for compression of U_{prep} and U_{evol} proposed by a previous study [34]. We also describe the techniques to improve the quality of U_{prep} and U_{evol} .

A. Time-series-data calculation

We generally describe the prepared state as

$$U_{\text{prep}}|0\rangle^{\otimes N+1} = a_0|0\rangle|\phi\rangle + a_1|1\rangle|\phi'\rangle, \quad (1)$$

where $a_0, a_1 \in \mathbb{C}$, $|a_0|^2 + |a_1|^2 = 1$, and $|\phi\rangle$ and $|\phi'\rangle$ are the states of a system. The expectation values of the circuits as a function of θ are

$$\begin{aligned} m(\theta) &= \text{Tr}\{[P(\theta)^\dagger U_{\text{prep}}|0\rangle\langle 0|^{\otimes N+1} U_{\text{prep}}^\dagger P(\theta)]\rho(t)\} \\ &= |a_0|^4 |\langle \phi | (U_{\text{evol}})^{\frac{t}{\Delta t}} | \phi \rangle|^2 \\ &\quad + e^{i\theta} |a_0|^2 |a_1|^2 \langle \phi' | (U_{\text{evol}})^{\frac{t}{\Delta t}} | \phi' \rangle \langle \phi | (U_{\text{evol}}^\dagger)^{\frac{t}{\Delta t}} | \phi \rangle \\ &\quad + e^{-i\theta} |a_0|^2 |a_1|^2 \langle \phi | (U_{\text{evol}})^{\frac{t}{\Delta t}} | \phi \rangle \langle \phi' | (U_{\text{evol}}^\dagger)^{\frac{t}{\Delta t}} | \phi' \rangle \\ &\quad + |a_1|^4 |\langle \phi' | (U_{\text{evol}})^{\frac{t}{\Delta t}} | \phi' \rangle|^2, \end{aligned} \quad (2)$$

where $\rho(t) = |\psi(t)\rangle\langle\psi(t)|$ and $|\psi(t)\rangle = (U_{\text{evol}})^{t/\Delta t} U_{\text{prep}} |0\rangle^{\otimes N+1}$.

Combining $m(\theta)$ for $\theta = 0, \pi/2, \pi$, and $3\pi/2$ yields time signal data s_t as

$$\begin{aligned} s_t &= \frac{1}{4|a_0|^2(1 - |a_0|^2)} \\ &\times \left\{ m(0) - m(\pi) - i \left[m\left(\frac{\pi}{2}\right) - m\left(\frac{3\pi}{2}\right) \right] \right\} \\ &= \langle \phi | (U_{\text{evol}}^\dagger)^{\frac{t}{\Delta t}} | \phi \rangle \langle \phi' | (U_{\text{evol}})^{\frac{t}{\Delta t}} | \phi' \rangle, \end{aligned} \quad (3)$$

where the first (last) two terms in the second line correspond to the real (imaginary) part of s_t . $|a_0|^2$ can be classically determined since we can use $U_{\text{prep}}|0\rangle^{\otimes N+1}$ as an MPS and $|a_0|^2 = |(\langle 0| \otimes I^{\otimes N}) U_{\text{prep}} |0\rangle^{\otimes N+1}|^2$.

Let $|\psi_k\rangle$ be an eigenstate such that $H|\psi_k\rangle = E_k|\psi_k\rangle$ with eigenvalues $E_0 < E_1 < E_2 < \dots$, where $|\psi_0\rangle = |\psi_g\rangle$ and $|\psi_1\rangle = |\psi_{\text{ex}}\rangle$. Expanding the states as

$$\begin{aligned} |\phi\rangle &= \sum_k \sqrt{p_k} e^{i\epsilon_k} |\psi_k\rangle, \\ |\phi'\rangle &= \sum_l \sqrt{q_l} e^{i\phi_l} |\psi_l\rangle, \end{aligned} \quad (4)$$

we obtain the spectral form

$$\begin{aligned} s_t &= \langle \phi | (U_{\text{evol}}^\dagger)^{\frac{t}{\Delta t}} | \phi \rangle \langle \phi' | (U_{\text{evol}})^{\frac{t}{\Delta t}} | \phi' \rangle \\ &= \sum_{k,l} p_k q_l e^{-it(E_l - E_k)} = \sum_J P_J e^{-i\Delta_J t}, \end{aligned} \quad (5)$$

where the indices k and l are merged as $J = (k, l)$, $\Delta_J = E_l - E_k$, and $P_J = p_k q_l \geq 0$ denotes the corresponding spectral weight.

When $|\phi\rangle \approx |\psi_g\rangle$ (ground state) and $|\phi'\rangle \approx |\psi_{\text{ex}}\rangle$ (excited state), the dominant frequency component with the largest P_J , Δ_J , corresponds to the target energy gap $\Delta_{J=(0,1)} = E_1 - E_0$, where we choose the first excited state to be the target excited state. Note that we can extract the information for the other gaps by adjusting $|\psi'_g\rangle$ and $|\psi'_{\text{ex}}\rangle$ in the MPS preparation in Eq. (14).

We explain the effect of depolarizing noise. The state before the measurement is

$$\tau(t) = U_{\text{prep}}^\dagger P(\theta) \rho(t) P(\theta)^\dagger U_{\text{prep}}. \quad (6)$$

Due to a depolarizing noise channel, the state transforms as

$$\tau'(t) = [1 - p_{\text{dep}}(t)] \tau(t) + \frac{p_{\text{dep}}}{2^{N+1}} I^{\otimes N+1}, \quad (7)$$

where p_{dep} is an error rate.

The measured probability $m'(\theta)$ is then

$$\begin{aligned} m'(\theta) &= \text{Tr}[|0\rangle\langle 0| \tau'(t)] \\ &= [1 - p_{\text{dep}}(t)] \text{Tr}[|0\rangle\langle 0| \tau(t)] \\ &\quad + \frac{p_{\text{dep}}}{2^{N+1}} \text{Tr}[|0\rangle\langle 0| I^{\otimes N+1}] \\ &= [1 - p_{\text{dep}}(t)] m(\theta) + \frac{p_{\text{dep}}(t)}{2^{N+1}}. \end{aligned} \quad (8)$$

By combining $m(\theta)$ for $\theta = 0, \pi/2, \pi, 3\pi/2$,

$$s_t = [1 - p_{\text{dep}}(t)] \langle \phi | (U_{\text{evol}}^\dagger)^{\frac{t}{\Delta t}} | \phi \rangle \langle \phi' | (U_{\text{evol}})^{\frac{t}{\Delta t}} | \phi' \rangle, \quad (9)$$

which is the same as Eq. (3), except for the amplitude decay $[1 - p_{\text{dep}}(t)]$.

The gap Δ_J can be extracted from the time-series data s_t by using classical processing. We adopt the data fitting [13] with the matrix pencil method [10,39,40] as an initial guess for the classical signal processing. See Appendix A for details.

B. Summary of classical unitary compression using tensor networks

The time-evolution operator U_{evol} and the state-preparation circuit U_{prep} are optimized in a brick-wall ansatz so that they satisfy

$$U_{\text{evol}} \approx e^{-iH\Delta t} \quad (10)$$

and

$$U_{\text{prep}} |0\rangle^{\otimes N+1} \approx \frac{1}{\sqrt{2}} (|0\rangle |\psi_g\rangle + |1\rangle |\psi_{\text{ex}}\rangle), \quad (11)$$

where Δt is the time step, N is the number of system qubits, and $|\psi_g\rangle$ and $|\psi_{\text{ex}}\rangle$ denote the ground and excited states, respectively.

We first describe the treatment of time evolution U_{evol} in Figs. 2(a)–2(c). As a reference, we prepare an accurate MPO representation of the time-evolution operator, denoted $U_{\text{ref}}^\dagger$ in Fig. 2(a). Assuming $H = \sum_{\beta=1}^K C_\beta P_\beta$, where C_β is a coefficient and P_β is a Pauli term, we construct $U_{\text{ref}}^\dagger$ as

$$\begin{aligned} U_{\text{ref}}^\dagger &= S_2 \left(\frac{-\Delta t}{m} \right)^m \\ &= \left\{ \prod_{\beta=1}^K \exp \left(i C_\beta P_\beta \frac{\Delta t}{2m} \right) \prod_{\beta=K}^1 \exp \left(i C_\beta P_\beta \frac{\Delta t}{2m} \right) \right\}^m, \end{aligned} \quad (12)$$

where $S_2(\cdot)$ is the second-order Trotter formula and m is the number of time slices.

We then optimize each of the two-qubit gates in the brick-wall circuit U_{evol} to minimize

$$\begin{aligned} \|U_{\text{ref}}^\dagger - U_{\text{evol}}\|_F^2 &= \text{Tr}[U_{\text{ref}}^\dagger U_{\text{ref}}] + \text{Tr}[U_{\text{evol}}^\dagger U_{\text{evol}}] \\ &\quad - 2\text{ReTr}[U_{\text{ref}}^\dagger U_{\text{evol}}] \\ &= 2^{N+1} - 2\text{ReTr}[U_{\text{ref}}^\dagger U_{\text{evol}}], \end{aligned} \quad (13)$$

where $\|\cdot\|_F$ is the Frobenius norm. Specifically, we first contract all of the tensors except for the target gate and obtain a four-leg tensor $G'^\dagger$ in Fig. 2(b). After the singular-value decomposition (SVD) of G' as $G' = UV^\dagger$, we obtain the local optimal gate $G_{\text{opt}} = UV^\dagger$ by substituting S in the identity, where U and $V^\dagger$ are unitary operators. This optimization is performed for each gate, where the gate optimization is performed in the order of sweeping up and down the zigzag path from left to right in Fig. 2(c). The initial gate was constructed by perturbing the identity operator.

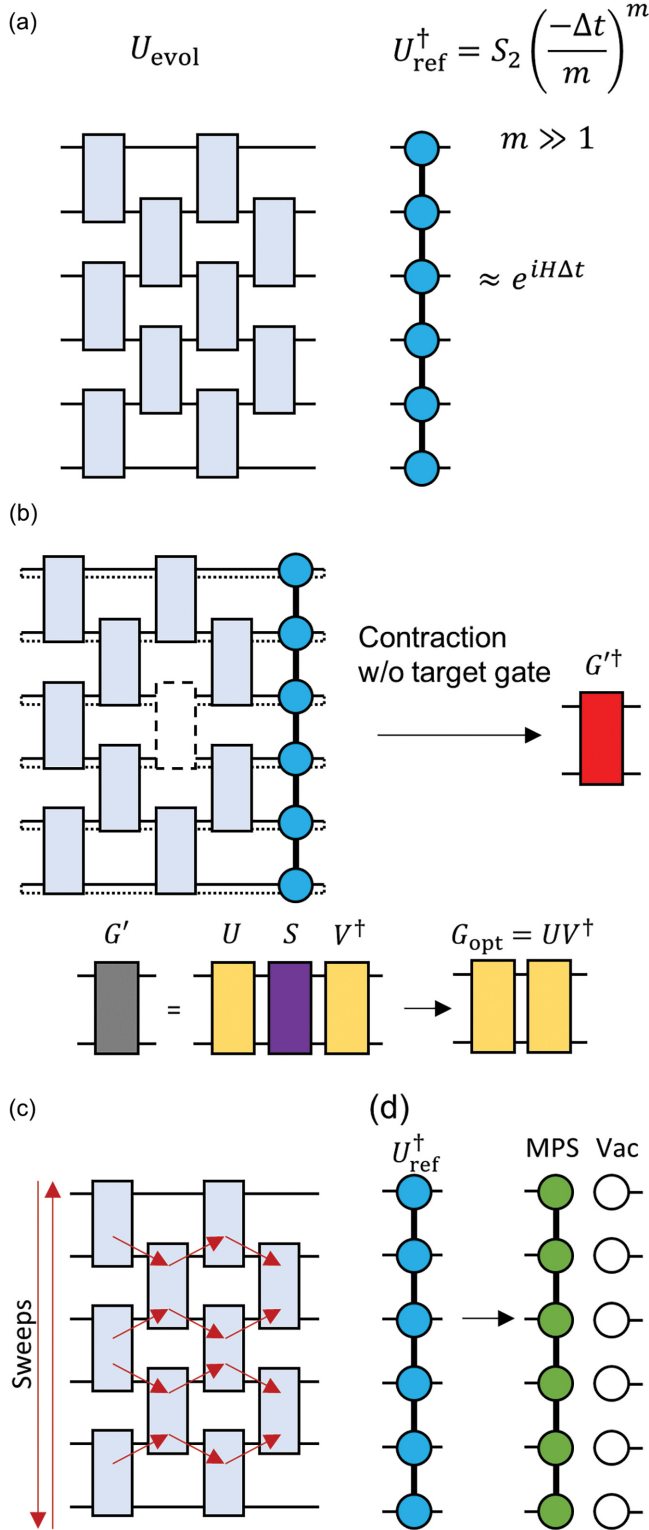


FIG. 2. Summary of the unitary compression in our previous work. (a) Initial preparations for U_{evol} and $U_{\text{ref}}^\dagger$. (b) Gate optimization for a target gate. (c) Optimization sequence for gates. (d) Substitution for U_{prep} .

The procedure for state preparation is almost the same as that for time evolution; we simply substitute an N -qubit brick-wall circuit U_{evol} with an $(N + 1)$ -qubit brick-wall circuit U_{prep}

and $U_{\text{ref}}^\dagger$ with $|0\rangle^{\otimes N+1} \langle S_{\text{MP}}|$ in Fig. 2(d). Here,

$$|S_{\text{MP}}\rangle = \frac{1}{\sqrt{2}}(|0\rangle|\psi'_g\rangle + |1\rangle|\psi'_{\text{ex}}\rangle), \quad (14)$$

which approximates $\frac{1}{\sqrt{2}}(|0\rangle|\psi_g\rangle + |1\rangle|\psi_{\text{ex}}\rangle)$, where the approximate ground and excited states, $|\psi'_g\rangle$ and $|\psi'_{\text{ex}}\rangle$, respectively, are prepared by the density-matrix renormalization group (DMRG) [41,42] as MPSs. The superposition state in Eq. (14) can be constructed from the MPSs $|\psi'_g\rangle$ and $|\psi'_{\text{ex}}\rangle$ [34], so we obtain $|S_{\text{MP}}\rangle$ through the normalization (and left orthogonalization) of $|0\rangle|\psi'_g\rangle + |1\rangle|\psi'_{\text{ex}}\rangle$.

Finally, we note that if $|\psi'_{\text{ex}}\rangle$ is replaced by $|0\rangle^{\otimes N}$, an absolute ground-state energy value can be calculated rather than the gap [34,43]. However, we checked numerically that the accuracy of the optimization of U_{prep} is poor, particularly for large-scale systems, and therefore, we do not investigate it in the present study.

C. Algorithmic error mitigation for a compressed time-evolution circuit

We construct the AEM formalism for a compressed time-evolution circuit based on the dynamical multiproduct formula [36]. Let U_{evol}^i be an operator approximating the time evolution of Δt with index i . The precise time-evolved density matrix $\rho(t)$ is expressed as

$$\rho_p(t) = |\psi_p(t)\rangle\langle\psi_p(t)|, \quad (15)$$

with

$$|\psi_p(t)\rangle = e^{-iHt}|\psi_{\text{prep}}\rangle, \quad (16)$$

where e^{-iHt} is a precise time-evolution operator, $|\psi_{\text{prep}}\rangle = U_{\text{prep}}|0\rangle^{\otimes N+1}$, and the MPO representation of $S_2(\frac{\Delta t}{m})^m$ with $m \gg 1$ is adopted as e^{-iHt} in our case. We define the approximate density matrix as

$$\mu(t) = \sum_i c_i(t) \rho_i(t), \quad (17)$$

where $\rho_i(t) = |\psi_i(t)\rangle\langle\psi_i(t)|$, $|\psi_i(t)\rangle = (U_{\text{evol}}^i)^{\frac{t}{\Delta t}}|\psi_{\text{prep}}\rangle$, and $c_i(t)$ is a real coefficient for time t . The indexed operator U_{evol}^i can be prepared by changing the accuracy in the time-evolution operator.

Here, we propose two procedures for preparing time-evolution operators in U_{evol}^i ; the first one is changing the number of slices m in the time-evolution MPO in Eq. (12) as

$$U_{\text{ref}}^{i\dagger} = S_2\left(\frac{-\Delta t}{m_i}\right)^{m_i} = \left\{ \prod_{\beta=1}^K \exp\left(iC_\beta P_\beta \frac{\Delta t}{2m_i}\right) \prod_{\beta=K}^1 \exp\left(iC_\beta P_\beta \frac{\Delta t}{2m_i}\right) \right\}^{m_i}. \quad (18)$$

The second one is changing the number of sweeps in the optimization of U_{evol} in Fig. 2(c).

We minimize the squared Frobenius norm

$$\begin{aligned} & \|\rho_p(t) - \mu(t)\|_F^2 \\ &= \text{Tr}[\rho_p(t)^\dagger \rho_p(t)] + \text{Tr}[\mu(t)^\dagger \mu(t)] - 2\text{ReTr}[\rho_p(t)^\dagger \mu(t)] \end{aligned}$$

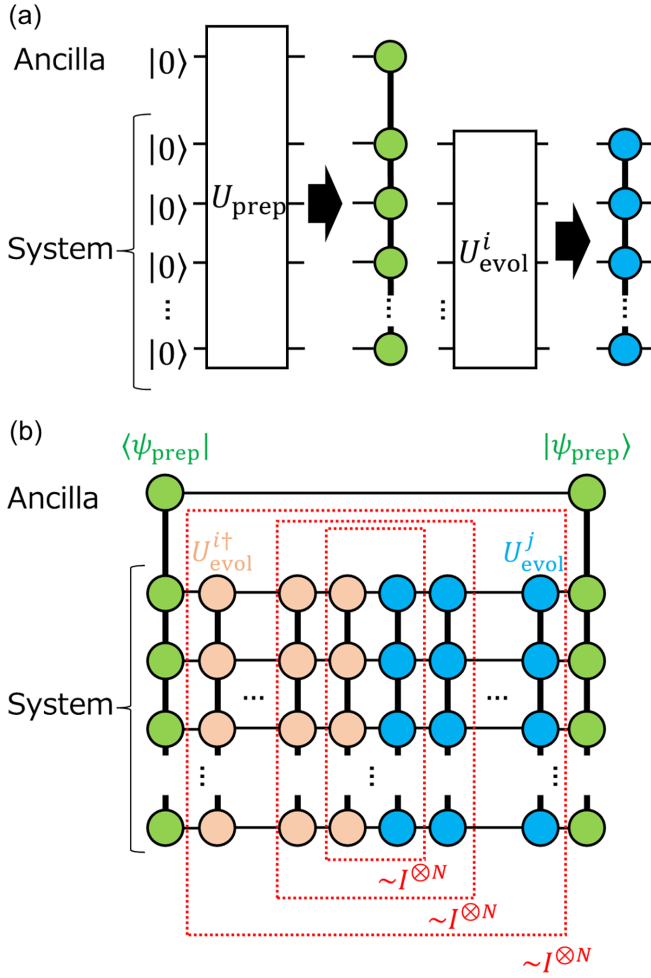


FIG. 3. Calculations for $M_{ij}(t)$ and $L_i(t)$. (a) Transformation from circuits of $|\psi_{\text{prep}}\rangle$ and U^i_{evol} to the MPS and MPO, respectively. (b) Calculation procedure for M_{ij} . Note that L_i can also be calculated in almost the same way.

$$= 1 + \sum_{ij} M_{ij}(t) c_i(t) c_j(t) - 2 \sum_i L_i(t) c_i(t). \quad (19)$$

Here, the matrix elements are defined as

$$M_{ij}(t) = |\langle \psi_{\text{prep}} | (U^i_{\text{evol}})^{\frac{t}{\Delta t}} (U^j_{\text{evol}})^{\frac{t}{\Delta t}} | \psi_{\text{prep}} \rangle|^2 \quad (20)$$

and

$$L_i(t) = |\langle \psi_{\text{prep}} | e^{iHt} (U^i_{\text{evol}})^{\frac{t}{\Delta t}} | \psi_{\text{prep}} \rangle|^2. \quad (21)$$

Since the objective function in Eq. (19) is quadratic in $c_i(t)$, the optimal coefficients can be obtained if we calculate $M_{ij}(t)$ and $L_i(t)$. Note that we used the constraints $\sum_i |c_i(t)| \leq 3$ and $\sum_i c_i(t) = 1$ in the optimization of $c_i(t)$ in the numerical demonstrations, where the latter corresponds to normalization of $\mu(t)$ since $\text{Tr}[\mu(t)] = \sum_i c_i(t) \text{Tr}[\rho_i(t)] = \sum_i c_i(t)$.

Figure 3 shows our implementation for calculating $M_{ij}(t)$ and $L_i(t)$. We first transform U^i_{evol} and $U_{\text{prep}}|0\rangle^{\otimes N+1}$ to the MPO and MPS, respectively, in Fig. 3(a). The MPS-MPO representation of $M_{ij}(t)$ is shown in Fig. 3(b). Naively, the calculation of $(U^j_{\text{evol}})^{t/\Delta t}$ requires exponential classical cost since $(U^j_{\text{evol}})^{t/\Delta t} \approx e^{-iHt}$. To mitigate the exponential growth

of t , we contract the network from the inside, as shown by the red dotted rectangles in Fig. 3(b), using the fact that $U^i_{\text{evol}} U^j_{\text{evol}} \approx I^{\otimes N}$ [36]. The computation of $L_i(t)$ can be carried out in the same manner by replacing U^i_{evol} with $e^{iH\Delta t}$.

The expectation value of $\mu(t)$ is represented using the optimized $c_i(t)$ as

$$\text{Tr}[O\mu(t)] = \sum_i c_i(t) \text{Tr}[O\rho_i(t)], \quad (22)$$

where O is an observable. So we can calculate $\text{Tr}[O\mu(t)]$ by executing the circuits for $\text{Tr}[O\rho_i(t)]$. We obtain the refined time-series data $m(\theta)$ in Eq. (2) from Eq. (22) by substituting $O = P(\theta)^\dagger U_{\text{prep}}|0\rangle\langle 0|^{\otimes N+1} U_{\text{prep}}^\dagger P(\theta)$ and $\mu(t) = \rho(t)$. Note that for the substituted O and $\mu(t)$, if $\rho_i(t)$ is affected by the depolarizing noise independent of i , the value becomes $\text{Tr}[O\mu'(t)] = [1 - p_{\text{dep}}(t)] \sum_i c_i(t) \text{Tr}[O\rho_i(t)] + \frac{p_{\text{dep}}(t)}{2^{N+1}}$, almost the same expression as Eq. (8).

D. Overlap enhancement for a compressed state-preparation circuit

In the circuit optimization described in Sec. II B, increasing the depth d_{prep} allows U_{prep} to prepare a state closer to $|S_{\text{MP}}\rangle$ in Eq. (14). However, the optimization time also grows exponentially with respect to d_{prep} due to the increase of the bond dimension in the MPS. The cost of contracting the tensor network is $O(\chi d 2^{4+2\lceil d/2 \rceil} + \chi^2 2^{2+2\lceil d/2 \rceil})$, where $d \in \{d_{\text{prep}}, d_{\text{evol}}\}$ and χ is the maximum bond dimension in the reference MPOs [24].

In reality, in the 37-qubit circuit for the one-dimensional Hubbard model presented later, performing 1000 sweeps requires only several tens of minutes on a consumer model laptop computer when $d_{\text{prep}} = 5$, whereas it takes more than a week on our high-performance computing system when $d_{\text{prep}} = 12$. The overlap was 0.97 for the 9-qubit circuit at $d_{\text{prep}} = 5$, whereas for the 37-qubit circuit it was only 0.76 even at $d_{\text{prep}} = 12$. Such degradation of the overlap generally leads to an increase in computational cost. For example, the cost of ground-state estimation via time-series QPE scales as η^{-4} with respect to the absolute value of overlap η between the initial state and the exact ground state [11]. Therefore, it is necessary to prepare a more accurate wave function using deeper circuits while avoiding exponential growth in the computational cost.

We propose a technique to alleviate this problem via iterative optimization. The procedure is as follows and is shown in Fig. 4.

(1) We set the index $k = 1$.

(2) As shown in Fig. 4(a), $U_{\text{prep}}^{(k)}$ is optimized to maximize the real part of the overlap $\text{Re}[\langle S_{\text{MP}}^{(k)} | U_{\text{prep}}^{(k)} | 0 \rangle^{\otimes N+1}]$, where we add the index k for $|S_{\text{MP}}\rangle$ and U_{prep} to identify the iteration count. The state $|S_{\text{MP}}^{(1)}\rangle$ is the target MPS that we want to approximate.

(3) As shown in Fig. 4(b), using the optimized circuit $U_{\text{prep}}^{*(k)}$, we create $|S_{\text{MP}}^{(k+1)}\rangle = U_{\text{prep}}^{*(k)\dagger} |S_{\text{MP}}^{(k)}\rangle$, which is closer to $|0\rangle^{\otimes N+1}$ than $|S_{\text{MP}}^{(k)}\rangle$.

(4) We increase $k \rightarrow k + 1$ for $U_{\text{prep}}^{(k)}$. Steps 2–4 are repeated until a termination condition is satisfied.

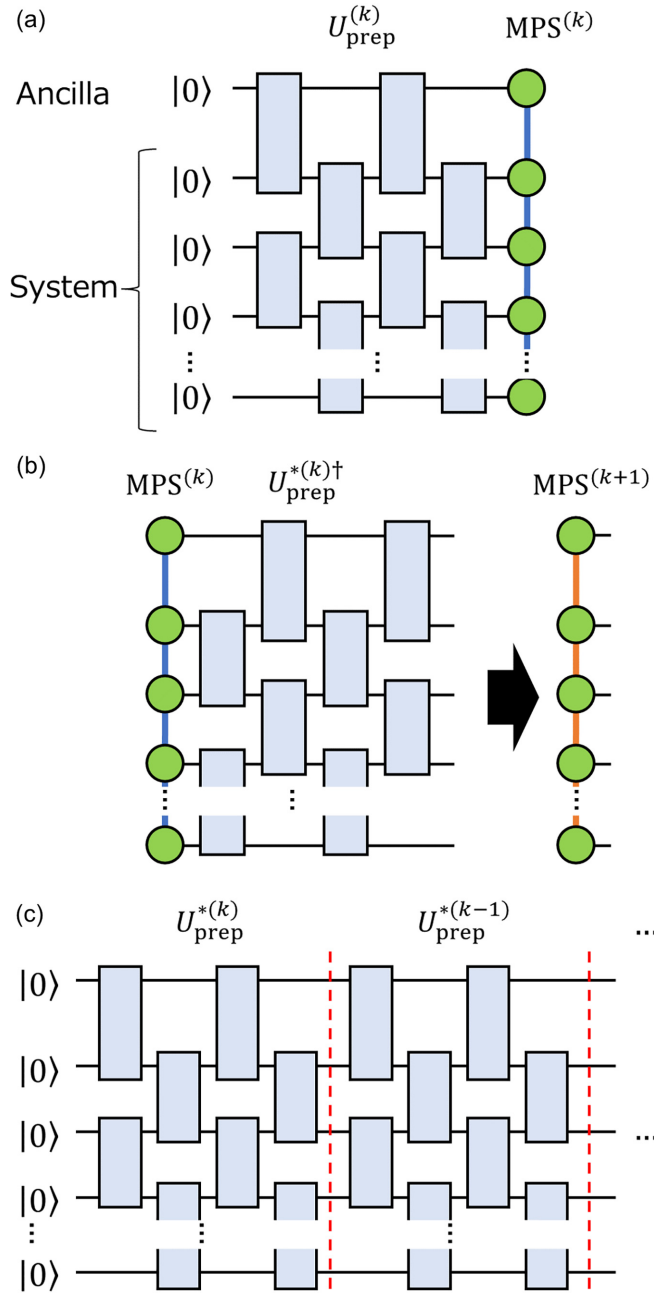


FIG. 4. Procedure for the overlap enhancement. (a) Circuit optimization. (b) Circuit merging. (c) Deep-circuit implementation.

(5) As shown in Fig. 4(c), we obtain a deep circuit $U_{\text{prep}} = \prod_{k=k_{\text{max}}}^1 U_{\text{prep}}^{*(k)}$. U_{prep} give a higher overlap than a single $U_{\text{prep}}^{(1)}$ since

$$\begin{aligned}
 \langle S_{\text{MP}}^{(1)} | U_{\text{prep}} | 0 \rangle^{\otimes N+1} &= \langle S_{\text{MP}}^{(1)} | \prod_{k=k_{\text{max}}}^1 U_{\text{prep}}^{*(k)} | 0 \rangle^{\otimes N+1} \\
 &= \langle S_{\text{MP}}^{(2)} | \prod_{k=k_{\text{max}}}^2 U_{\text{prep}}^{*(k)} | 0 \rangle^{\otimes N+1} \\
 &= \dots \\
 &= \langle S_{\text{MP}}^{(k_{\text{max}})} | 0 \rangle^{\otimes N+1}. \quad (23)
 \end{aligned}$$

Note that the bond dimension of $|S_{\text{MP}}^{(k)}\rangle$ will increase for k in general, but it may decrease in a high-overlap regime since when $\text{Re}[\langle S_{\text{MP}}^{(k)} | U_{\text{prep}}^{*(k)} | 0 \rangle^{\otimes N+1}] \approx 1$, $|S_{\text{MP}}^{(k+1)}\rangle = U_{\text{prep}}^{*(k)\dagger} |S_{\text{MP}}^{(k)}\rangle \approx |0\rangle^{\otimes N+1}$, i.e., a separable state. Also, a related approach has been proposed in which, instead of maximizing the overlap with the MPS, each gate is optimized and merged into the MPS to reduce the entanglement entropy of the MPS [31]. The gate optimization is executed only once for each gate in the approach, so it can be useful when prioritizing the improvement of the overlap over circuit-depth efficiency.

III. NUMERICAL DEMONSTRATION

A. Models and calculation conditions

We execute numerical experiments using a noiseless simulator and real devices. We estimate the energy gap between the ground-state and first-excited-state energies, which is represented as $\Delta_{(0,1)}^{\text{est}}$. The target system is the one-dimensional Hubbard model:

$$\begin{aligned}
 H = & -T \sum_{q=1}^{n_s-1} \sum_{\sigma \in \{\uparrow, \downarrow\}} (a_{q+1\sigma}^\dagger a_{q\sigma} + a_{q\sigma}^\dagger a_{q+1\sigma}) \\
 & + U \sum_{q=1}^{n_s} n_{q\uparrow} n_{q\downarrow} - \frac{U}{2} \sum_{q=1}^{n_s} (n_{q\uparrow} + n_{q\downarrow}), \quad (24)
 \end{aligned}$$

where q (σ) is the orbital (spin) index, $T = 1$ is the unitless hopping energy, U is the on-site Coulomb energy, $a_{q\sigma}^\dagger$ ($a_{q\sigma}$) is the creation (annihilation) operator, and $n_{q\sigma}$ is the number operator $n_{q\sigma} = a_{q\sigma}^\dagger a_{q\sigma}$. We choose $U = 10$, corresponding to the strong-correlation regime; the number of system qubits N is $2n_s$, where n_s is the number of physical sites, and the fermion-qubit mapping is a Jordan-Wigner transformation [44] with the up-down qubit sequence $1 \uparrow, 1 \downarrow, 2 \uparrow, \dots, n_s \downarrow$.

Our default conditions are as follows. We calculate the circuits of the $N = 8, 36, 52$ system qubits with one ancilla qubit. The value of d_{prep} is 5 for the 9-qubit circuit and 12 for the 37- and 53-qubit circuits, and $d_{\text{evol}} = 5$. The number of sweeps is 10^3 for U_{prep} . For U_{evol} , the number is 10^4 . The cutoff value of the SVD in the optimization of U_{prep} and U_{evol} is 10^{-12} . The number of slices m in Eq. (12) is $\Delta t/0.001$. The number of shots in a circuit execution is 10^5 . The maximum number of time steps is 100 in the next section and 50 in the others, but the execution is stopped when no signal is found for a time step.

We calculate reference energies for the gap $\Delta_{(0,1)}^{\text{ref}}$ using the exact diagonalization for the eight-qubit model and DMRG with a maximum bond dimension of 1000 for the other large models. The tensor-network compression was executed by itensor [45], and the circuits were executed by qiskit [46]. We use the Q-CTRL error-suppression module, Fire Opal [47,48], for real device demonstrations. We use the Aer simulator for the noiseless simulations and IBM Heron devices for real device execution.

The other conditions are the same as in [34].

B. Time-step-size dependences

We confirmed the accuracy of the time step size Δt in the nine-qubit circuit. Figure 5 shows the error of the estimated gap from the reference gap for each time step, $\Delta_{(0,1)}^{\text{est}} - \Delta_{(0,1)}^{\text{ref}}$,

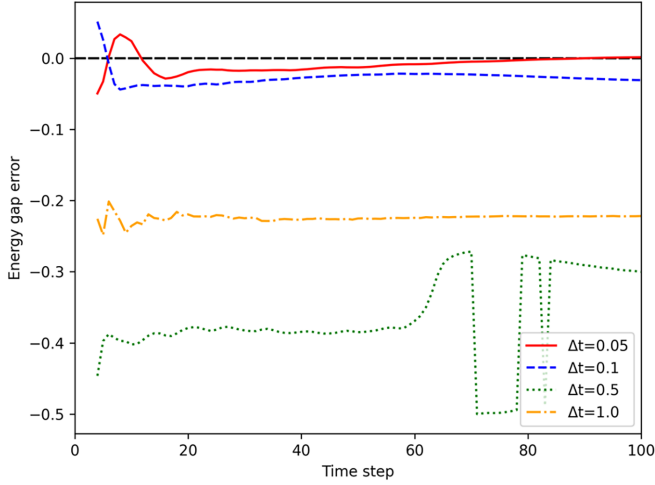


FIG. 5. Energy-gap errors $\Delta_{(0,1)}^{\text{est}} - \Delta_{(0,1)}^{\text{ref}}$ for a time step with several step sizes Δt .

where the exact gap $\Delta_{(0,1)}^{\text{ref}}$ is 0.254, and the black dashed horizontal line represents the zero-gap error. The gap at a given time step was computed using the time-series data up to that point. Note that because the step size differs between plots, the total time t corresponding to the same step number is not identical in each plot.

For $\Delta t = 0.1$ and 0.05 , the final absolute error was below 0.1 . Excluding $\Delta t = 0.5$, the fluctuations stabilize around step 50 regardless of Δt . For $\Delta t = 0.5$, the results became unstable after 60 steps, likely due to the accumulation of unitary-approximation errors. Further considering that large-scale hardware demonstrations may not reliably reach 100 steps, we use 50 steps in the analysis starting in the next section.

Regarding the time-step size, noiseless simulations are performed with $\Delta t = 0.05$ in the following section, where the absolute relative errors at 50 and 100 steps were 0.012 and 0.001 , corresponding to error ratios of 4.7% and 0.4% , respectively. Here, we aim for target accuracy as $T/100 = 0.01$ [49]. On the other hand, we set $\Delta t = 0.1$ in real device demonstrations since Fire Opal provided the best balance between algorithmic precision and hardware noise for this value.

C. Verification of algorithmic error mitigation

We verify the AEM for the two cases of preparation: changing the number of slices and sweeps. Figures 6(a) and 6(b) show the slice and sweep dependences for the gap in the $(N + 1 = 9)$ -qubit circuit. We do not find a significant improvement for the slice case. On the other hand, we do find an improvement in the sweep case. The degree of improvement depends on the included sweep sets, but the absolute error becomes less than 0.01 for all of the sweep lists.

We can reduce the calculation cost by choosing rough cutoffs for SVD in the calculation procedure for $M_{ij}(t)$ and $L_i(t)$. Figures 7(a) and 7(c) show the maximum bond dimension of $L_i(t)$, precisely that of $e^{iHt}(U_{\text{evol}}^i)^{t/\Delta t}$, for the time step in the slice and sweep cases, respectively. The values rapidly

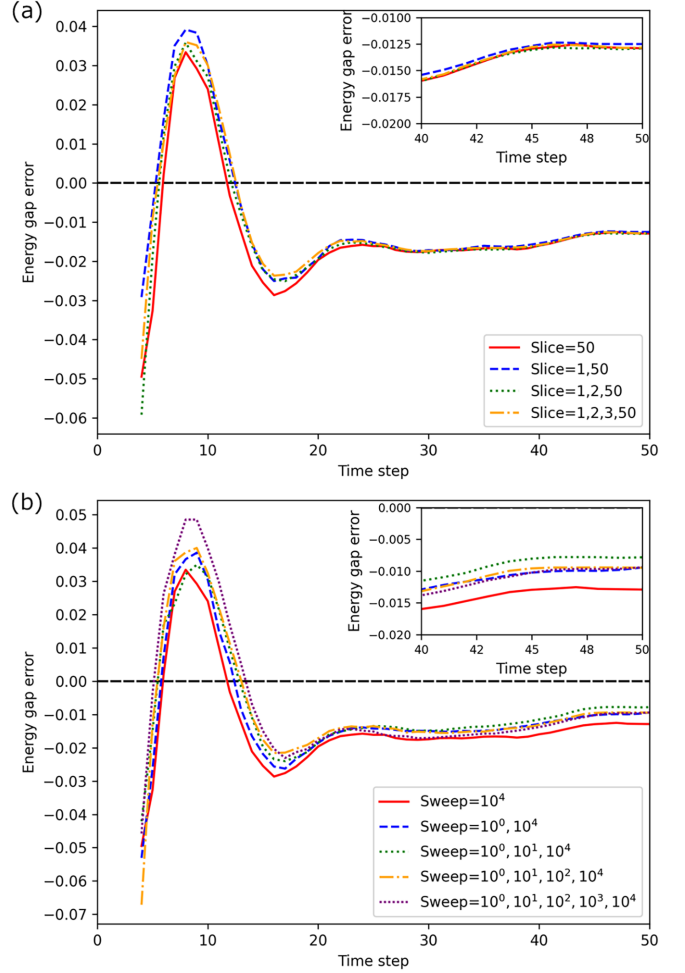


FIG. 6. Energy-gap errors $\Delta_{(0,1)}^{\text{est}} - \Delta_{(0,1)}^{\text{ref}}$ with and without the AEM. The inset shows an enlarged view. (a) Dependence on slice sets. (b) Dependence on sweep sets.

increase for the step and saturate to the maximum value of 256 , but the growth is suppressed by choosing rough cutoffs, where the maximum bond dimension for the $(N + 1)$ -qubit MPO is $4^{\lfloor \frac{N+1}{2} \rfloor}$ [42]. Figures 7(b) and 7(d) show the cutoff dependences for the gap for the time step in the slice and sweep cases, respectively. From the sweep case, the AEM will work correctly up to the cutoff value of 10^{-8} .

D. Verification of overlap enhancement

We evaluated the overlap enhancement for a circuit with $N + 1 = 37$ qubits, where the 9-qubit circuit was not tested since its accuracy was already sufficient at $d_{\text{prep}} = 5$. We first performed test calculations with the depth increment d_{prep} per iteration set to 6, the number of sweeps for each optimization of $U_{\text{prep}}^{(k)}$ set to 10, and the SVD cutoff during optimization set to 10^{-8} . Figure 8(a) shows the evolution of the overlap as a function of the sweep count; the dashed vertical lines indicate where the iteration index is incremented. The overlap increases as the number of sweeps grows, and the procedure successfully ran up to $d_{\text{prep}} = 24$ without running out of memory, which was not achievable with the previous method. However, the final overlap 0.62 was much smaller than the

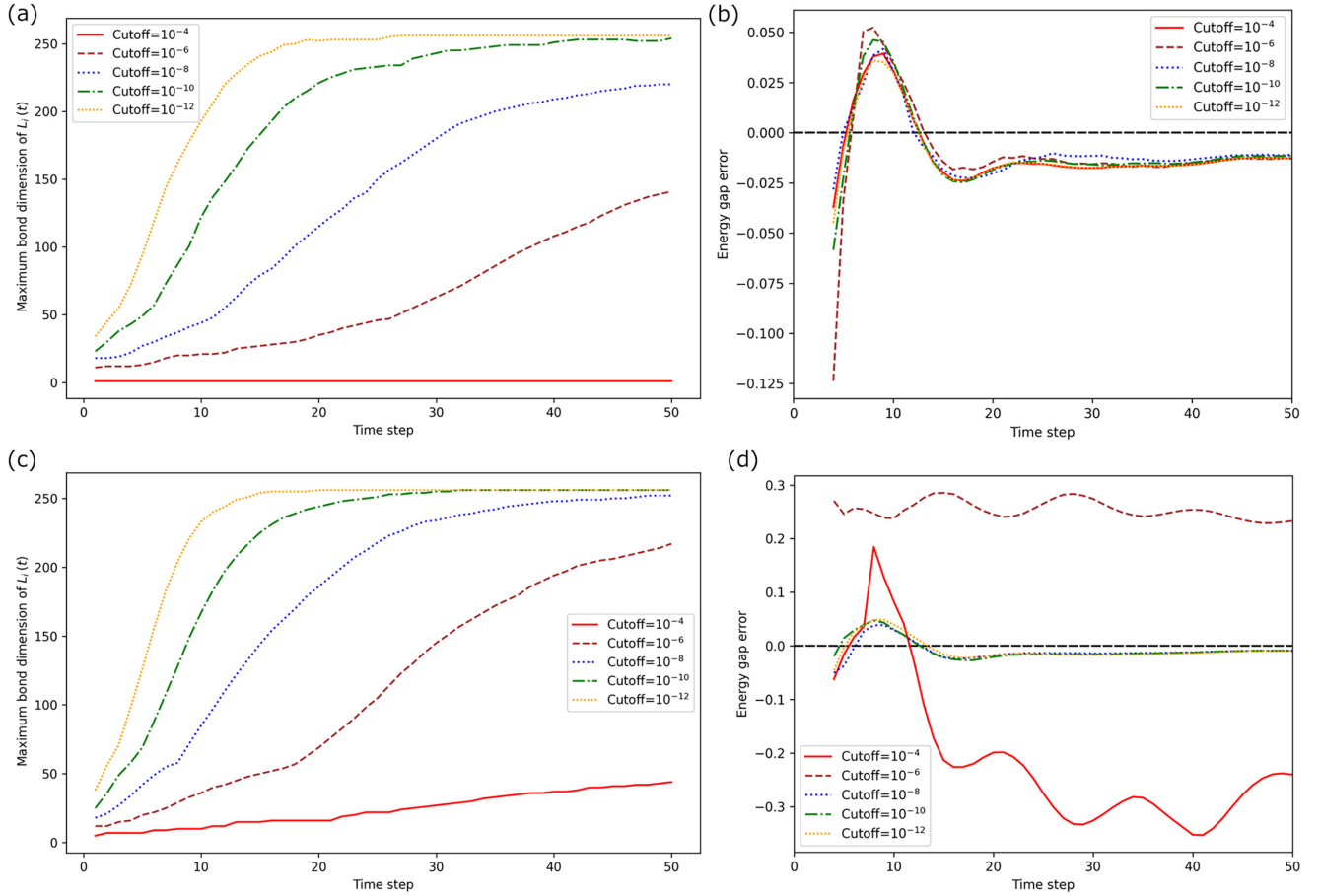


FIG. 7. SVD dependences for the bond dimension and energy-gap errors, $\Delta_{(0,1)}^{\text{est}} - \Delta_{(0,1)}^{\text{ref}}$. Plots for the (a) and (b) slice and (c) and (d) sweep cases. (a) and (c) represent the maximum bond dimensions for $L_i(t)$, and (b) and (d) represent the gap for the time step.

value obtained for $d_{\text{prep}} = 12$ in the previous approach, 0.76. This indicates that unless d_{prep} per iteration and the sweep count are sufficiently large, increasing the circuit depth does not yield a significant overlap improvement.

The results for the case with d_{prep} per iteration equal to 12 and 1000 sweeps for each $U_{\text{prep}}^{(k)}$ are shown in Fig. 8(b). The calculation ran out of memory at $k = 2$. The resulting overlap was 0.81, which is larger than 0.76. These results demonstrate that, with appropriate choices of circuit depth per iteration and sweep conditions, it is possible to prepare states with larger overlap than in the previous approach.

E. Real device demonstration

Figure 9(a) shows the results of gap estimation for the nine-qubit circuit with $\Delta t = 0.1$. From the noiseless simulator results, we find that even at $\Delta t = 0.1$, higher accuracy is achieved with the AEM (blue) than without the AEM (red). Here, the AEM uses circuits with sweep counts of 10^0 , 10^1 , and 10^4 . The obtained values are -0.023 and -0.009 , respectively, indicating that the use of the AEM yields an accuracy on the order of 10^{-3} . Similar behavior is observed in the real device demonstration, where the result with the AEM (yellow) also achieves a high accuracy of 0.001, which is better than that without the AEM, 0.033 (green). The numbers of two-

qubit (cz and rzz) gates in the final steps with and without the AEM are 1699 and 1698, respectively.

Figures 9(b) and 9(c) show the results for the 37- and 53-qubit circuits, respectively. We executed 11 trials since the large variances were expected due to rapid signal decay: Only up to 19 and 11 steps, respectively, were executed before the signal was lost. The errors were on the order of 0.1 for the averages (red lines) but of the order of 0.01 or less in the best trials, for which the reference gap values $\Delta_{(0,1)}^{\text{ref}}$ are 0.076 and 0.055, mean gap errors are 0.343 and -0.524 , and the numbers of two-qubit gates are 4146 and 4091 for the 37- and 53-qubit models, respectively. The zero errors lie within the error bars, where the standard deviations are 0.660 and 0.595 for the 37- and 53-qubit models, respectively.

We note that the classical computation in the AEM, specifically, the evaluation of $L_i(t)$, was halted at 11 steps for those systems, for which the suppression of the cost increase by the implementation shown in Fig. 3(b) is discussed in Appendix B. Although an efficient implementation of AEM may be needed, from the viewpoint of scaling, it may be possible to achieve similar scaling of the accuracy in time evolution even without using AEM. See Appendix C for an analysis. We also note that hardware execution was attempted for the 37-qubit model with d_{prep} per iteration equal to 2 and $d_{\text{prep}} = 24$, but the run terminated after four steps, with a gate count of 2684.

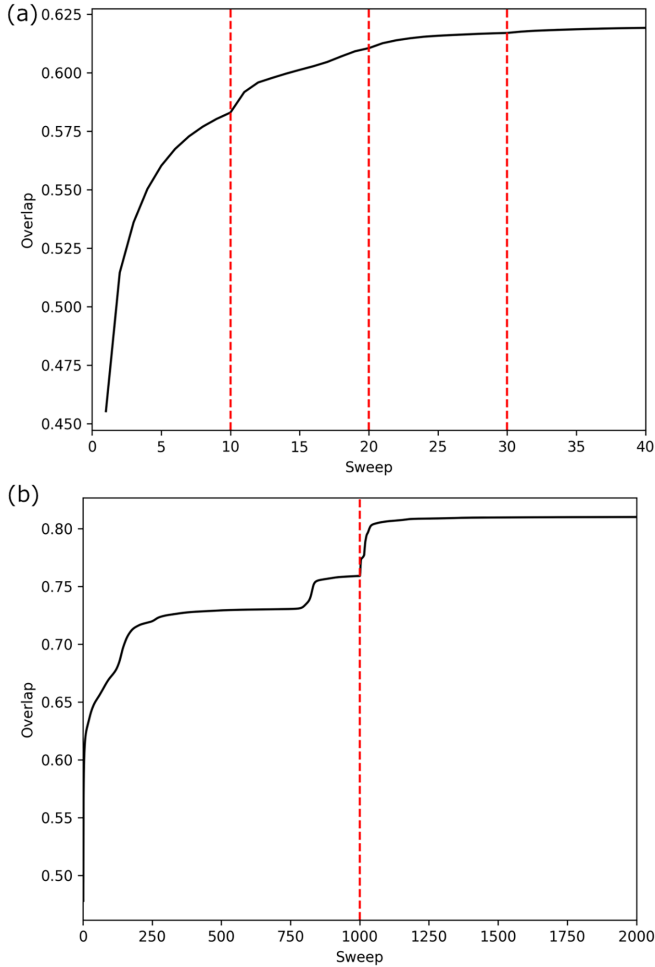


FIG. 8. Overlap enhancement. The vertical break line in red represents the increment of k . (a) Verification for d_{prep} per iteration equal to 6 with 10 sweeps for each k . (b) Verification for d_{prep} per iteration equal to 12 with 1000 sweeps for each k .

From these results, our algorithm not only demonstrates good performance in large-scale QPE-type algorithms but also highlights challenges that remain for the future: The proposed method has been shown to be an efficient approach well suited for current devices. With future improvements in gate performance, it is expected that a large-scale demonstration of QPE-type algorithms could be executed with higher precision.

IV. CONCLUSION

In this study, we proposed a phase-estimation-type algorithm that extracts time-series data related to the energy gap from quantum circuits compressed using tensor networks. The circuits considered here use only a single ancilla qubit and system qubits and are composed of only neighboring gates. By combining measurement results from four types of circuits, time-series data represented as complex numbers can be obtained. Numerical validation showed that, with an appropriate choice of the time-step size, the results converge within 50 steps regardless of the time step. In addition, we proposed techniques to improve the accuracy of both time evolution and state preparation in our algorithm. The former

is based on AEM, while the latter relies on iterations combining circuit optimization with MPS merging. We executed the time-series algorithm on noiseless devices and IBM quantum hardware using Q-CTRL's error-suppression module. For one-dimensional Hubbard models with 8, 36, and 52 qubits, accuracies on the order 10^{-3} or larger were observed depending on conditions.

Future work may include achieving higher accuracy and larger-scale implementations by developing more efficient algorithms on both the quantum and classical sides. For example, tensor networks other than an MPS could be employed [27–29,32]. In addition, the reduction of classical computational overhead is important since the AEM and quantum computational cost associated with improving state-preparation accuracy are both large. Also, there is a possibility that the sampling cost could be reduced by modifying the protocol. In this work, from the standpoint that it is difficult to estimate the maximum executable time t_{max} and the decay of the signal s_t in advance, we obtained s_t for sequential values from $t = 1$ to t_{max} . If the decay can be estimated easily or if there is no decay due to error correction, it may be possible to use a more efficient protocol that achieves the Heisenberg limit [11,13] by adaptive strategies. Finally, since this algorithm is crucial not only for current devices but also for FTQCs, verification of QPE focusing on the FTQC era is also promising.

ACKNOWLEDGMENTS

Part of this work was performed for the Council for Science, Technology and Innovation (CSTI), Cross-ministerial Strategic Innovation Promotion Program (SIP), “Promoting the application of advanced quantum technology platforms to social issues” (Funding agency: QST). Also, it was supported by MEXT Quantum Leap Flagship Program Grants No. JPMXS0118067285 and No. JPMXS0120319794. Some of the calculations were performed on the Mitsubishi Chemical Corporation (MCC) high-performance computer (HPC) system NAYUTA (NAYUTA is a nickname for the MCC HPC and is not a product or service name of MCC). We acknowledge the use of IBM Quantum services for experiments in this paper. The views expressed are those of the authors and do not reflect the official policy or position of IBM or the IBM Quantum team. We thank Q-CTRL for the support of the error-suppression module. S.K. thanks Keithley Kimberlee for careful reading of the manuscript and for helpful suggestions on English usage. K.S. acknowledges partial support from the Center of Innovations for Sustainable Quantum AI (Grant No. JPMJPF2221) from the Japan Science and Technology Agency (JST), Japan, and Grants-in-Aid for Scientific Research C (Grant No. 21K03407) and for Transformative Research Area B (Grant No. 23H03819) from the Japan Society for the Promotion of Sciences (JSPS), Japan.

DATA AVAILABILITY

The data that support the findings of this article are openly available [50].

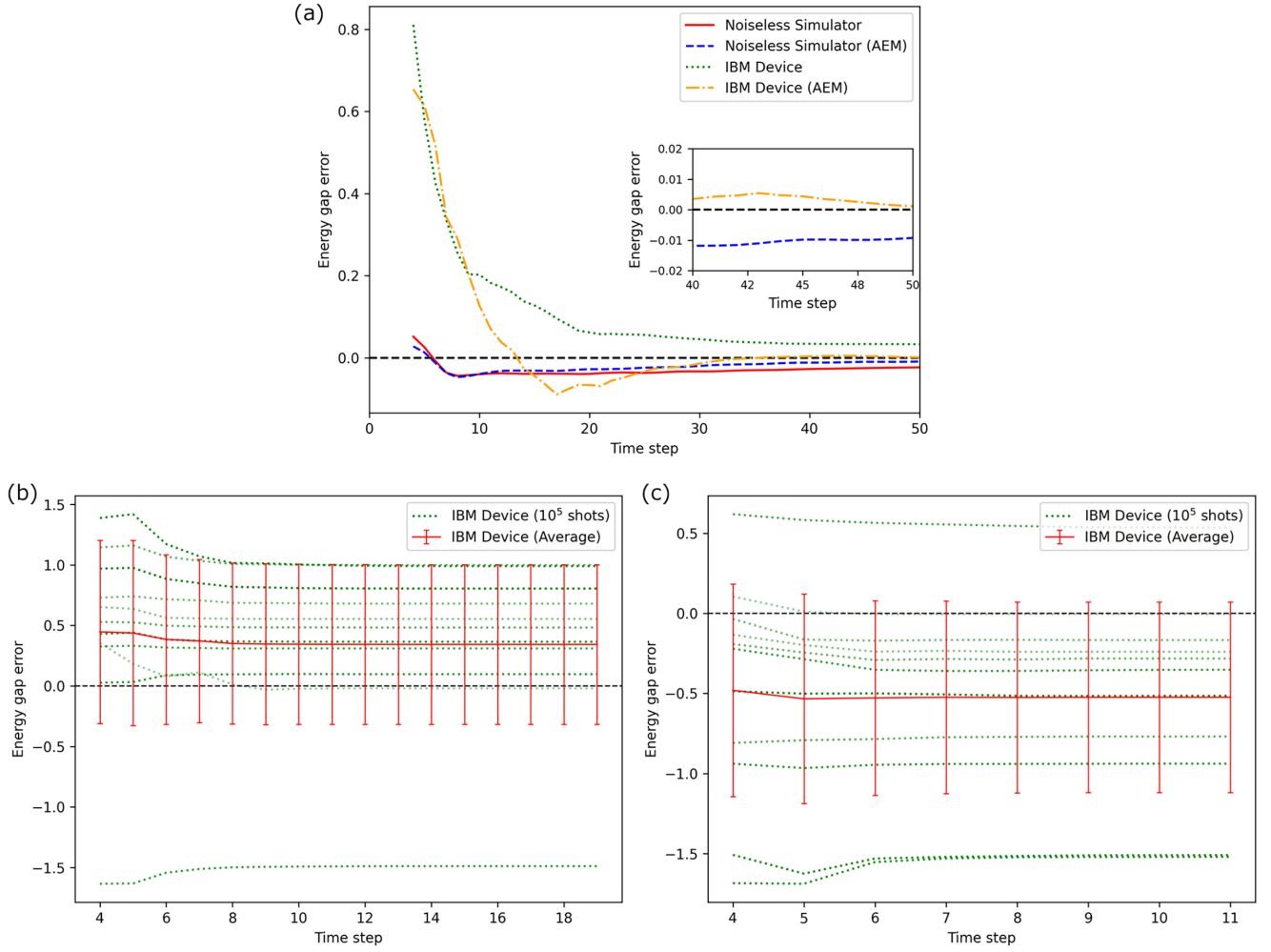


FIG. 9. Real device demonstrations for energy-gap errors $\Delta_{(0,1)}^{\text{est}} - \Delta_{(0,1)}^{\text{ref}}$ using IBM Kawasaki. The inset in (a) shows an enlarged view. The error bars in (b) and (c) represent standard deviations, and 11 trials are represented by gradated green dotted lines. (a) Nine-qubit circuit results, including noiseless simulator results, (b) 37-qubit circuit results, and (c) 53-qubit circuit results.

APPENDIX A: EIGENVALUE EXTRACTION FROM TIME-SERIES DATA

Assume that the signal s_t is obtained from the quantum circuits as

$$s_t = e(t) \sum_J P_J e^{-i\Delta_J t}, \quad (\text{A1})$$

where $e(t)$ is signal attenuation and the sampling times are $t \in \{t_r\}$ ($t_r = t_1, t_2, \dots, t_{\max}$) with fixed intervals $\Delta t = t_{i+1} - t_i$. The signal attenuation can be taken as $e(t) \approx e^{-\alpha t}$ under depolarizing noise, although it depends on circuit compilation details.

We adopt the matrix pencil method as an initial guess. We construct the shifted matrices

$$A_0 = \begin{pmatrix} s_1 & s_2 & \cdots & s_l \\ s_2 & s_3 & \cdots & s_{l+1} \\ \vdots & \vdots & \ddots & \vdots \\ s_{m-1} & s_m & \cdots & s_{m+l-2} \end{pmatrix} \quad (\text{A2})$$

and

$$A_1 = \begin{pmatrix} s_2 & s_3 & \cdots & s_{l+1} \\ s_3 & s_4 & \cdots & s_{l+2} \\ \vdots & \vdots & \ddots & \vdots \\ s_m & s_{m+1} & \cdots & s_{m+l-1} \end{pmatrix}, \quad (\text{A3})$$

where we choose $l = t_{\max}/2$ and $m = t_{\max}/2 + 1$. The eigenvalues are obtained by solving the generalized eigenvalue problem

$$A_1 x = \lambda A_0 x. \quad (\text{A4})$$

The eigenvalue corresponding to the dominant spectral component is expressed as

$$\lambda \approx g e^{-(i\Delta_J + \alpha)\Delta t}, \quad (\text{A5})$$

where Δ_J corresponds to the desired eigenvalue and g is a constant. The amplitude P_J is determined by fitting g as

$$\min_{P_J} \sum_t |s_t - P_J e^{-(i\Delta_J + \alpha)\Delta t}|^2, \quad (\text{A6})$$

using the least-squares method.

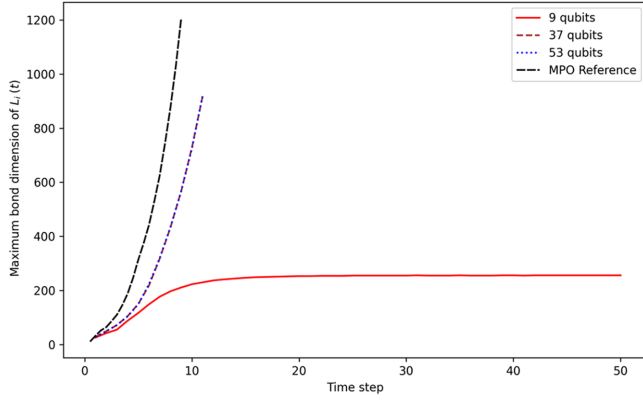


FIG. 10. SVD dependences for the bond dimension in $\Delta t = 0.1$. The calculations for M_{ij} and L_i are executed using the sweep of 10^0 , 10^1 , and 10^4 counts. The reference is the value for the MPO of $(U_{\text{ref}}^\dagger)^{\frac{t}{2\Delta t}}$ in the 52-qubit system.

Then we refine the eigenvalue by nonlinear simultaneous optimization of P_J , Δ_J , and α (i.e., data fitting) using the initial guess:

$$\min_{P_J, \Delta_J, \alpha} \sum_t |s_t - P_J e^{-(i\Delta_J + \alpha)\Delta t}|^2, \quad (\text{A7})$$

where we used the least-squares method for the optimization.

APPENDIX B: BOND DIMENSION FOR REAL DEVICE EXECUTIONS

Figure 10 shows the SVD dependences for the bond dimension in Fig. 9. The value in the 9-qubit circuit (red) saturates at 256. The values in the larger models (brown for the 37-qubit circuit and blue for the 53-qubit circuit) are higher. The values are almost the same for the 37- and 53-qubit circuits, indicating that the increase is independent of size for large systems. We also show the values for the MPO of $(U_{\text{ref}}^\dagger)^{\frac{t}{2\Delta t}}$ for 52 qubits for reference, where $\frac{t}{2\Delta t}$ (half-time-step value) comes from the fact that two MPOs merge in the calculation of $M_{ij}(t)$ and $L_i(t)$ in Fig. 3(b), and the SVD cutoff is 10^{-8} . We find that the increase in calculation cost is suppressed in the AEM due to the merge. Considering that a bond dimension greater than 10^4 is tractable by a high-end high-performance computing system [51], it is expected that the use of more powerful classical devices or efficient contraction schemes may enable AEM to be executed for longer durations.

APPENDIX C: DEPTH SCALING FOR A COMPRESSED TIME-EVOLUTION CIRCUIT

We analyze the depth scaling of d_{evol} in the one-dimensional Hubbard model. AEM was introduced to mitigate

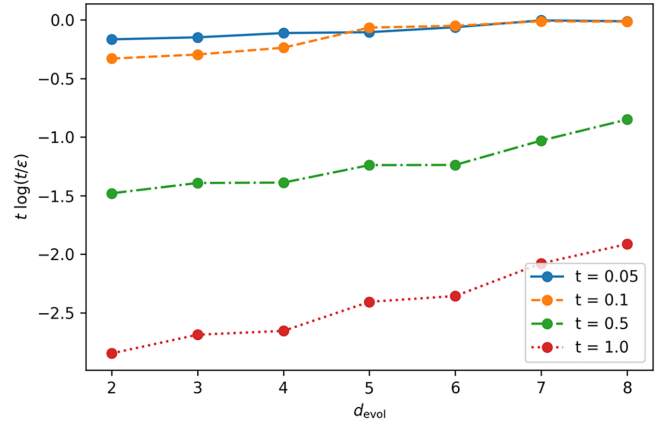


FIG. 11. Scaling for the depth d_{evol} in the Hubbard model. We adopt Δt and $\|U_{\text{ref}}^\dagger - U_{\text{evol}}\|_F$ as t and ε in the compressed circuits, respectively.

Trotter errors [35], reducing the order of the error in p th-order Trotter approximation from $O(t^{1+\frac{1}{p}}/\varepsilon^{\frac{1}{p}})$ to the near-optimal scaling for ε , $O(t \text{ polylog}(t/\varepsilon))$, without relying on coherent techniques [52,53]. On the other hand, in addition to AEM, a procedure to decompose the time-evolution operator of a short-range Hamiltonian into gates with circuit depth $O(t \text{ polylog}(t/\varepsilon))$ using the Lieb-Robinson bound exists [54]. In a related work [32], it was numerically suggested that a tensor-compressed time-evolution circuit can also be implemented with this scaling, where a one-dimensional Heisenberg model was adopted.

We therefore examined whether this scaling holds for the one-dimensional Hubbard model as well based on a procedure in Ref. [32]. For several values of Δt , we computed U_{evol} while varying d_{evol} . To confirm the dependence of $d_{\text{evol}} \sim t \log(t/\varepsilon)$, we plotted $t \log(t/\varepsilon)$ for d_{evol} by setting $t = \Delta t$ and $\varepsilon = \|U_{\text{ref}}^\dagger - U_{\text{evol}}\|_F$, as shown in Fig. 11, where the eight-qubit model is chosen. The plots look linear for each Δt , suggesting a $\log(1/\varepsilon)$ dependence for ε .

Note that we numerically checked the dependence on t as in Ref. [32] using a least-squares fitting for

$$d_{\text{evol}}(t, \varepsilon) = (u_0 + u_1 t^{u_2}) \log\left(\frac{t}{\varepsilon}\right) + (u_3 + u_4 t^{u_5}), \quad (\text{C1})$$

where $u_0, u_1, \dots, u_5$ are parameters. We obtained the values $u_0 = 1.40, u_1 = 5.50, u_2 = 1.44, u_3 = 6.78, u_4 = 14.95$, and $u_5 = 1.19$, with $R^2 = 0.89$.

[1] A. Yu. Kitaev, Quantum measurements and the Abelian Stabilizer Problem, [arXiv:quant-ph/9511026](https://arxiv.org/abs/quant-ph/9511026).

[2] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Quantum algorithms revisited, *Proc. Math. Phys. Eng. Sci.* **454**, 339 (1998).

- [3] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th anniversary ed. (Cambridge University Press, Cambridge, 2010).
- [4] H. Gao, S. Imamura, A. Kasagi, and E. Yoshida, Distributed implementation of full configuration interaction for one trillion determinants, *J. Chem. Theory Comput.* **20**, 1185 (2024).
- [5] B. L. Higgins, D. W. Berry, S. D. Bartlett, H. M. Wiseman, and G. J. Pryde, Entanglement-free Heisenberg-limited phase estimation, *Nature (London)* **450**, 393 (2007).
- [6] S. Kimmel, G. H. Low, and T. J. Yoder, Robust calibration of a universal single-qubit gate set via robust phase estimation, *Phys. Rev. A* **92**, 062315 (2015).
- [7] N. Wiebe and C. Granade, Efficient Bayesian phase estimation, *Phys. Rev. Lett.* **117**, 010503 (2016).
- [8] R. D. Somma, Quantum eigenvalue estimation via time series analysis, *New J. Phys.* **21**, 123025 (2019).
- [9] T. E. O'Brien, B. Tarasinski, and B. M. Terhal, Quantum phase estimation of multiple eigenvalues for small-scale (noisy) experiments, *New J. Phys.* **21**, 023022 (2019).
- [10] A. Dutkiewicz, B. M. Terhal, and T. E. O'Brien, Heisenberg-limited quantum phase estimation of multiple eigenvalues with few control qubits, *Quantum* **6**, 830 (2022).
- [11] L. Lin and Y. Tong, Heisenberg-limited ground-state energy estimation for early fault-tolerant quantum computers, *PRX Quantum* **3**, 010318 (2022).
- [12] G. Wang, D. S. Franca, R. Zhang, S. Zhu, and P. D. Johnson, Quantum algorithm for ground state energy estimation using circuit depth with exponentially improved dependence on precision, *Quantum* **7**, 1167 (2022).
- [13] Z. Ding and L. Lin, Even shorter quantum circuit for phase estimation on early fault-tolerant quantum computers with applications to ground-state energy estimation, *PRX Quantum* **4**, 020331 (2023).
- [14] Z. Ding and L. Lin, Simultaneous estimation of multiple eigenvalues with short-depth quantum circuit on early fault-tolerant quantum computers, *Quantum* **7**, 1136 (2023).
- [15] Z. Ding, Y. Dong, T. Yu, and L. Lin, Robust ground-state energy estimation under depolarizing noise, *arXiv:2307.11257*.
- [16] N. S. Blunt, L. Caune, R. Izsák, E. T. Campbell, and N. Holzmann, Statistical phase estimation and error mitigation on a superconducting quantum processor, *PRX Quantum* **4**, 040341 (2023).
- [17] M. S. Rudolph, J. Chen, J. Miller, A. Acharya, and A. Perdomo-Ortiz, Decomposition of matrix product states into shallow quantum circuits, *Quantum Sci. Technol.* **9**, 015012 (2024).
- [18] S.-J. Ran, Encoding of matrix product states into quantum circuits of one- and two-qubit gates, *Phys. Rev. A* **101**, 032310 (2020).
- [19] S.-H. Lin, R. Dilip, A. G. Green, A. Smith, and F. Pollmann, Real- and imaginary-time evolution with compressed quantum circuits, *PRX Quantum* **2**, 010342 (2021).
- [20] J. Dborin, F. Barratt, V. Wimalaweera, L. Wright, and A. G. Green, Matrix product state pre-training for quantum machine learning, *Quantum Sci. Technol.* **7**, 035014 (2022).
- [21] J. Dborin, V. Wimalaweera, F. Barratt, E. Ostby, T. E. O'Brien, and A. G. Green, Simulating groundstate and dynamical quantum phase transitions on a superconducting quantum computer, *Nat. Commun.* **13**, 5977 (2022).
- [22] C. Mc Keever and M. Lubasch, Classically optimized Hamiltonian simulation, *Phys. Rev. Res.* **5**, 023146 (2023).
- [23] B. Anselme Martin, T. Ayril, F. Jamet, M. J. Rančić, and P. Simon, Combining matrix product states and noisy quantum computers for quantum simulation, *Phys. Rev. A* **109**, 062437 (2024).
- [24] L. Causer, F. Jung, A. Mitra, F. Pollmann, and A. Gammon-Smith, Scalable simulation of nonequilibrium quantum dynamics via classically optimized unitary circuits, *Phys. Rev. Res.* **6**, 033062 (2024).
- [25] T. Shirakawa, H. Ueda, and S. Yunoki, Automatic quantum circuit encoding of a given arbitrary quantum state, *Phys. Rev. Res.* **6**, 043008 (2024).
- [26] C. Mc Keever and M. Lubasch, Towards adiabatic quantum computing using compressed quantum circuits, *PRX Quantum* **5**, 020362 (2024).
- [27] M. D'Anna, Y. Zhang, R. Wiersema, M. S. Rudolph, and J. Carrasquilla, Circuit compression for 2D quantum dynamics, *arXiv:2507.01883*.
- [28] S. Sugawara, K. Inomata, T. Okubo, and S. Todo, Embedding of tree tensor networks into shallow quantum circuits, *arXiv:2501.18856*.
- [29] J. Gibbs and L. Cincio, Deep circuit compression for quantum dynamics via tensor networks, *Quantum* **9**, 1789 (2025).
- [30] I. N. M. Le, S. Sun, and C. B. Mendl, Riemannian quantum circuit optimization based on matrix product operators, *Quantum* **9**, 1833 (2025).
- [31] R. Mansuroglu and N. Schuch, Preparation circuits for matrix product states by classical variational disentanglement, *Phys. Rev. A* **113**, 042430 (2026).
- [32] E. Karacan, Phase estimation with compressed controlled time evolution, *Phys. Rev. A* **113**, 042420 (2026).
- [33] K. Sugisaki, C. Sakai, K. Toyota, K. Sato, D. Shiomi, and T. Takui, Bayesian phase difference estimation: A general quantum algorithm for the direct calculation of energy gaps, *Phys. Chem. Chem. Phys.* **23**, 20152 (2021).
- [34] S. Kanno, K. Sugisaki, H. Nakamura, H. Yamauchi, R. Sakuma, T. Kobayashi, Q. Gao, and N. Yamamoto, Tensor-based quantum phase difference estimation for large-scale demonstration, *Proc. Natl. Acad. Sci. USA* **122**, e2425026122 (2025).
- [35] S. Endo, Q. Zhao, Y. Li, S. Benjamin, and X. Yuan, Mitigating algorithmic errors in a Hamiltonian simulation, *Phys. Rev. A* **99**, 012334 (2019).
- [36] N. F. Robertson, B. Pokharel, B. Fuller, E. Switzer, O. Shtanko, M. Amico, A. Byrne, A. D'Urbano, S. Hayes-Shuptar, A. Akhriev, N. Keenan, S. Bravyi, and S. Zhuk, Tensor network enhanced dynamic multiproduct formulas, *PRX Quantum* **6**, 020360 (2025).
- [37] K. Kanno, M. Kohda, R. Imai, S. Koh, K. Mitarai, W. Mizukami, and Y. O. Nakagawa, Quantum-Selected Configuration Interaction: classical diagonalization of Hamiltonians in subspaces selected by quantum computers, *arXiv:2302.11320*.
- [38] J. Robledo-Moreno, M. Motta, H. Haas, A. Javadi-Abhari, P. Jurcevic, W. Kirby, S. Martiel, K. Sharma, S. Sharma, T. Shirakawa, I. Sitdikov, R.-Y. Sun, K. J. Sung, M. Takita, M. C. Tran, S. Yunoki, and A. Mezzacapo, Chemistry beyond the scale of exact diagonalization on a quantum-centric supercomputer, *Sci. Adv.* **11**, eadu9991 (2025).
- [39] T. Sarkar, J. Nebat, D. Weiner, and V. Jain, Suboptimal approximation/identification of transient waveforms from electromagnetic systems by pencil-of-function method, *IEEE Trans. Antennas Propag.* **28**, 928 (1980).

- [40] A. Steffens, P. Rebentrost, I. Marvian, J. Eisert, and S. Lloyd, An efficient quantum algorithm for spectral estimation, *New J. Phys.* **19**, 033005 (2017).
- [41] S. R. White, Density matrix formulation for quantum renormalization groups, *Phys. Rev. Lett.* **69**, 2863 (1992).
- [42] U. Schollwöck, The density-matrix renormalization group in the age of matrix product states, *Ann. Phys. (Amsterdam, Neth.)* **326**, 96 (2011).
- [43] K. Sugisaki, C. Sakai, K. Toyota, K. Sato, D. Shiomi, and T. Takui, Quantum algorithm for full configuration interaction calculations without controlled time evolutions, *J. Phys. Chem. Lett.* **12**, 11085 (2021).
- [44] P. Jordan and E. Wigner, Über das Paulische öquivalenzverbot, *Z. Phys.* **47**, 631 (1928).
- [45] M. Fishman, S. R. White, and E. M. Stoudenmire, The itensor software library for tensor network calculations, *SciPost Phys. Codebases* **4** (2022).
- [46] A. Javadi-Abhari, M. Treinish, K. Krsulich, C. J. Wood, J. Lishman, J. Gacon, S. Martiel, P. D. Nation, L. S. Bishop, A. W. Cross, B. R. Johnson, and J. M. Gambetta, Quantum computing with Qiskit, [arXiv:2405.08810](https://arxiv.org/abs/2405.08810).
- [47] P. S. Mundada, A. Barbosa, S. Maity, Y. Wang, T. Merkh, T. M. Stace, F. Nielson, A. R. R. Carvalho, M. Hush, M. J. Biercuk, and Y. Baum, Experimental benchmarking of an automated deterministic error-suppression workflow for quantum algorithms, *Phys. Rev. Appl.* **20**, 024034 (2023).
- [48] Y. Baum, M. Amico, S. Howell, M. Hush, M. Liuzzi, P. Mundada, T. Merkh, A. R. R. Carvalho, and M. J. Biercuk, Experimental deep reinforcement learning for error-robust gate-set design on a superconducting quantum computer, *PRX Quantum* **2**, 040324 (2021).
- [49] I. D. Kivlichan, C. Gidney, D. W. Berry, N. Wiebe, J. McClean, W. Sun, Z. Jiang, N. Rubin, A. Fowler, A. Aspuru-Guzik, H. Neven, and R. Babbush, Improved fault-tolerant quantum simulation of condensed-phase correlated electrons via trotterization, *Quantum* **4**, 296 (2020).
- [50] TS-TQPDE_open, https://github.com/sk888ks/TS-TQPDE_open.
- [51] M. Ganahl, J. Beall, M. Hauru, A. G. M. Lewis, T. Wojno, J. H. Yoo, Y. Zou, and G. Vidal, Density matrix renormalization group with tensor processing units, *PRX Quantum* **4**, 010317 (2023).
- [52] J. D. Watson and J. Watkins, Exponentially reduced circuit depths using trotter error mitigation, *PRX Quantum* **6**, 030325 (2025).
- [53] K. Mizuta, On the commutator scaling in Hamiltonian simulation with multi-product formulas, *Quantum* **10**, 1974 (2026).
- [54] J. Haah, M. B. Hastings, R. Kothari, and G. H. Low, Quantum algorithm for simulating real time evolution of lattice Hamiltonians, *SIAM J. Comput.* **52**, FOCS18-250 (2021).