

Quantum dynamics simulations beyond the coherence time on noisy intermediate-scale quantum hardware by variational Trotter compression

Noah F. Berthussen,^{1,2,*} Thaís V. Trevisan,^{1,3} Thomas Iadecola^{1,3,†} and Peter P. Orth^{1,3,‡}

¹Ames Laboratory, Ames, Iowa 50011, USA

²Department of Electrical and Computer Engineering, Iowa State University, Ames, Iowa 50011, USA

³Department of Physics and Astronomy, Iowa State University, Ames, Iowa 50011, USA



(Received 5 January 2022; revised 3 March 2022; accepted 12 April 2022; published 4 May 2022)

We demonstrate a postquench dynamics simulation of a Heisenberg model on present-day IBM quantum hardware that extends beyond the coherence time of the device. This is achieved using a hybrid quantum-classical algorithm that propagates a state using Trotter evolution and then performs a classical optimization that effectively compresses the time-evolved state into a variational form. When iterated, this procedure enables simulations to arbitrary times with an error controlled by the compression fidelity and a fixed Trotter step size. We show how to measure the required cost function, the overlap between the time-evolved and variational states, on present-day hardware, making use of several error mitigation methods. In addition to carrying out simulations on real hardware, we investigate the performance and scaling behavior of the algorithm with noiseless and noisy classical simulations. We find the main bottleneck in going to larger system sizes to be the difficulty of carrying out the optimization of the noisy cost function.

DOI: [10.1103/PhysRevResearch.4.023097](https://doi.org/10.1103/PhysRevResearch.4.023097)

I. INTRODUCTION

Simulating quantum dynamics of interacting many-body systems is one of the main potential applications of quantum computing, going back to Feynman's visionary work on simulating physics with quantum computers [1]. For quantum many-body systems evolving under a generic nonintegrable Hamiltonian H , such simulations are exponentially hard on classical devices due to the growth of entanglement during the simulation time [2,3]. Quantum processing units (QPUs), on the other hand, can efficiently simulate quantum dynamics with resources scaling polynomially in the number of particles N and in the simulation time t . Since Lloyd's original proposal [4] to use a first-order Lie-Trotter product formula to decompose the unitary time-evolution operator into elementary gates, several other methods have been discovered that exhibit a more favorable asymptotic scaling of the required number of gates at or near the optimal scaling of $O(Nt)$ [5–12]. Concrete resource estimates, however, show that this often comes at the cost of requiring a large number of ancilla qubits or having a large constant overhead of gates [9]. Therefore product formulas still remain a preferred choice on noisy intermediate-scale

quantum (NISQ) hardware [13] due to their simplicity and competitive performance for physically relevant (e.g., local) Hamiltonians that fulfill additional properties [14].

Despite this progress, direct quantum simulation algorithms face a critical drawback on NISQ QPUs: The limited coherence time of the device imposes an upper bound on the depth of the quantum circuits that can be implemented with high fidelity. This in turn upper-bounds the simulation time t that can be reached before the output is overwhelmed by errors—on current hardware, this timescale is roughly of order 1 in the natural units imposed by the Hamiltonian being simulated [15]. Quantum simulation beyond the coherence time using “fast-forwarding” algorithms [16–22] is possible, but only for nongeneric Hamiltonians, including those that can be mapped to free fermions [23].

Variational quantum algorithms (VQAs) provide a promising route to overcome the coherence-time obstacle for generic Hamiltonians [24,25]. Their starting point is to represent the wave function by a variational *Ansatz* $|\psi(\boldsymbol{\vartheta})\rangle = \prod_{j=1}^{\mathcal{N}} e^{-i\vartheta_j A_j} |\psi_0\rangle$ with $\mathcal{N}$ real parameters $\boldsymbol{\vartheta} = (\vartheta_1, \dots, \vartheta_{\mathcal{N}})$, initial state $|\psi_0\rangle$, and Hermitian generators A_j , which are often chosen to be single Pauli strings or sums of commuting Pauli strings. To simulate quantum dynamics, the parameters ϑ_j are updated in a way that allows the variational state to follow the exact dynamics; different algorithms to perform the update have been proposed in the literature.

In one class of VQAs, one derives an equation of motion for the variational parameters, $\sum_j M_{ij} \dot{\vartheta}_j(t) = V_i$ [26–28], by extremizing the distance between the variational state and the exact time-evolved state at every (infinitesimally small) time step in the evolution. Here, the matrix M_{ij} and vector V_j must be obtained at every time step by performing measurements on the QPU, and the main bottleneck of the algorithm is the

*Present address: Department of Computer Science, University of Maryland, College Park, MD 20742, USA.

[†]iadecola@iastate.edu

[‡]porth@iastate.edu

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

large number of measurements. Since the number of components M_{ij} scales quadratically with the number of variational parameters $\mathcal{N}$, the number of measurements $\mathcal{M}$ scales as $\mathcal{M} \propto \mathcal{N}^2$.

An alternative approach that reduces the number of measurements is to determine the time dependence of the variational parameters $\boldsymbol{\vartheta}(t)$ by optimizing the state overlap fidelity $\mathcal{F} = |\langle \psi(\boldsymbol{\vartheta}) | \psi(t) \rangle|^2$ between the variational state and the exact time-evolved state $|\psi(t)\rangle$ [29–33]. In practice, the exact state can be approximated with high fidelity, for example, using Trotter evolution over a time interval τ sufficiently short to guarantee a desired high accuracy at finite gate depth. Such a “variational Trotter compression” (VTC) approach combines the desirable aspects of Trotter evolution with the shallow gate depth requirements of variational methods. We note that such circuit compression algorithms are not limited to specific models and can be applied to both integrable and nonintegrable systems.

Here, we expand on previous works that explored VTC algorithms on classical computers [29–33] by performing the first implementation and benchmarking of a VTC algorithm on real quantum hardware. We explicitly demonstrate a simulation of quantum quench dynamics in a Heisenberg spin chain beyond the coherence time on the IBM Santiago and Quito QPUs. To achieve this goal, we employ several error mitigation strategies: We combine zero-noise extrapolation (ZNE) with Pauli twirling [26,34] and symmetry-based postselection. At the current levels of noise on the QPU, we find it advantageous to avoid the gradient-based optimization used in the variant of the VTC algorithm described in Ref. [31], and instead choose the non-gradient-based genetic optimization algorithm referred to as the covariance matrix adaptation evolution strategy (CMA-ES) [35]. This has the additional advantage of allowing quantum circuit executions for an entire generation to be submitted in parallel, facilitating rapid communication between the classical and the quantum computer. We compare different ways to compute the overlap fidelity on QPUs and find that a method based on a “double-time contour” circuit that foregoes the use of ancilla qubits and nonlocal SWAP gates [36–38] is most suited for current NISQ hardware. Finally, we demonstrate the scalability of the VTC algorithm both on noisy and noiseless quantum simulators up to system sizes of $M = 6$ and $M = 11$ sites, respectively.

The remainder of the paper is organized as follows: In Sec. II we present the general VTC algorithm and point out differences from previous works [29,31]. We also discuss different quantum circuit implementations of the optimization cost function. Then, in Sec. III, we apply VTC to simulate postquench dynamics in antiferromagnetic Heisenberg chains. In Sec. III A, we introduce the model and describe the variational *Ansatz* we use, and in Sec. III B we benchmark the capability of the *Ansatz* to capture the exact time-evolved state. We then execute VTC on different classical simulators and on real quantum hardware. In Sec. III C, we present results obtained on a state-vector simulator. In Sec. III D, we use an ideal circuit simulator to consider sample noise due to a finite number of quantum measurements, and in Sec. III E, we show results from classical noisy circuit simulations, where the noise model parameters correspond to the IBM Santiago back end. Finally, in Sec. III F, we demonstrate our key result:

a quantum dynamics simulation beyond the qubit coherence time on the real IBM Santiago and Quito devices. We conclude and discuss future research directions in Sec. IV.

II. ALGORITHM

In this section we describe the different parts of the variational Trotter compression (VTC) algorithm.

A. Choice of *Ansatz*

First, one needs to choose a particular way to build the variational *Ansatz* $|\psi(\boldsymbol{\vartheta})\rangle$. The main part of the algorithm is independent of the form of the *Ansatz* but requires that it is able to faithfully represent the time-evolved state up to a desired fidelity. Since the entanglement content of the state increases during time evolution, the complexity of the *Ansatz* and the number of required parameters grows with time t (and also with system size M). We characterize this growth in detail for a specific model in Sec. III. A key insight from previous works is that the growth with t of the number of variational parameters can be much slower (linear versus exponential) than in purely classical algorithms based on matrix product states (MPSs) [29], giving the VTC algorithm a potential quantum advantage.

The variational *Ansatz* can be chosen either to have a fixed form or to be adaptively modified during the computation [28,39,40]. Here, we choose a fixed *Ansatz* that is inspired by the Hamiltonian variational *Ansatz* (HVA) [41,42] and that takes a layered form,

$$|\psi(\boldsymbol{\vartheta})\rangle = U(\boldsymbol{\vartheta}) |\psi_0\rangle = \prod_{l=1}^{\ell} \prod_{i=1}^N e^{-i\vartheta_{l,i} A_i} |\psi_0\rangle. \quad (1)$$

Here, ℓ denotes the number of layers, and N is the number of variational parameters per layer. We choose the Hermitian operators A_i to be sums of commuting Pauli strings that correspond to the terms in the Hamiltonian being simulated. Finally, $|\psi_0\rangle$ is an initial state that can be chosen arbitrarily; in particular, it does not need to correspond to the state of the system at the initial time t_i , which we denote by $|\psi_i\rangle$.

B. Variational Trotter compression

The central part of the algorithm is the variational compression step. The key idea is to accurately propagate the variational state over a short, finite time interval τ using Trotter evolution and then to re-express the resulting state, $U_{\text{Trot}}(\tau) |\psi(\boldsymbol{\vartheta}_t)\rangle$, in variational form via optimization of the overlap cost function

$$\mathcal{C} = |\langle \psi(\boldsymbol{\vartheta}_{t+\tau}) | U_{\text{Trot}}(\tau) |\psi(\boldsymbol{\vartheta}_t) \rangle|^2. \quad (2)$$

The cost function is measured on the QPU and feeds into a classical optimization routine that determines the updated variational parameters as $\hat{\boldsymbol{\vartheta}}_{t+\tau} = \arg \min_{\boldsymbol{\vartheta}_{t+\tau}} \mathcal{C}$. The complete compression step is then iterated until the final time t_f .

The maximal value of τ in a single compression step is determined by the number of Trotter steps n_{max} that can be executed on the noisy QPU, and by the desired Trotter error threshold ε , which is a function of τ/n_{max} [4,43]. There exist different methods to determine the overlap cost function on

a QPU, which will be discussed in more detail below in Sec. II C. The simplest method concatenates the Trotter and the variational *Ansatz* circuits and calculates the probability that the system returns to the initial state, i.e.,

$$\mathcal{C} = |\langle \psi_0 | U^\dagger(\boldsymbol{\vartheta}_{t+\tau}) U_{\text{Trot}}(\tau) U(\boldsymbol{\vartheta}_t) | \psi_0 \rangle|^2. \quad (3)$$

Here, $U_{\text{Trot}}(\tau)$ is a (first-order) Trotter circuit using n steps. We find this straightforward method to be robust under noise and preferable on NISQ hardware with nearest-neighbor connectivity compared with alternative methods that use ancilla qubits or multiple SWAP operations [36–38]. Since the variational and Trotter circuits are applied consecutively in Eq. (3), the depth of the *Ansatz*, which can be expressed as the number of layers ℓ , must be taken into account when choosing the number of Trotter steps n . Loosely speaking, one must have that $2\ell + n < n_{\text{max}}$, where n_{max} is the maximal number of Trotter steps that can be executed within the finite coherence time of the device.

Different methods can be used for the classical optimization of the overlap cost function $\mathcal{C}$. Reference [31] proposes a gradient-based optimizer, where the gradient is measured directly using a quantum circuit, while Ref. [29] performs the optimization purely classically using tensor-network-based methods. Similar, tensor-network-based optimization methods are employed in Ref. [32], which proposes to update the variational parameters for each term in the Trotter expansion. Reference [33] proposes using a state-agnostic cost function that is the operator distance between the variational and the Trotter (or the classically computed exact) unitary operators. Translational symmetry is exploited to find approximate expressions of the time-evolution operator for classically intractable system sizes. Finally, Ref. [30] exploits translational invariance using matrix product state (MPS) techniques. It employs the SWAP test (see below) to compute state overlaps on classical circuit simulators and the Rotosolve algorithm [44] (also called “sequential minimal optimization” [45]) for the optimization. We have compared the performance of gradient- and non-gradient-based methods and find that non-gradient-based methods such as CMA-ES [35] are preferable for noisy cost functions when considering realistic noise levels present on current quantum hardware.

C. Quantum circuit implementation of cost function measurement

As mentioned previously, when implementing the VTC algorithm on a real quantum device, one needs to decide how to evaluate the overlap cost function $\mathcal{C}$ in Eq. (2), since the full wave function is not accessible. We consider two methods to evaluate $\mathcal{C}$ that are physically realizable on current NISQ devices, and we compare their resource scaling. Since the fidelities of two-qubit entangling gates are much lower than those of single-qubit gates on current hardware, we focus on the scaling of the number of two-qubit gates.

1. SWAP-test circuit

One common method for computing the overlap of two quantum states is the SWAP test [36,37,46], whose circuit implementation is shown in Fig. 1. Here, the two crosses denote SWAP operations between pairs of individual qubits in the two

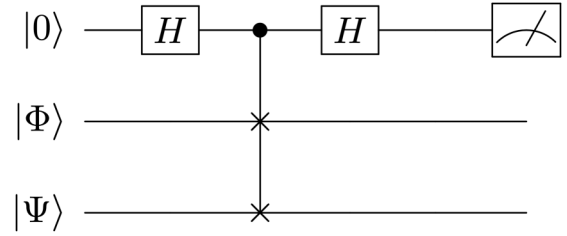


FIG. 1. SWAP-test circuit diagram for computing the overlap between two quantum states $|\Psi\rangle$ and $|\Phi\rangle$ via the expectation value $\langle Z \rangle = |\langle \Psi | \Phi \rangle|^2$ of an ancilla qubit.

quantum registers containing the states $|\Phi\rangle$ and $|\Psi\rangle$. The SWAP test thus requires $2M + 1$ qubits for a system of size M . The ancilla expectation value $\langle Z \rangle = |\langle \Psi | \Phi \rangle|^2$ is equal to the overlap of the states and thus serves as a cost function for VTC. Specifically, we initialize the states $|\Phi\rangle = U_{\text{Trot}}(\tau) U(\boldsymbol{\vartheta}_t) |\psi_0\rangle$ and $|\Psi\rangle = U(\boldsymbol{\vartheta}_{t+\tau}) |\psi_0\rangle$ and find the optimal parameters $\boldsymbol{\vartheta}_{t+\tau}$ by maximizing $\langle Z \rangle$ for the ancilla.

Let us now estimate the number of two-qubit gates required to implement this circuit on two quantum registers of length M , each corresponding to a system of M qubits. Swapping arbitrary M -qubit states $|\Phi\rangle$ and $|\Psi\rangle$ requires M controlled-SWAP gate operations, each of which can be broken down into two controlled-NOT (CNOT) gates and one Toffoli gate. The Toffoli gate can be further decomposed into a circuit containing six CNOTs. Therefore the SWAP test incurs a cost of $8M$ CNOTs in total. One major drawback of using the SWAP test on QPUs with local qubit connectivity, however, is that the ancilla must be coupled to every other qubit during the cascading controlled-SWAPs. As the current generation of superconducting QPUs lacks such nonlocal connectivities, one would need to compile a nonlocal SWAP into many local CNOTs, which significantly increases the resource cost of the SWAP test. Specifically, assuming a linear chain of qubits like the IBM QPU Santiago we use here (see Sec. III F) and placing the ancilla qubit in the middle of the chain, the SWAP gates are between qubit pairs that are separated by a distance $1, 2, 3, \dots$ from the ancilla. This requires $\sum_{n=0}^{M-1} 2n = M(M-1)$ local nearest-neighbor SWAP gates, each of which decomposes into three nearest-neighbor CNOTs, resulting in $\sum_{n=0}^{M-1} 6n = 3M(M-1)$ additional CNOT gates that add to the $8M$ CNOTs that occur for a local SWAP test. The total number of CNOTs in the SWAP-test circuit for a linear qubit chain layout is therefore $M(3M+5)$ CNOTs.

2. Double-time contour circuit

The simplest way to evaluate the overlap cost function, which does not require any qubit overhead or ancillae, is to implement a “double-time contour” circuit, corresponding to the direct implementation of the overlap matrix element in Fig. 2. The key idea is that the inverse of the updated variational circuit $U^\dagger(\boldsymbol{\vartheta}_{t+\tau})$ effectively unwinds the evolution induced by $U_{\text{Trot}}(\tau) U(\boldsymbol{\vartheta}_t)$. The probability $p_{|\psi_0\rangle}$ for the system to end up in its initial state $|\psi_0\rangle$ is maximal for the optimal parameters $\boldsymbol{\vartheta}_{t+\tau}$. We find this cost function to be robust to noise under realistic conditions. The circuit is shown in Fig. 2 and consists of a consecutive application of three unitary

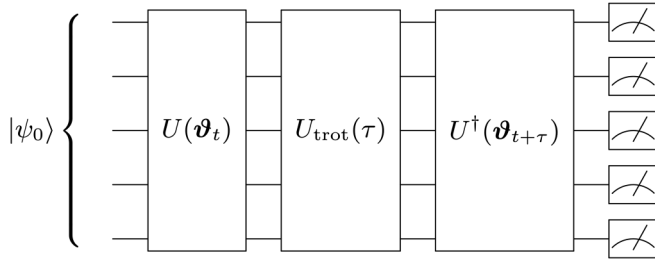


FIG. 2. Double-time contour quantum circuit for ancilla-free evaluation of the overlap cost function $\mathcal{C}$ in Eq. (3). The circuit is shown for $M = 5$.

circuits onto a fixed initial state $|\psi_0\rangle$. For convenience, we choose the initial state $|\psi_0\rangle$ to be a Z-basis state. This method to evaluate $\mathcal{C}$ does not require any additional overhead, and the number of required qubits is equal to M , the number of qubits in the system. Note, however, that the depth of the circuit $U^\dagger(\boldsymbol{\vartheta}_{t+\tau})U_{\text{Trot}}(\tau)U(\boldsymbol{\vartheta}_t)$ is about 1.5 times larger than that required for the SWAP test.

III. APPLICATION TO HEISENBERG SPIN CHAIN DYNAMICS

In this section, we apply the variational Trotter compression algorithm to investigate postquench dynamics in antiferromagnetic Heisenberg chains. We first describe the variational *Ansatz* we employ and then present results using a state-vector simulator, ideal and noisy circuit simulators, and real quantum hardware.

A. Model Hamiltonians and variational *Ansatz*

To benchmark the VTC algorithm and to study its robustness with respect to noise, we apply it to investigate quantum quench dynamics in integrable and nonintegrable Heisenberg spin chains. The Hamiltonian of the pure Heisenberg chain, which is integrable, is given by

$$H_0 = \frac{J}{4} \sum_{i=1}^M (X_i X_{i+1} + Y_i Y_{i+1} + Z_i Z_{i+1}). \quad (4)$$

Here, X_i, Y_i, Z_i are Pauli operators at site i , and M denotes the total number of sites of the chain. For concreteness, we will focus in the following on the antiferromagnetic model with $J > 0$ and consider the quantum spin dynamics that arises when initially preparing the system in the classical Néel ground state $|\psi_i\rangle = |0101\dots\rangle$. The classical Néel state is not an eigenstate of H_0 , leading to nontrivial dynamics of $|\psi(t)\rangle = e^{-iH_0 t} |\psi_i\rangle$. To study the difference between simulations of integrable versus nonintegrable dynamics, we also consider an integrability-breaking next-neighbor interaction in the model,

$$H_1 = H_0 + \frac{J}{4} \sum_{i=1}^M Z_i Z_{i+2}. \quad (5)$$

Unless explicitly stated otherwise, we employ periodic boundary conditions in the following.

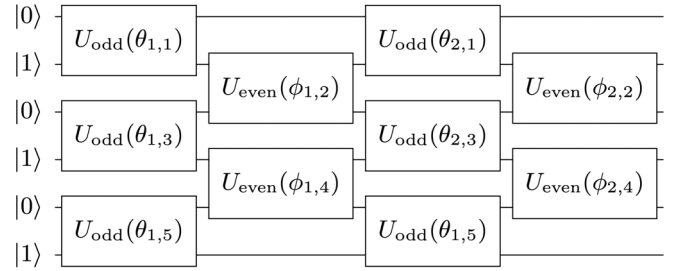


FIG. 3. Quantum circuit implementing the brick-wall *Ansatz* $|\psi(\boldsymbol{\vartheta}^{(\ell)})\rangle$ in Eq. (6) for $\ell = 2$ and $M = 6$. For simplicity, we show the circuit for open boundary conditions (OBCs). For periodic boundary conditions (PBCs), there exists an additional even layer gate $U_{\text{even}}(\phi_{l,M})$ between the first and the last qubit.

Let us now discuss our choice of variational *Ansatz*. For the integrable case, we consider a brick-wall-type quantum circuit with ℓ layers (shown in Fig. 3) that we apply to an initial product state $|\psi_0\rangle = |0101\dots\rangle$:

$$|\psi(\boldsymbol{\vartheta}^{(\ell)})\rangle = \prod_{l=1}^{\ell} U_{\text{even}}(\boldsymbol{\phi}_l) U_{\text{odd}}(\boldsymbol{\theta}_l) |\psi_0\rangle. \quad (6)$$

Here, $\boldsymbol{\vartheta}^{(\ell)} \equiv (\boldsymbol{\theta}_1, \boldsymbol{\phi}_1, \dots, \boldsymbol{\theta}_\ell, \boldsymbol{\phi}_\ell)$ are $M\ell$ variational parameters, and

$$U_{\text{odd}}(\boldsymbol{\theta}_l) = \prod_{j \text{ odd}} e^{-i\theta_{l,j} (X_j X_{j+1} + Y_j Y_{j+1} + Z_j Z_{j+1})}, \quad (7a)$$

$$U_{\text{even}}(\boldsymbol{\phi}_l) = \prod_{j \text{ even}} e^{-i\phi_{l,j} (X_j X_{j+1} + Y_j Y_{j+1} + Z_j Z_{j+1})} \quad (7b)$$

are the unitary operators acting on the odd and the even bonds of the chain, respectively. Here, $\theta_{l,j}$ is the j th entry of the parameter vector $\boldsymbol{\theta}_l$, and $\phi_{l,j}$ is the j th entry of $\boldsymbol{\phi}_l$. For M even, j odd runs over $1, 3, \dots, M-1$. For M odd, however, there exists a boundary term $U_{\text{boundary}}(\theta_{l,M}) = e^{-i\theta_{l,M} (X_M X_1 + Y_M Y_1 + Z_M Z_1)}$ that does not commute with every term in U_{odd} or U_{even} and needs to be included separately in the circuit. The quantum circuit has depth 2ℓ , corresponding to the number of unitaries applied to every qubit. Note that the length of the vector $(\boldsymbol{\theta}_l, \boldsymbol{\phi}_l)$ is equal to the number of sites M and $\boldsymbol{\vartheta}^{(\ell)}$ has $M\ell$ components. A compact quantum circuit representation of the unitary operators in Eqs. (7a) and (7b) has been given in Ref. [47] and is shown in Fig. 4.

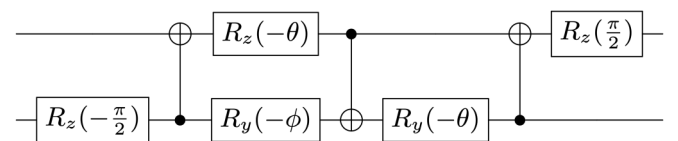


FIG. 4. Quantum circuit implementing the unitary operator $U(\alpha) = \exp[-i\alpha(X_j X_{j+1} + Y_j Y_{j+1} + Z_j Z_{j+1})]$, where $\theta = \frac{\pi}{2} - 2\alpha$ and $\phi = 2\alpha - \frac{\pi}{2}$.

For the nonintegrable model H_1 , we add an additional unitary to each brick-wall layer l :

$$U_Z(\gamma_l) = \prod_{j=1}^M e^{-i\gamma_{l,j} Z_j Z_{j+2}}. \quad (8)$$

This doubles the number of parameters in the *Ansatz*, which is given by $2M\ell$ in the nonintegrable case.

B. Required number of layers ℓ

To benchmark the ability of the variational *Ansatz* in Eq. (6) to represent the exact wave function within a desired accuracy ϵ , we determine the minimal number of layers $\ell_{\min}(t, \epsilon, M)$ needed such that the variational *Ansatz* can represent the exact time-evolved state $|\psi(t)\rangle$ at time t up to an infidelity $1 - \mathcal{F} < \epsilon$. To do this, for each t we numerically minimize the infidelity

$$1 - \mathcal{F}(t, \hat{\vartheta}^{(\ell)}) = 1 - |\langle \psi(\hat{\vartheta}^{(\ell)}) | \psi(t) \rangle|^2 \quad (9)$$

over the parameters $\hat{\vartheta}^{(\ell)}$ to obtain the optimal parameters $\hat{\vartheta}^{(\ell)}$. Repeating this for different values of ℓ , we find $\ell_{\min}(t, \epsilon, M)$ as the smallest ℓ for which the minimal infidelity $1 - \mathcal{F}(t, \hat{\vartheta}^{(\ell)})$ falls below the desired error threshold ϵ . Here, we obtain the exact state $|\psi(t)\rangle = e^{-iHt} |\psi_i\rangle$ via exact diagonalization with $H = H_0$ (or $H = H_1$) and initial state $|\psi_i\rangle = |0101 \dots\rangle$. The optimization is performed using the gradient-based optimizer referred to as the box-constraint variant of the limited-memory Broyden-Fletcher-Goldfarb-Shanno algorithm (L-BFGS-B) [48].

We first study the time dependence of the minimal infidelity for a fixed number of layers ℓ . As shown in Fig. 5, $1 - \mathcal{F}(t, \hat{\vartheta}^{(\ell)})$ first increases rapidly after the quench (unless ℓ is sufficiently large) and then saturates at a value that decreases with increasing layer number. This can be understood by comparison to the behavior of the entanglement entropy when tracing out half the system's degrees of freedom [see inset of Fig. 5(b)]: It grows linearly over time until it reaches saturation due to the finite system size. Since each layer of the brick-wall quantum circuit couples only nearest-neighbor qubits, the spread of entanglement across the system in the variational state is limited by the total number of layers ℓ . As a result, for a fixed layer number ℓ , the infidelity $1 - \mathcal{F}$ grows as a function of time as the variational *Ansatz* is unable to capture the entanglement that builds up in the system during time evolution.

In Fig. 5(b), we show the growth with time of the layer number $\ell^*(t)$ required to keep the infidelity at time t below a fixed threshold that we set to $\epsilon = 1 \times 10^{-4}$ [see also dashed line in Fig. 5(a)]. Like the entanglement entropy, the required layer number grows linearly in time before it reaches saturation. This linear growth in the number of layers and variational parameters provides an opportunity for quantum advantage, since one generically expects state-of-the-art classical techniques based on matrix product techniques to exhibit an exponential scaling with time of the number of parameters [2,29]. Finally, due to finite system size, the required layer number saturates at long times. We denote the saturation value of the layer number by $\ell^* \equiv \lim_{t \rightarrow \infty} \ell^*(t, \epsilon, M)$. In practice, we choose a time t_f much larger than the saturation timescale

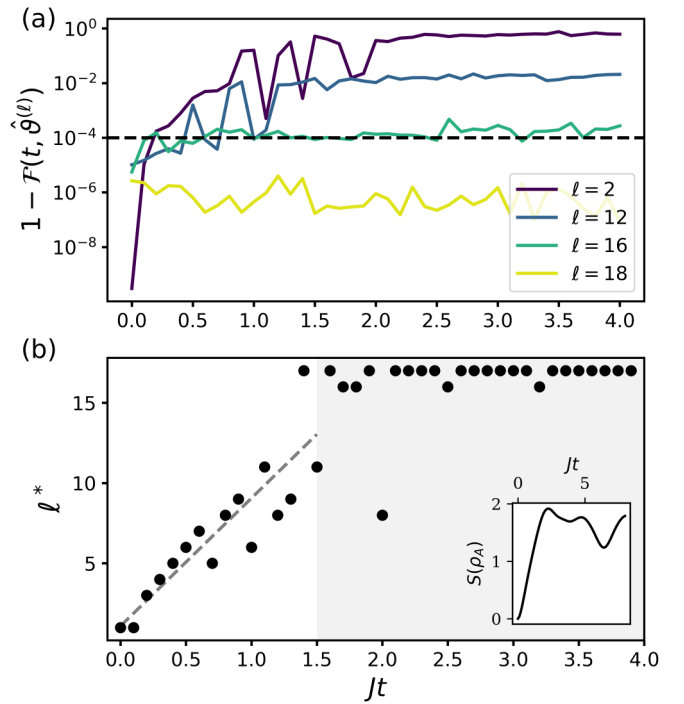


FIG. 5. (a) Minimal infidelity $1 - \mathcal{F}(t, \hat{\vartheta}^{(\ell)})$ as a function of time t for different values of ℓ . Results are for the $M = 8$ pure Heisenberg model H_0 , Eq. (4), using PBCs. We obtain the infidelity by performing a numerical optimization of Eq. (9) every $0.1Jt$. (b) Required layer number $\ell^*(t) \equiv \ell_{\min}(t, \epsilon = 1 \times 10^{-4}, M = 8)$ as a function of time. The inset shows the dynamics of the half-chain entanglement entropy for comparison.

and define $\ell^* \equiv \ell^*(t_f)$. Note that an *Ansatz* with ℓ^* layers is able to represent the time-evolved state out to arbitrary times [see also Fig. 5(a)].

We now systematically study the dependence of the saturation layer number ℓ^* on system size M and desired error threshold ϵ . We therefore fix a final time $t_f = 50J^{-1}$, which is larger than the saturation timescale of the entanglement entropy for all system sizes we consider. In Fig. 6 we show the minimal infidelity, $1 - \mathcal{F}(t_f, \hat{\vartheta}^{(\ell)})$, as a function of layer number ℓ . We present results for the integrable Heisenberg model H_0 in Fig. 6(a) and for the nonintegrable model H_1 in Fig. 6(b). As a function of ℓ , the infidelity curves are first relatively flat until a characteristic M -dependent value, where they start to plunge to much smaller values. For definiteness, we set the desired error threshold to $\epsilon = 5 \times 10^{-3}$, which is indicated by the black dashed line in Fig. 6. The insets show the resulting $\ell^*(\epsilon = 5 \times 10^{-3}, M)$ as a function of system size M . We find that ℓ^* grows exponentially with M , which implies an exponential growth of the number of variational parameters $\mathcal{N} \propto M\ell$ that are needed to describe the long-time dynamics. For $M = 10$ we find $\ell^*(M = 10) \approx 40$ in the integrable model and $\ell^*(M = 10) \approx 20$ in the nonintegrable one. In both cases, the number of variational parameters $\mathcal{N} \approx 350$, since the nonintegrable *Ansatz* has twice as many parameters per layer as the integrable one. This exponential scaling behavior at large times constitutes a bottleneck of the approach when considering larger system sizes. Note, however, that at short

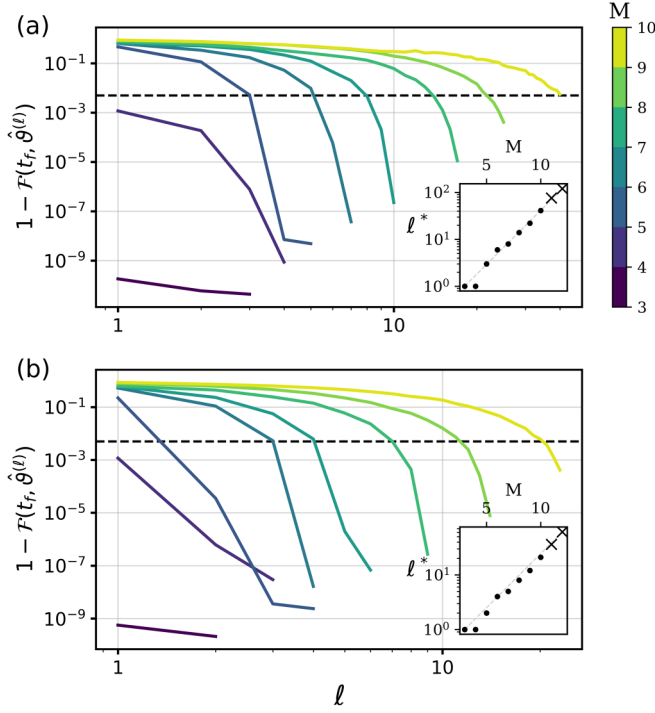


FIG. 6. Minimal infidelity $1 - \mathcal{F}(t_f, \hat{\theta}^{(\ell)})$ at late time $t_f = 50J^{-1}$ as a function of layer number ℓ . Different curves are for different system sizes M . (a) is for the pure Heisenberg model H_0 , Eq. (4), and (b) is for the nonintegrable model H_1 , Eq. (5). Both results are for PBCs. The black dashed line shows a fixed error threshold $\epsilon = 5 \times 10^{-3}$, which defines ℓ^* . The insets show that ℓ^* exhibits exponential scaling with system size M . Since we find that $1 - \mathcal{F}$ decays exponentially with increasing ℓ , the values of ℓ^* denoted by a cross are obtained by exponential extrapolation of $1 - \mathcal{F}(\ell)$.

times we find a favorable linear scaling of ℓ and $\mathcal{N}$ with time t , which provides an opportunity for quantum advantage. Such behavior was also reported previously using a sequential quantum circuit *Ansatz* [29].

C. State-vector simulator results

We now discuss the performance of the VTC algorithm in simulating postquench dynamics of the antiferromagnetic Heisenberg chain. In the following we focus on the integrable model H_0 , since the results of Fig. 6 indicate that the algorithm's performance will not differ substantially between the two cases. In this section, we discuss results obtained using an exact state-vector simulator, where we have direct access to the overlap cost function $\mathcal{C}$ in Eq. (2). Recall that the VTC algorithm consists of two steps: a propagation step, where we use a first-order Trotter product formula to evolve the state from time t to $t + \tau$, and a compression step, where the time-evolved state is compressed into variational form by numerical optimization of the overlap cost function $\mathcal{C}$.

In Fig. 7, we present results for the state overlap fidelity of the VTC-evolved state with the exact state:

$$\mathcal{F}(t, \hat{\theta}_t) = |\langle \psi(t) | \psi(\hat{\theta}_t) \rangle|^2, \quad (10)$$

where $|\psi(t)\rangle = e^{-iH_0 t} |\psi_i\rangle$ and $\hat{\theta}_t$ are the optimal variational parameters maximizing the cost function $\mathcal{C}$ in Eq. (2) at time

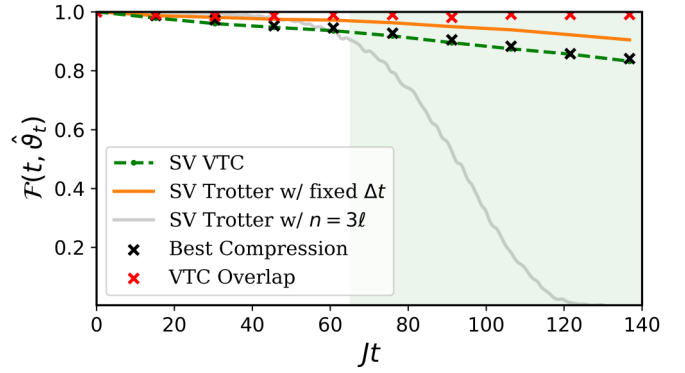


FIG. 7. State-vector (SV) simulator results for the VTC state overlap fidelity $\mathcal{F} = |\langle \psi(t) | \psi(\hat{\theta}_t) \rangle|^2$ (green dashed curve) with the exact state $|\psi(t)\rangle$ during postquench dynamics in the $M = 11$ pure Heisenberg chain H_0 , Eq. (4). The system is initially prepared in the classical Néel state $|\psi_i\rangle$. We use $\ell = 76$ layers in the variational *Ansatz* and $n = 76$ Trotter steps in the VTC propagation step from $t \rightarrow t + \tau$ with $\tau = 15.2J^{-1}$. The error threshold during compression is set to $\epsilon = 5 \times 10^{-3}$, and we employ the gradient-based optimizer L-BFGS-B. The orange curve depicts the Trotter simulation fidelity $\mathcal{F}_{\text{Trot}, \Delta t}$ with fixed step size $\Delta t = \tau/n = 0.2J^{-1}$, corresponding to VTC with perfect compression. The gray curve denotes fidelity from Trotter simulations with fixed number of steps $n = 3\ell = 228$ such that the circuit depth is equal to the double-time contour circuit used as the VTC cost function. In the region highlighted in green, $t > 60J^{-1}$, the state fidelity is larger for VTC than for the fixed-step Trotter simulation, showing an advantage at larger times. The black crosses denote the best compression fidelity for a given VTC propagation step [see Eq. (11)], and the red crosses denote the VTC overlap [see Eq. (12)].

t . Note that in this and the following sections, we drop the (ℓ) superscript when referring to the variational parameters. Instead, we always use $\ell^* \equiv \lim_{t \rightarrow \infty} \ell^*(t, \epsilon, M)$ for a specified ϵ and M .

In Fig. 7, we show results of a VTC simulation for a system of size $M = 11$ over a long time $t_f = 140J^{-1}$ using a variational *Ansatz* with $\ell = 76$ layers that is able to represent the exact time-evolved state over the full time interval. The VTC fidelity $\mathcal{F}(t, \hat{\theta}_t)$ is shown as a green dashed curve in Fig. 7. We find a sizable fidelity $\mathcal{F}(t_f) = 0.83$ at the end of the simulation after performing nine propagation and compression steps at times $t = m\tau$ with $\tau = 15.2$ and $m = 1, \dots, 9$. Here, we have chosen $n = \ell$ Trotter steps to propagate the state from time t to $t + \tau$. If we were to measure the cost function $\mathcal{C}$ using the double-time contour circuit in Fig. 2, the corresponding quantum circuit would contain $N_{\text{CNOT}} = 9M\ell = 7524$ CNOT gates, as a single brick-wall layer contains $3M$ CNOT gates (see Figs. 3 and 4).

This value of $\mathcal{F}(t_f)$ should be contrasted with the fidelity obtained by direct Trotter simulation using the same circuit depth as the VTC circuits: The Trotter simulation fidelity $\mathcal{F}_{\text{Trot}, n=3\ell}(t) = |\langle \psi_{\text{Trot}}(t) | \psi(t) \rangle|^2$ for a fixed number of Trotter steps $n = 3\ell$ falls to zero already at time $t = 120J^{-1}$ (gray curve). We choose 3ℓ Trotter steps because the double-time contour circuit used to measure the VTC cost function $\mathcal{C}$ is of length 3ℓ if one uses $n = \ell$ during the VTC propagation step

[see Eq. (3)]. The Trotter circuit with $n = 3\ell$ thus contains $N_{\text{CNOT}} = 9M\ell = 7524$ CNOT gates as well.

In Fig. 7, we also include the Trotter simulation fidelity $\mathcal{F}_{\text{Trot}, \Delta t}(t)$ for fixed step size $\Delta t = \tau/n = 0.2J^{-1}$ (orange curve). Then, the only source of error is the finite Trotter step size Δt . The Trotter circuit depth thus grows with time, and at the final time $t_f = 140J^{-1}$ it contains $N_{\text{CNOT}} = 3M \frac{t_f}{\Delta t} = 23\,100$ CNOT gates, i.e., about three times as many as the VTC cost function circuit. The chosen fixed Trotter step size $\Delta t = \tau/n = 0.2J^{-1}$ equals the one used in the VTC propagation step, and the deviation of the VTC result (green curve) from the Trotter with fixed Δt (orange) is due to imperfect compression.

It is worth emphasizing that during the VTC algorithm there are two independent sources of error that occur during the propagation and the compression step, respectively. First, a Trotter error occurs during state propagation that is controlled by the Trotter step size τ/n . Second, a compression error arises that is given by the final (i.e., minimal) value of $1 - \mathcal{C}$ at the end of the numerical optimization. This error is controlled by the optimization parameter ϵ that sets the threshold for convergence. Of course, a smaller value of ϵ makes the optimization more difficult and time-consuming. In the ideal case of perfect compression, the Trotter error could be brought down arbitrarily by reducing τ . However, since we have to perform more compression steps for smaller τ , this will increase the compression error in practice. This leads to our choice of parameters for the VTC simulation in Fig. 7, where the two errors are comparable.

To further quantify the performance of the VTC algorithm, we include in Fig. 7 also a “best compression” target fidelity for a single VTC propagation step (black crosses). This is defined as the overlap of the perfectly compressed state after one VTC propagation step, using a Trotter circuit with $n = \ell$ steps, with the exact state:

$$\text{best compression} \equiv |\langle \psi(t) | U_{\text{Trot}}(n = \ell) | \psi(\hat{\boldsymbol{\theta}}_{t-\tau}) \rangle|^2. \quad (11)$$

Here, $|\psi(t)\rangle = e^{-iH_0 t} |\psi_i\rangle$ is the exact state. In other words, after Trotter-evolving the compressed state obtained at the previous time $|\psi(\hat{\boldsymbol{\theta}}_{t-\tau})\rangle$, the black cross denotes the overlap of the resulting state with the exact state. This can be regarded as the target for the VTC algorithm at a given time step, and we observe in Fig. 7 that the VTC fidelity closely follows the black crosses during the evolution. Finally, in Fig. 7, we also include as red crosses the value of the overlap between the best possible compression state and the state to which the algorithm converged during the compression optimization step:

$$\text{VTC overlap} \equiv |\langle \psi(\hat{\boldsymbol{\theta}}_t) | U_{\text{Trot}} | \psi(\hat{\boldsymbol{\theta}}_{t-\tau}) \rangle|^2. \quad (12)$$

We note that the VTC overlap is equal to the cost function $\mathcal{C}$ at the end of the optimization. In Fig. 7, we find an average VTC overlap of 0.990 over all compressions, which means that the average error during compression was about twice as large as the desired error threshold $\epsilon = 0.005$.

The main bottleneck of the algorithm in going to larger system sizes is the time needed to perform the compression optimization. Due to the large number of parameters, e.g., $\mathcal{N} = M\ell = 836$ for $M = 11$ and $\ell = 76$, the optimization of the variational *Ansatz* to produce a single compression point

in Fig. 7 takes a few hours of CPU time, preventing long-time simulations for larger systems. On the other hand, if one limits the simulation to shorter times, one can reduce the number of layers ℓ in the variational *Ansatz* while keeping the product $\mathcal{N} = M\ell$ constant, and thereby simulate larger systems. This was demonstrated for a mixed-field Ising model using a different classical optimization method based on tensor network techniques in Ref. [29].

D. Ideal quantum circuit simulator results

Next, we investigate the performance of the VTC algorithm on a noiseless quantum circuit simulator, where we evaluate the overlap cost function using either the SWAP-test (Fig. 1) or the double-time contour circuit (Fig. 2) using a finite number of quantum measurements, or “samples.” The fact that the overlap cost function now exhibits sample noise increases the difficulty of the numerical optimization during the compression step. In particular, we find that the non-gradient-based optimizer CMA-ES is more reliable and converges much faster to a minimum of the noisy overlap cost function compared with a gradient-based optimizer such as L-BFGS-B. While the sample noise can in principle be made arbitrarily small by increasing the number of samples, we here focus on a realistic number of samples, between 2^{14} and 2^{16} , that could be executed on current NISQ hardware.

The presence of sample noise limits the largest system size we are able to simulate using VTC to $M = 6$. We do not observe any significant difference in the performance of the algorithm depending on whether we use the SWAP-test or the double-time contour circuit to measure the cost function. As shown in Fig. 8(a), the fidelity for the VTC simulation is consistently larger than that for the noiseless direct Trotter simulation at large times. As expected, the VTC fidelity increases with the number of circuit samples as the sample noise is reduced. Specifically, we find a fidelity larger than 0.88 at times $t > 25J^{-1}$, where the fidelity using direct Trotter simulation with 3ℓ steps has already fallen to zero due to the accumulation of Trotter error. Note that we use a state-vector simulator for the direct Trotter calculations.

To further analyze the algorithm’s performance for the maximal number (2^{16}) of samples, we include in Fig. 8(b) the fidelity (with respect to the exact time-evolved state) of the best possible compression during the next propagation period τ . Explicitly, this is given by the overlap $|\langle \psi(t) | U_{\text{Trot}} | \psi(\hat{\boldsymbol{\theta}}_{t-\tau}) \rangle|^2$; the data are represented as black crosses in the figure. The red crosses indicate the overlap of the best possible compression state, $U_{\text{Trot}} | \psi(\hat{\boldsymbol{\theta}}_{t-\tau}) \rangle$, with the state to which the algorithm converged; explicitly, this is given by $|\langle \psi(\hat{\boldsymbol{\theta}}_t) | U_{\text{Trot}} | \psi(\hat{\boldsymbol{\theta}}_{t-\tau}) \rangle|^2$. Note that this is equal to the value of the cost function $\mathcal{C}$ at the end of the optimization. An interesting feature relative to the noiseless simulation considered in Fig. 7 is that the VTC algorithm can converge to a state whose fidelity with the exact time-evolved state exceeds that of the “best compression” state $U_{\text{Trot}} | \psi(\hat{\boldsymbol{\theta}}_{t-\tau}) \rangle$ —this is visible in the last data point in Fig. 8(b). This feature is a result of the sample noise impacting the optimization during the compression step. Indeed, for the noiseless state-vector simulation, the actual fidelity of the converged VTC state is upper bounded by the “best compression” value, as expected.

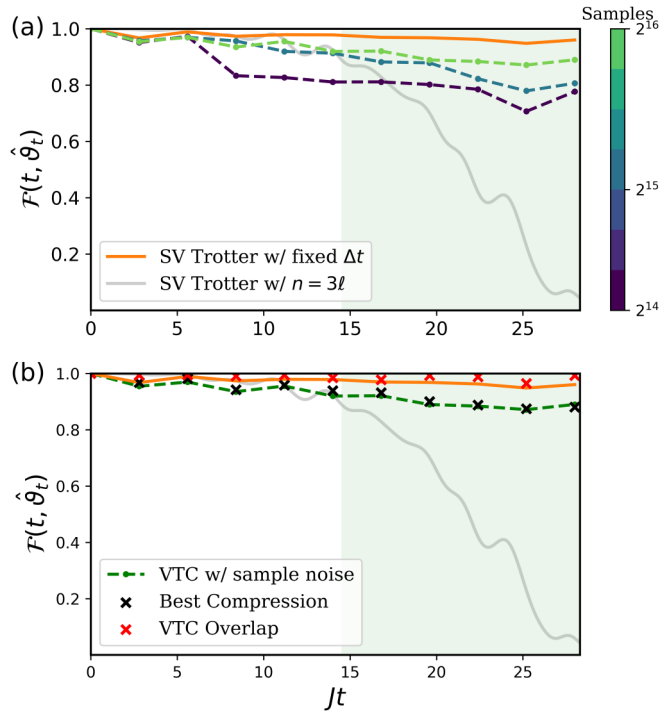


FIG. 8. Optimal fidelity $\mathcal{F}(t, \hat{\theta}_t)$ for VTC simulations on an ideal quantum circuit simulator. (a) Dependence of the fidelity on the number of circuit samples. Dashed curves of different colors correspond to executions of the algorithm using different numbers of quantum measurements (samples) for each cost function evaluation, as indicated by the color bar. Results are for $M = 6$, $\ell = n = 7$, $\tau = 2.8J^{-1}$, and compression error threshold $\epsilon = 5 \times 10^{-3}$. We use CMA-ES for the compression optimization. For comparison, we plot the results of noiseless state-vector (SV) Trotter simulations using $3\ell = 21$ Trotter steps (gray) and using a variable number of Trotter steps with fixed $\Delta t = \tau/n = 0.4J^{-1}$ (orange). The orange curve corresponds to the best possible VTC result using these parameters. (b) Analysis of the fidelity data for 2^{16} samples in the vein of Fig. 7. The black crosses denote the best compression fidelity for a given VTC propagation step [see Eq. (11)], and the red crosses denote the VTC overlap [see Eq. (12)]. We note that the shown VTC overlap is computed using state-vector simulators, i.e., it corresponds to the exact value of the cost function $\mathcal{C}$ at the end of the optimization, as opposed to the value provided by the circuit simulator that is affected by shot noise.

A single compression step for the parameters in Fig. 8 takes up to 10 h of CPU time. The difficulty of the noisy optimization during the compression step constitutes the main bottleneck of the circuit simulations in going to larger system sizes. This bottleneck could be mitigated by parallelizing the circuit evaluations across different CPUs (or indeed QPUs), which would allow the accumulation of more samples to reduce the sample noise. Another possibility worth exploring is to replace the overlap cost function by, e.g., a reduced density matrix fidelity [49] defined only over a subset of the full system where local quantities of interest are to be computed.

E. Noisy circuit simulator results

We now discuss VTC simulation results on a noisy quantum circuit simulator that takes gate imperfections and finite

qubit coherence times into account. To connect with the simulations on the real IBM QPU Santiago, which will be discussed below, we choose a noise model with parameters drawn from that chip, as implemented in QISKIT AER [50]. Due to the increased noise in the simulations arising from gate errors, readout errors, and finite qubit coherence times T_1 and T_2 , the numerical optimization during the compression step is even more challenging than for the ideal quantum circuit simulator. We use the non-gradient-based method CMA-ES for the classical optimization, which we find to be more reliable than gradient-based approaches.

To reduce the noise in the cost function, it is absolutely essential to exploit a combination of standard and specifically tailored error mitigation techniques. We employ standard readout error mitigation as built into QISKIT IGNIS, and zero-noise extrapolation (ZNE) with a linear fit, which we implement using the software package MITIQ [51]. We also apply a specific postselection protocol to the results: Since the Heisenberg model preserves the magnetization operator $S^z = \sum_{i=1}^M Z_i$, only computational basis states with the same expectation value of S^z as the initial state are physically allowed. We thus discard any counts of computational basis states that do not fulfill the total S^z conservation law after performing readout error mitigation.

In Fig. 9(a) we present VTC results for the postquench dynamics in an $M = 3$ spin chain with open boundary conditions using the noisy quantum circuit simulator. The noise parameters are drawn from the IBM QPU Santiago. We prepare the system initially in the state $|110\rangle$ and time-evolve it with the antiferromagnetic Heisenberg Hamiltonian. We use $\ell = 2$ layers in the variational *Ansatz*, use $n = 2$ Trotter steps during the VTC propagation step, and average over 8192 circuit samples. Since the calculation of each compression point only takes about a minute, we are able to run the noisy simulation 50 times and average over the resulting VTC fidelities (green dashed curve). The error bars on the green dashed curve correspond to the standard deviation over these 50 runs. We observe that the noisy VTC simulations agree within error bars with a Trotter simulation with fixed step size $\Delta t = \tau/n = J^{-1}$ (orange curve). The Trotter simulation with fixed Δt , chosen to be identical to the step size used in VTC, corresponds to noiseless VTC with perfect compression. Importantly, for times larger than $t > 9J^{-1}$ the VTC fidelity is larger than the fidelity of a noiseless direct Trotter simulation with a fixed number of Trotter steps $n = 3\ell = 6$. We chose $n = 3\ell$ as this corresponds to the circuit depth when evaluating the VTC cost function. While the Trotter fidelity drops to zero around $t \approx 16J^{-1}$, the VTC fidelity remains above 0.9 throughout the full simulation until $t = 30J^{-1}$.

F. Results on IBM QPUs

We now demonstrate quantum dynamics simulations beyond the qubit coherence time on real IBM quantum hardware. This is achieved by running VTC simulations on the IBM QPUs Santiago (`ibmq_santiago`) and Quito (`ibmq_quito`) for a Heisenberg model with $M = 3$ spins. As before, we prepare the system initially in the state $|110\rangle$ and time-evolve it with the antiferromagnetic Heisenberg Hamiltonian. The results for the state fidelity overlap with the exact

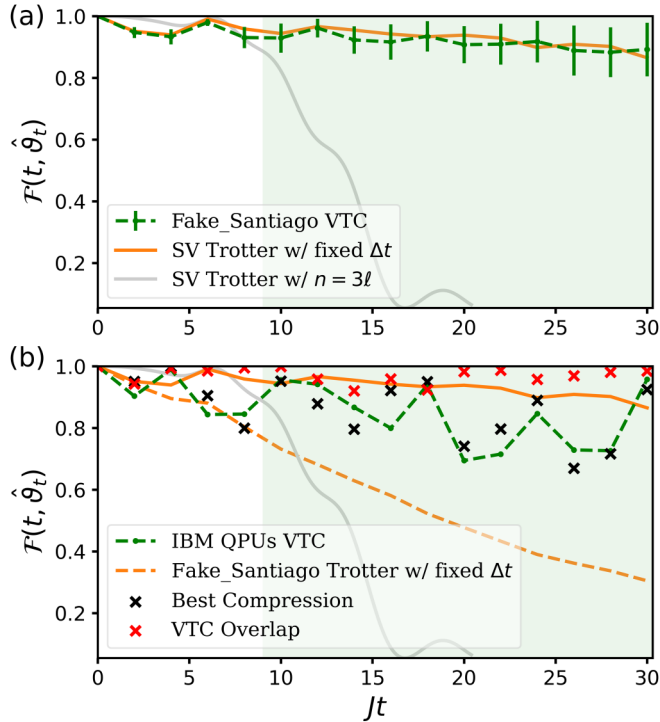


FIG. 9. (a) VTC results on a noisy circuit simulator using error model parameters corresponding to the IBM fake_santiago back end. We use $M = 3$, $\ell = n = 2$, $\tau = 2J^{-1}$, $\epsilon = 5 \times 10^{-3}$, and 2^{13} samples. The numerical optimization is done using CMA-ES. The green dashed curve shows the mean fidelity found after 50 runs of the VTC algorithm with the standard deviation shown as error bars. The orange curve shows results of noiseless Trotter simulation with fixed step size $\Delta t = \tau/n = J^{-1}$, corresponding to noiseless VTC with perfect compression. The gray curve is a noiseless Trotter simulation with fixed number of $n = 3\ell = 6$ steps. The green shading highlights the region where the VTC fidelity exceeds the one from Trotter simulation with fixed n . (b) VTC results on real IBM QPUs ibmq_santiago and ibmq_quito for the same parameters as in (a), except that we used 2×10^4 samples on ibmq_quito. Each data point on the green VTC curve represents a single run of the compression algorithm. The dashed orange curve is obtained by running a Trotter evolution with fixed step size $\Delta t = \tau/n = J^{-1}$ on a noisy simulator of the IBM fake_santiago back end and then performing state tomography with respect to the exact state $|\psi(t)\rangle$. The black crosses denote the best compression fidelity for a given VTC propagation step [see Eq. (11)], and the red crosses denote the VTC overlap [see Eq. (12)]. Note that the VTC overlap is the exact value of the cost function $\mathcal{C}$ at the end of optimization, not the one provided by the noisy QPU.

solution are shown in Fig. 9(b). The key result is that at times $t > 9J^{-1}$ the overlap of the VTC state with the exact state $\mathcal{F}(t, \hat{\theta}_t)$ significantly exceeds the fidelity obtained from noiseless direct Trotter simulations using the same circuit depth (gray curve), i.e., $n = 3\ell$ Trotter steps. The Trotter fidelity quickly decays due to the accumulation of Trotter error. In contrast, in VTC the time-evolved state is compressed at intermediate times, and the VTC simulation can thus be extended out to arbitrarily long times.

We use a variational Ansatz with $\ell = 2$ layers that has in total four variational parameters, since we use open boundary

conditions. We employ $n = 2$ Trotter steps during the state propagation step of VTC and average over $2^{13} = 8192$ circuit samples on ibmq_santiago and over 2×10^4 samples on ibmq_quito. The complete algorithm is executed on real quantum hardware, i.e., all circuits during the compression optimizations are being executed on the QPU, and the obtained fidelity is fed back into the classical optimizer to update the variational parameters. The VTC results (green dashed curve) for $0 \leq Jt \leq 20$ were obtained on ibmq_santiago, and results for $Jt > 20$ were obtained on ibmq_quito. These results were collected over the course of 5 days, since the calculation of each compression point takes about 2–3 h, which includes waiting time in the IBM execution queue.

We choose a subset of three connected qubits on the devices that experiences the smallest average CNOT gate error, which was $\text{Err}_{\text{CNOT, Santiago}} = 6.01 \times 10^{-3}$ and $\text{Err}_{\text{CNOT, Quito}} = 6.30 \times 10^{-3}$, respectively. The average readout errors were $\text{Err}_{\text{readout, Santiago}} = 2.17 \times 10^{-2}$ and $\text{Err}_{\text{readout, Quito}} = 3.05 \times 10^{-2}$, and the relaxation and dephasing timescales were given by $T_1 = 94 \mu\text{s}$, $T_2 = 100 \mu\text{s}$ for ibmq_santiago and $T_1 = 98 \mu\text{s}$, $T_2 = 125 \mu\text{s}$ for ibmq_quito. The specifications of the two devices we use in this simulation are thus comparable. To obtain our results, we applied all error mitigation techniques described in Sec. III E. In addition, we found it essential to use a Pauli twirling technique, which dresses the two-qubit CNOT gates with random Pauli gates [26]. This is important in order to convert the noise on the real device into a stochastic form, which is necessary to justify ZNE. Without the additional Pauli twirling protocol, we found the ZNE error mitigation method to be unreliable.

In Fig. 9(b) we analyze the data from the QPU using the methodology of Figs. 7 and 8. We find that the fidelity of the state to which the compression optimization converges can again exceed that of the best possible compression. This effect, which was also apparent in the results of Fig. 8, is much more pronounced on the real QPU due to the presence of both sample and gate noise. The red crosses again represent the VTC overlap, which is defined as the fidelity between the best possible compression and the state found by the algorithm. We note that the VTC overlap is calculated using a state-vector simulator and thus represents the actual final value of the cost function $\mathcal{C}$ at the end of the optimization, as opposed to the noisy value that was provided by the QPU. Over the 15 compression steps, the algorithm was able to re-create the Trotter evolution of the state from the previous compression step with a mean fidelity of 0.971.

To explicitly demonstrate that we have achieved simulation beyond the device coherence time, we compare with Trotter simulations with fixed $\Delta t = \tau/n = J^{-1}$. In the noiseless state-vector case, Trotter evolution with fixed Δt yields the orange solid curve in Fig. 9(b), which corresponds to VTC with perfect compression. In contrast, executing these Trotter circuits on the fake_santiago back end and performing quantum state tomography in the final state yields the orange dashed curve, which decays over time. At the final time $t_f = 30J^{-1}$ the Trotter fidelity has decayed to a value smaller than 0.4. This is in agreement with a simple estimate of the final state fidelity based on the fact that the Trotter circuit at $t = 30J^{-1}$ contains 180 CNOT gates: $\mathcal{F}(t_f) \approx (1 - \text{Err}_{\text{CNOT, Santiago}})^{180} = 0.34$. Importantly, the VTC fidelity

with the exact state consistently lies above the Trotter results for times $t > 5J^{-1}$ and is equal to $\mathcal{F}[t_f, \hat{\theta}(t_f)] = 0.96$ at the final time $t = 30J^{-1}$. Even though the impact of noise on the real QPU is more severe compared with the noisy simulator, we are able to achieve quantum dynamics simulations beyond the coherence of the device using the VTC algorithm.

Number of measurement circuits and comparison with VQDS

Let us now compare the number of circuits that we had to evaluate to obtain the results shown in Fig. 9, with the number required for a variational quantum dynamics simulation (VQDS) based on MacLachlan's principle [26]. The number of measurement circuits $\mathcal{M}_i$ that are executed in VTC at a single compression step i depends on the number of cost function evaluations $N_{\text{cost}}(i)$ during the classical optimization. The value of $N_{\text{cost}}(i)$ depends on the desired state infidelity ϵ between the compressed variational state and the Trotter-evolved state. Choosing a larger ϵ yields faster convergence of the classical optimization algorithm and thus smaller N_{cost} . This comes at the cost of acquiring a larger error during compression. We choose $\epsilon = 5 \times 10^{-3}$ for which on average $\langle N_{\text{cost}}(i) \rangle_{i=1, \dots, 15} = 380$ for the data in Fig. 9(b). The total simulation until time $t_f = 30J^{-1}$ involved 15 compression steps as $\tau = 2J^{-1}$, and the total number of measurement circuits is thus given by $\mathcal{M}_{\text{tot, VTC}} = 5700$.

This can be compared to the number of circuits that need to be evaluated within the VQDS approach [26,28]. To obtain a meaningful and fair comparison, we consider a fixed *Ansatz* that is identical to the one used in VTC. For the results in Fig. 9, the *Ansatz* contains $\mathcal{N} = (M - 1)\ell = 4$ variational parameters. Within the VQDS method, the variational parameters evolve according to a deterministic equation of motion $\Delta\theta_i = \sum_{j=1}^{\mathcal{N}} (M^{-1})_{ij} V_j \delta t$, where M_{ij} and V_i are obtained by evaluating in total $\mathcal{M}_{\text{VQDS}} = 49$ circuits at each time step δt . It is important to note that the step size in VQDS, δt , must typically be chosen to be much smaller than the VTC Trotter step size $\Delta t = \tau/n$. This originates from the typically large condition number of the matrix M_{ij} , and we find in practice that $\delta t \simeq \Delta t/N_c$ with $1 \lesssim N_c \lesssim 10$ (for details, see Sec. II.C.3 in Ref. [28]). The total number of VQDS circuits that must be evaluated to obtain a comparable final state fidelity as shown in Fig. 9(b) can thus be estimated as $\mathcal{M}_{\text{tot, VQDS}} = \mathcal{M}_{\text{VQDS}} \frac{t_f}{\delta t} = \mathcal{M}_{\text{VQDS}} \frac{N_c t_f n}{\tau} = N_c \times 49 \times 30 \times \frac{2}{\tau} = 1470N_c$. Using the value of $1 \lesssim N_c \lesssim 10$ that was reported in Ref. [28], one finds that VQDS requires evaluating $1500 \lesssim \mathcal{M}_{\text{tot, VQDS}} \lesssim 15000$ circuits. The two approaches, VTC and VQDS, therefore require execution of a comparable number of measurement circuits for the parameters used in Fig. 9.

IV. CONCLUSION

We have demonstrated a simulation of the postquench dynamics in a three-site antiferromagnetic Heisenberg chain beyond the qubit coherence time on real quantum hardware. This was achieved by compressing the Trotter time-evolved state at intermediate time steps into a variational form. The

state overlap served as a cost function for the compression optimization step, which was executed on NISQ hardware without any additional qubit overhead using a double-time contour circuit. The VTC circuits require no ancilla qubits and, for comparison, their depth is chosen to be identical (3ℓ) to a direct first-order Trotter circuit with fixed number of steps, which it clearly outperforms. The method yields an average fidelity of 0.86 with the exact state and maintains a fidelity of 0.80 in a region where a first-order Trotter expansion using the same circuit depth already vanishes.

We show that Trotter simulations with fixed step size, which corresponds to VTC with perfect compression, require the execution of deep circuits that are beyond the coherence time on the device. This is demonstrated using quantum state tomography on a noisy IBM “fake-device” back end. Explicitly, at the final time of the simulation, the Trotter circuit contains 180 CNOT gates, whereas each VTC circuit only contains 36 CNOTs. Since the time-evolved state is compressed at intermediate times, the VTC simulation can be extended out to arbitrarily long times, as long as the time-evolved state can be accurately captured by the variational circuit. Since it requires a noisy cost function optimization at every compression step, the time interval between compression points should be chosen sufficiently large. Finally, we provide an explicit gate count for $M = 3$ and find the total number of circuits that need to be evaluated for VTC to be similar to the other variational approaches such as VQDS based on MacLachlan's principle.

We have further benchmarked the VTC algorithm on larger Heisenberg chains using state-vector as well as noiseless and noisy circuit simulators. While most of our results were obtained for an integrable model, we also tested the ability of a variational *Ansatz* to capture the dynamics generated by the Heisenberg Hamiltonian in the presence of an integrability-breaking term. These tests revealed that the performance of the VTC algorithm does not depend strongly on the presence or absence of integrability—indeed, an *Ansatz* with the same number of parameters captures the wave function to the same accuracy in both cases.

The main goal moving forward is to increase the system sizes accessible to VTC simulations, for which several key issues need to be addressed. First, the noise level of the overlap cost function needs to be reduced in order to accelerate the classical optimization that is performed during the compression step. In addition to reducing device errors by hardware improvements, one can devise several error mitigation strategies [52] such as probabilistic error cancellation [34], virtual distillation [53,54], Clifford data regression [55], or a combination thereof [56–60]. Sample noise can be reduced by increasing the number of circuit shots beyond the current limit of 2^{13} on the IBM back end we used. The fidelity of the Trotter time evolution during the propagation step can be increased by using pulse level control of the gates, which can often lead to shorter gate times. Finally, other future directions would be to explore alternative variational *Ansätze* such as hardware-efficient ones and to employ other classical optimizers for noisy cost functions.

We provide all required programs as open-source software, and we make the raw data of our results openly accessible [61].

ACKNOWLEDGMENTS

The authors acknowledge valuable discussions with M. Sohaib Alam, Andy C. Y. Li, Stephan Rachel, and Yong-

Xin Yao. This material is based upon work supported by the National Science Foundation under Grant No. DMR-2038010.

-
- [1] R. P. Feynman, Simulating physics with computers, *Int. J. Theor. Phys.* **21**, 467 (1982).
 - [2] T. Prosen and M. Žnidarič, Is the efficiency of classical simulations of quantum dynamics related to integrability?, *Phys. Rev. E* **75**, 015202(R) (2007).
 - [3] U. Schollwöck, The density-matrix renormalization group in the age of matrix product states, *Ann. Phys. (Amsterdam)* **326**, 96 (2011).
 - [4] S. Lloyd, Universal quantum simulators, *Science* **273**, 1073 (1996).
 - [5] A. M. Childs and N. Wiebe, Hamiltonian simulation using linear combinations of unitary operations, *Quantum Inf. Comput.* **12**, 901 (2012).
 - [6] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Efficient quantum algorithms for simulating sparse Hamiltonians, *Commun. Math. Phys.* **270**, 359 (2007).
 - [7] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Simulating Hamiltonian Dynamics with a Truncated Taylor Series, *Phys. Rev. Lett.* **114**, 090502 (2015).
 - [8] G. H. Low and I. L. Chuang, Optimal Hamiltonian Simulation by Quantum Signal Processing, *Phys. Rev. Lett.* **118**, 010501 (2017).
 - [9] A. M. Childs, D. Maslov, Y. Nam, N. J. Ross, and Y. Su, Toward the first quantum simulation with quantum speedup, *Proc. Natl. Acad. Sci. USA* **115**, 9456 (2018).
 - [10] A. M. Childs and Y. Su, Nearly Optimal Lattice Simulation by Product Formulas, *Phys. Rev. Lett.* **123**, 050503 (2019).
 - [11] G. H. Low, V. Kliuchnikov, and N. Wiebe, Well-conditioned multiproduct Hamiltonian simulation, *arXiv:1907.11679* [physics, physics:quant-ph].
 - [12] J. Haah, M. B. Hastings, R. Kothari, and G. H. Low, Quantum algorithm for simulating real time evolution of lattice Hamiltonians, *SIAM J. Comput.*, doi:10.1137/18M1231511.
 - [13] J. Preskill, Quantum computing in the NISQ era and beyond, *Quantum* **2**, 79 (2018).
 - [14] M. C. Tran, S.-K. Chu, Y. Su, A. M. Childs, and A. V. Gorshkov, Destructive Error Interference in Product-Formula Lattice Simulation, *Phys. Rev. Lett.* **124**, 220502 (2020).
 - [15] A. Smith, M. S. Kim, F. Pollmann, and J. Knolle, Simulating quantum many-body dynamics on a current digital quantum computer, *npj Quantum Inf.* **5**, 106 (2019).
 - [16] A. Cervera-Lierta, Exact Ising model simulation on a quantum computer, *Quantum* **2**, 114 (2018).
 - [17] C. Cîrstoiu, Z. Holmes, J. Iosue, L. Cincio, P. J. Coles, and A. Sornborger, Variational fast forwarding for quantum simulation beyond the coherence time, *npj Quantum Inf.* **6**, 82 (2020).
 - [18] J. Gibbs, K. Gili, Z. Holmes, B. Commeau, A. Arrasmith, L. Cincio, P. J. Coles, and A. Sornborger, Long-time simulations with high fidelity on quantum hardware, *arXiv:2102.04313* [quant-ph].
 - [19] L. Bassman, R. V. Beeumen, E. Younis, E. Smith, C. Iancu, and W. A. de Jong, Constant-depth circuits for dynamic simulations of materials on quantum computers, *Mater. Theory* **6**, 13 (2022).
 - [20] D. Camps, E. Kökcü, L. Bassman, W. A. de Jong, A. F. Kemper, and R. V. Beeumen, An algebraic quantum circuit compression algorithm for Hamiltonian simulation, *arXiv:2108.03283* [math.NA].
 - [21] E. Kökcü, D. Camps, L. Bassman, J. K. Freericks, W. A. de Jong, R. V. Beeumen, and A. F. Kemper, Algebraic compression of quantum circuits for Hamiltonian evolution, *Phys. Rev. A* **105**, 032420 (2022).
 - [22] S. Gulania, B. Peng, Y. Alexeev, and N. Govind, Quantum time dynamics of 1D-Heisenberg models employing the Yang-Baxter equation for circuit compression, *arXiv:2112.01690* [quant-ph].
 - [23] Y. Atia and D. Aharonov, Fast-forwarding of Hamiltonians and exponentially precise measurements, *Nat. Commun.* **8**, 1572 (2017).
 - [24] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, Variational quantum algorithms, *Nat. Rev. Phys.* **3**, 625 (2021).
 - [25] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W.-K. Mok, S. Sim, L.-C. Kwek, and A. Aspuru-Guzik, Noisy intermediate-scale quantum algorithms, *Rev. Mod. Phys.* **94**, 015004 (2022).
 - [26] Y. Li and S. C. Benjamin, Efficient Variational Quantum Simulator Incorporating Active Error Minimization, *Phys. Rev. X* **7**, 021050 (2017).
 - [27] X. Yuan, S. Endo, Q. Zhao, Y. Li, and S. C. Benjamin, Theory of variational quantum simulation, *Quantum* **3**, 191 (2019).
 - [28] Y.-X. Yao, N. Gomes, F. Zhang, C.-Z. Wang, K.-M. Ho, T. Iadecola, and P. P. Orth, Adaptive Variational Quantum Dynamics Simulations, *PRX Quantum* **2**, 030307 (2021).
 - [29] 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).
 - [30] F. Barratt, J. Dborin, M. Bal, V. Stojevic, F. Pollmann, and A. G. Green, Parallel quantum simulation of large systems on small NISQ computers, *npj Quantum Inf.* **7**, 79 (2021).
 - [31] S. Barison, F. Vicentini, and G. Carleo, An efficient quantum algorithm for the time evolution of parameterized circuits, *Quantum* **5**, 512 (2021).
 - [32] M. Benedetti, M. Fiorentini, and M. Lubasch, Hardware-efficient variational quantum algorithms for time evolution, *Phys. Rev. Research* **3**, 033083 (2021).
 - [33] R. Mansuroglu, T. Eckstein, L. Nützel, S. A. Wilkinson, and M. J. Hartmann, Classical variational optimization of gate sequences for time evolution of translational invariant quantum systems, *arXiv:2106.03680* [quant-ph].
 - [34] K. Temme, S. Bravyi, and J. M. Gambetta, Error Mitigation for Short-Depth Quantum Circuits, *Phys. Rev. Lett.* **119**, 180509 (2017).

- [35] N. Hansen, yoshihikoueno, ARF1, K. Nozawa, M. Chan, Y. Akimoto, and D. Brockhoff, CMA-ES/pycma:r3.1.0 (2021), <https://doi.org/10.5281/zenodo.5002422>.
- [36] H. Buhrman, R. Cleve, J. Watrous, and R. de Wolf, Quantum Fingerprinting, *Phys. Rev. Lett.* **87**, 167902 (2001).
- [37] J. C. Garcia-Escartin and P. Chamorro-Posada, SWAP test and Hong-Ou-Mandel effect are equivalent, *Phys. Rev. A* **87**, 052330 (2013).
- [38] B. F. Schiffer, J. Tura, and J. I. Cirac, Adiabatic spectroscopy and a variational quantum adiabatic algorithm, [arXiv:2103.01226](https://arxiv.org/abs/2103.01226) [quant-ph].
- [39] H. R. Grimsley, S. E. Economou, E. Barnes, and N. J. Mayhall, An adaptive variational algorithm for exact molecular simulations on a quantum computer, *Nat. Commun.* **10**, 3007 (2019).
- [40] H. L. Tang, V. O. Shkolnikov, G. S. Barron, H. R. Grimsley, N. J. Mayhall, E. Barnes, and S. E. Economou, Qubit-ADAPT-VQE: An Adaptive Algorithm for Constructing Hardware-Efficient Ansätze on a Quantum Processor, *PRX Quantum* **2**, 020310 (2021).
- [41] D. Wecker, M. B. Hastings, and M. Troyer, Progress towards practical quantum variational algorithms, *Phys. Rev. A* **92**, 042303 (2015).
- [42] R. Wiersema, C. Zhou, Y. de Sereville, J. F. Carrasquilla, Y. B. Kim, and H. Yuen, Exploring Entanglement and Optimization within the Hamiltonian Variational Ansatz, *PRX Quantum* **1**, 020319 (2020).
- [43] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe, and S. Zhu, Theory of Trotter Error with Commutator Scaling, *Phys. Rev. X* **11**, 011020 (2021).
- [44] M. Ostaszewski, E. Grant, and M. Benedetti, Structure optimization for parameterized quantum circuits, *Quantum* **5**, 391 (2021).
- [45] K. M. Nakanishi, K. Fujii, and S. Todo, Sequential minimal optimization for quantum-classical hybrid algorithms, *Phys. Rev. Research* **2**, 043158 (2020).
- [46] S. Foulds, V. Kendon, and T. Spiller, The controlled SWAP test for determining quantum entanglement, *Quantum Sci. Technol.* **6**, 035002 (2021).
- [47] F. Vatan and C. Williams, Optimal quantum circuits for general two-qubit gates, *Phys. Rev. A* **69**, 032315 (2004).
- [48] F. Gerber, florafauna/optimParallel-python: test Zenodo (2020), <https://doi.org/10.5281/zenodo.3888570>
- [49] A. Bolens and M. Heyl, Reinforcement Learning for Digital Quantum Simulation, *Phys. Rev. Lett.* **127**, 110502 (2021).
- [50] G. Aleksandrowicz, T. Alexander, P. Barkoutsos, L. Bello, Y. Ben-Haim, D. Bucher, F. J. Cabrera-Hernández, J. Carballo-Franquis, A. Chen, C.-F. Chen, J. M. Chow, A. D. Córcoles-Gonzales, A. J. Cross, A. Cross, J. Cruz-Benito, C. Culver, S. De La Puente González, E. De La Torre, D. Ding, E. Dumitrescu *et al.*, Qiskit: An open-source framework for quantum computing (2019), [http://doi.org/10.5281/zenodo.2562111](https://doi.org/10.5281/zenodo.2562111).
- [51] R. LaRose, A. Mari, P. J. Karalekas, N. Shammah, and W. J. Zeng, Mitiq: A software package for error mitigation on noisy quantum computers (2020), [arXiv:2009.04417](https://arxiv.org/abs/2009.04417) [quant-ph].
- [52] S. Endo, Z. Cai, S. C. Benjamin, and X. Yuan, Hybrid quantum-classical algorithms and quantum error mitigation, *J. Phys. Soc. Jpn.* **90**, 032001 (2021).
- [53] W. J. Huggins, S. McArdle, T. E. O'Brien, J. Lee, N. C. Rubin, S. Boixo, K. B. Whaley, R. Babbush, and J. R. McClean, Virtual Distillation for Quantum Error Mitigation, *Phys. Rev. X* **11**, 041036 (2021).
- [54] B. Koczor, Exponential Error Suppression for Near-Term Quantum Devices, *Phys. Rev. X* **11**, 031057 (2021).
- [55] P. Czarnik, A. Arrasmith, P. J. Coles, and L. Cincio, Error mitigation with Clifford quantum-circuit data, *Quantum* **5**, 592 (2021).
- [56] Z. Cai, Multi-exponential error extrapolation and combining error mitigation techniques for NISQ applications, *npj Quantum Inf.* **7**, 80 (2021).
- [57] A. Mari, N. Shammah, and W. J. Zeng, Extending quantum probabilistic error cancellation by noise scaling, *Phys. Rev. A* **104**, 052607 (2021).
- [58] A. Lowe, M. H. Gordon, P. Czarnik, A. Arrasmith, P. J. Coles, and L. Cincio, Unified approach to data-driven quantum error mitigation, *Phys. Rev. Research* **3**, 033098 (2021).
- [59] D. Bultrini, M. H. Gordon, P. Czarnik, A. Arrasmith, P. J. Coles, and L. Cincio, Unifying and benchmarking state-of-the-art quantum error mitigation techniques, [arXiv:2107.13470](https://arxiv.org/abs/2107.13470) [quant-ph].
- [60] C. Piveteau, D. Sutter, and S. Woerner, Quasiprobability decompositions with reduced sampling overhead, *npj Quantum Inf.* **8**, 12 (2022).
- [61] N. Berthussen, T. V. Trevisan, T. Iadecola, and P. P. Orth, Variational Trotter compression (VTC) algorithm applied to simulation of Heisenberg model dynamics (2022), https://figshare.com/articles/software/Variational_Trotter_compression_VTC_algorithm_applied_to_simulation_of_Heisenberg_model_dynamics/17712053