Variational preparation of normal matrix product states on quantum computers

Ben Jaderberg [1,*] George Pennington [2] Kate V. Marshall [1] Lewis W. Anderson [1] Abhishek Agarwal [3] Lachlan P. Lindoy [3] Ivan Rungger [3,4] Stefano Mensa [2] and Jason Crain [5,6]

[1]*IBM Quantum, IBM Research Europe, Hursley, Winchester SO21 2JN, United Kingdom*

[2]*The Hartree Centre, STFC, Sci-Tech Daresbury, Warrington WA4 4AD, United Kingdom*

[3]*National Physical Laboratory, Hampton Road, Teddington TW11 0LW, United Kingdom*

[4]*Department of Computer Science, Royal Holloway, University of London, Egham TW20 0EX, United Kingdom*

[5]*IBM Research Europe, The Hartree Centre, Sci-Tech Daresbury, Warrington WA4 4AD, United Kingdom*

[6]*Department of Physics, Clarendon Laboratory, University of Oxford, Oxford OX1 3QU, United Kingdom*

(Received 28 March 2025; accepted 10 December 2025; published 26 January 2026)

Preparing matrix product states (MPSs) on quantum computers is an essential routine in the simulation of many-body physics. However, widely used schemes based on staircase circuits are often too deep to execute on current hardware. Here, we demonstrate that MPSs with short-range correlations can be prepared with shallow circuits by leveraging heuristics from approximate quantum compiling (AQC). We achieve this with ADAPT-AQC, an adaptive-ansatz preparation algorithm, and introduce a generalized initialization procedure for the existing AQC-Tensor algorithm. We first compare these methods for the task of preparing a molecular electronic structure ground state. We then use them to prepare an antiferromagnetic (AFM) ground state of the 50-site Heisenberg XXZ spin chain near the AFM-XY phase boundary. Through the execution of circuits with up to 59 CZ depth and 1251 CZ gates, we perform a global quench and observe the relaxation of magnetic ordering in a parameter regime previously inaccessible due to deep ground-state preparation circuits. Our results demonstrate how the integration of quantum and classical resources can push the boundary of what can be studied on quantum computers.

DOI: 10.1103/m5kb-4f43

I. INTRODUCTION

Matrix product states (MPSs) are a powerful tool in the study of quantum physics, serving as a compact representation of wavefunctions with low entanglement. Despite being classically simulable, the correspondence of efficient MPSs to the ground states of gapped local Hamiltonians makes preparing them on quantum computers an important subroutine in quantum algorithms that generate high entanglement elsewhere. Formally, the task of preparing an MPS on a quantum computer requires the construction of a sequence of unitaries such that their action on a set of qubits produces a state equivalent to the MPS up to a desired accuracy.

In the general case, exactly preparing an MPS of maximum bond dimension χ requires the sequential application of $\log_2(\chi)$-local unitaries, producing a circuit with a "staircase" structure [1]. This structure is necessary to capture MPSs with maximal correlation length, and as such, the same blueprint is used widely by approximate [2] and variational [3–5] preparation methods including of infinite MPSs [6,7]. However, certain classes of MPS may admit significantly shallower circuits by using different ansatz structures. Identifying these solutions is of practical relevance for quantum computing, since shallower circuits produce less noise in the near term and incur fewer resources in the fault-tolerant era.

This was the original motivation behind AQC-Tensor [8,9], an MPS preparation algorithm using classical variational optimization of a brickwork ansatz. This ansatz has been shown to be particularly effective in a number of application areas [10–12]. Notably, the development of AQC-Tensor demonstrated the fundamental connection between fixed-input approximate quantum compiling (AQC) [13,14] and MPS preparation. When considering one-dimensional (1D) systems at quantum utility scales (i.e., beyond the reach of state-vector simulation), AQC algorithms in fact require solving the problem of MPS preparation as a subroutine.

In this work, we extend the use of AQC-inspired solutions for MPS preparation, enabling the study of higher-entanglement, more complex states on quantum computers. We explicitly consider normal MPSs [15], formally defined by the spectrum of their transfer matrix [16,17], but importantly characterized by having short-range correlations. This property has made normal MPSs the subject of extensive research into asymptotically optimal preparation schemes [18–20] and also implies their suitability to brickwork or other types of ansatz structures [3,21] for variational methods.

First, we describe an adaptive-ansatz MPS preparation algorithm (ADAPT-AQC) where the structure of the circuit is dynamically chosen during optimization. This follows the success of other variational algorithms with adaptive ansätze

*Contact author: benjamin.jaderberg@ibm.com

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

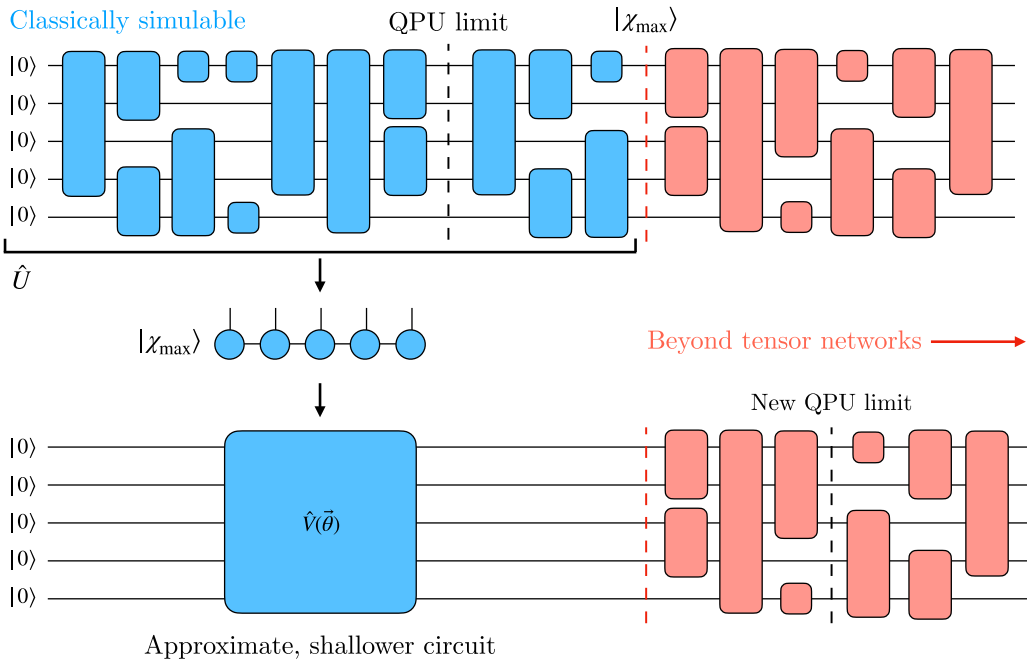


FIG. 1. Generalization of the scheme first proposed in Ref. [8]. A subcircuit of a quantum algorithm can be classically simulated using tensor networks such as matrix product states. From this, a shallower circuit can be found using approximate quantum compiling, which is equivalent to MPS preparation for large 1D circuits. In an idealized setting, the bond dimension $\chi_{\max}$ required to simulate $|\chi_{\max}\rangle$ is at the limit of high-performance computing (HPC). In this regime, the depth reduction found by AQC may allow the execution of a quantum circuit beyond the capabilities of tensor networks.

[22–24]. At each iteration, a selection criterion determines the placement of a two-qubit unitary, after which all parameters are updated via local and global sweeps. We demonstrate how this enables the preparation of several types of normal MPSs, without the need to assume a particular ansatz structure.

Second, we develop an initialization scheme for AQC-Tensor that enables it to be applied to preparing ground states. We show that by initializing the ansatz to prepare the compressed $\chi = 1$ approximation of the target MPS, the initial fidelity and parameter gradients are large enough to enable convergence of the algorithm when preparing ground states. We also find that this initialization strategy scales significantly better than alternatives, such as initializing the ansatz parameters randomly or to form the identity.

Following benchmarking on preparing random matrix product states (RMPSs), we apply these methods to two physical systems. First, we prepare the ground state of a molecular electronic structure Hamiltonian used in the modeling of proton transfer in malonaldehyde. Of the AQC methods, we find that only ADAPT-AQC is able to produce a high-fidelity solution, giving early evidence for the potential of an adaptive heuristics in MPS preparation. In the second case, we consider preparing an antiferromagnetic (AFM) ground state near the XY phase boundary of the 50-site Heisenberg XXZ model. Here, both AQC methods achieve significant depth reduction compared to competing methods, while AQC-Tensor with the initialization strategy produces the shallower solution of the two. We use these circuits to subsequently study the dynamics of the system following a global quench from the ground state. Through the execution of circuits with up to 59 CZ depth and 1251 CZ gates, we accurately obtain the signature relaxation of magnetic ordering to within statistical errors. Promisingly, this represents a significant advancement in the complexity of large-scale XXZ quenches realized on real quantum hardware, which were previously restricted to starting from the Néel product state [25,26]. Our results demonstrate how the addition of classical tensor network resources can push the boundary of what can be studied on quantum computers. Given the ability to prepare an MPS of sufficiently large bond dimension, this scheme could pave the way for simulations beyond the capabilities of tensor networks [27], as illustrated in Fig. 1.

II. METHOD

A. Approximate quantum compiling as an MPS preparation algorithm

Given a quantum algorithm, of which the core routine involves executing a quantum circuit $|\psi_t\rangle = \prod_i \hat{U}_i|\psi\rangle$, the goal of AQC is to find an alternative sequence of unitaries $|\tilde{\psi}_t\rangle = \prod_j \hat{V}_j|\psi\rangle$ that produces an approximately equal state but with a smaller resource overhead. More precisely, given a desired approximation error ϵ, AQC is successful if the cost function

$$C = 1 - |\langle\tilde{\psi}_t|\psi_t\rangle|^2 \tag{1}$$

satisfies the constraint $C \leqslant \epsilon$. This formulation of AQC is analogous to approximate state preparation and is different from the task of compiling the unitary sequence for any initial state.

In AQC, the resource being reduced is often the two-qubit depth (the longest nonparallel sequence of two-qubit unitaries), or the total number of two-qubit unitaries. Each of these are determinants of the amount of noise produced when executing the circuit on current quantum computers,

depending on the assumed noise model. However, prioritizing CNOT depth is important for reducing noise on current superconducting quantum computers, where the accumulation of dephasing and decoherence over time is more significant than errors from imperfect execution of gates. Furthermore, even in the era of fault-tolerant quantum computing, minimizing resources such as non-Clifford gates remains a large priority in reducing run-time.

The AQC procedure involves optimizing a variational ansatz $|\tilde{\psi}_t(\vec{\theta})\rangle = \prod_j \hat{V}_j(\vec{\theta}_j)|\psi\rangle$ in order to minimize the cost function in Eq. (1). At each step of the procedure, the cost function is evaluated based on the parametrization of the current best guess for the solution. If the cost is below the threshold ϵ, then AQC terminates, otherwise the parameters $\vec{\theta}$ are updated according to a classical optimizer.

This procedure has been used across many different works [13,14,28] with many different variations [29–31]. However, in all of these works, the cost function is evaluated using a classical state-vector simulator, limiting practical usage of AQC to $N \leqslant 20$ qubits due to the exponentially increasing memory usage. Nonetheless, as first proposed in AQC-Tensor [8], it is possible to perform AQC at quantum utility scales of $N \geqslant 50$ qubits with the use of tensor networks. Focusing specifically on 1D geometries, the first step is to compute the MPS representation of the target circuit:

$$|\psi_t\rangle = \sum_{i_1,i_2,\ldots,i_L} A_1^{i_1} A_2^{i_2} \cdots A_L^{i_L} |i_1 i_2 \cdots i_L\rangle, \tag{2}$$

where $A_k^{i_k}$ is a matrix with dimension $\chi_{k-1} \times \chi_k$, which defines the bond dimensions, and the indices i_k run over the qubit computational basis states. This target MPS can be obtained with an MPS simulator of quantum circuits such as Qiskit Aer [32].

The second step of utility-scale AQC is to variationally train a circuit to reproduce the target MPS in Eq. (2). At every step of optimization, an MPS representation of the ansatz circuit $\prod_j \hat{V}_j(\vec{\theta}_j)|\psi\rangle$ is computed using the same simulator, from which Eq. (1) can be evaluated via contractions between the two MPSs. Once the routine is finished, a compiled circuit is produced. Thus, given that this second step is entirely focused on MPS to circuit conversion, it is simple to see that MPS preparation is in fact a core subroutine of utility-scale AQC. Furthermore, it is not necessary for the target MPS to be derived from a circuit, and instead the ansatz optimization step of AQC can be used to directly prepare any MPS. For example, in Sec. III C we use AQC methods to prepare an MPS produced by density matrix renormalization group (DMRG) [33,34], which had no initial circuit representation.

B. Specific utility-scale AQC algorithms

1. AQC-Tensor

AQC-Tensor, first developed in Ref. [8] and now available as a Qiskit addon [9], is an AQC algorithm using MPS simulation. We refer the reader to the original references for more detail and best practices, but provide a brief overview here.

In AQC-Tensor, a fixed ansatz with brickwork structure is optimized to prepare a target MPS. The ansatz is optimized by minimizing Eq. (1), or using a modified cost that includes additional target states with fixed Hamming distance from the true target [35]. As described generally in Sec. II A, the optimization is performed classically, at scale, using efficient MPS contractions to obtain the cost function.

In this work, we use the following specific settings. The two-qubit gate used in the AQC-Tensor ansatz is a parametrized SU(4) unitary decomposed via a Cartan decomposition [36]. We then define a layer of the ansatz to consist of the repetition of this unitary in a brickwork structure and the full ansatz to consist of $\mathcal{L}$ layers. The ansatz parameters are trained using either the L-BFGS-B [37] or Adam [38] optimizers to minimize the fidelity cost as defined by Eq. (1).

An important factor for achieving convergence with AQC-Tensor, and for variational algorithms in general [39–41], is the quality of the initial guess for the ansatz parameters. In previous work, AQC-Tensor was applied to preparing time-evolved states, where a good initialization comes naturally from considering circuits derived from a few large Trotter steps (TSs) [8]. However, in the context of preparing arbitrary MPSs, we propose a general initialization strategy for AQC-Tensor. Given the target MPS $|\psi_t\rangle$ of maximum bond dimension χ, we initialize the ansatz to a product state corresponding to the best compressed $\chi = 1$ approximation of the state. This compression can be done using either singular value decomposition or variational optimization of the tensors, the latter of which is more effective for large compression. Once the compressed MPS $|\psi_{t_{\chi=1}}\rangle$ is obtained, each tensor $A_k^{i_k}$ is directly encoded as a single-qubit unitary into the corresponding kth qubit of a quantum circuit. This quantum circuit is then passed to AQC-Tensor, which sets the initial parameters of the ansatz to produce the same state.

We find that this simple initialization scheme is essential to successfully prepare the MPSs considered in this work and enables the optimization to start from an initial fidelity several orders of magnitude larger than other initialization strategies. Numerical evidence of this, specifically for the XXZ Heisenberg model, is given in Sec. III C. Furthermore, finding the initialization circuit is computationally cheap; for the same XXZ model experiment, the wall time is on the order of seconds.

2. ADAPT-AQC

Typically, AQC uses a predetermined structure for the variational ansatz, with each iteration of optimization focused on updating the gate parameters. In adaptive AQC, the variational ansatz is built dynamically throughout the optimization process, with gates being added at each iteration based on the current state of the solution. AQC with an adaptive ansatz was first introduced as incremental structural learning (ISL) [24] and has been used to compile small circuits [42–45]. This follows a wider adoption of adaptive ansätze in variational quantum algorithms such as ADAPT-VQE [22] and ADAPT-QAOA [23].

In this work, we develop ADAPT-AQC, a utility-scale AQC algorithm and a method for MPS preparation. This algorithm builds on top of ISL, using the same core routine and optimizer [46], but represents a significant evolution through its use of MPSs to run at scales beyond classical state-vector simulation. The code for ADAPT-AQC is available in Ref. [47].

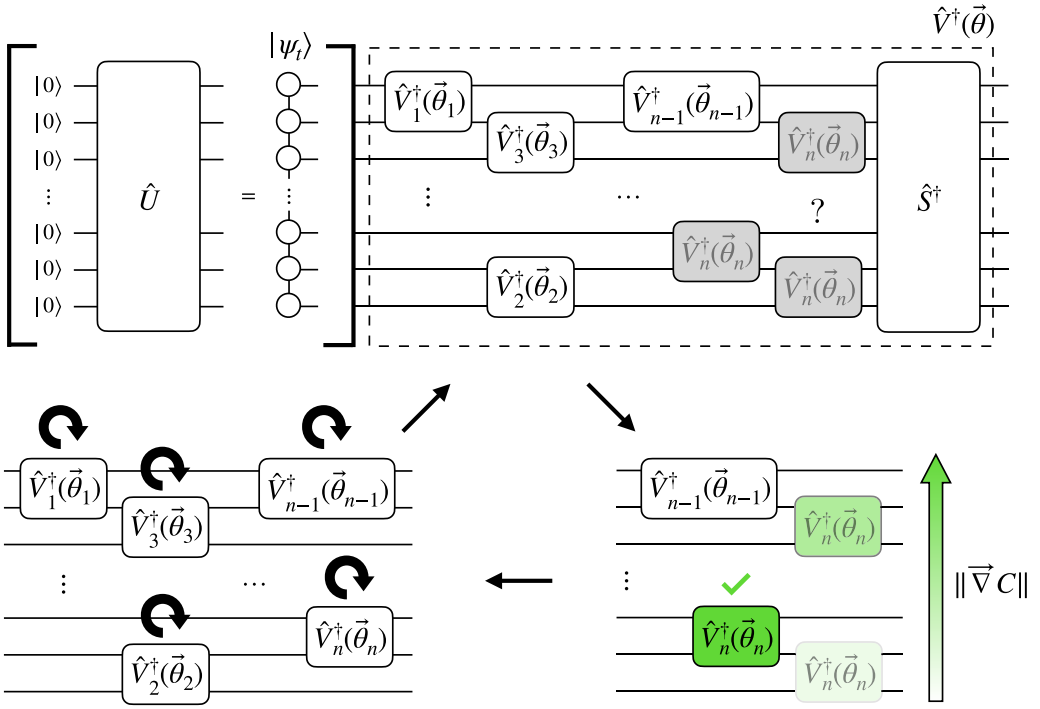


FIG. 2. An intermediate stage of the ADAPT-AQC algorithm. (Top) The compilation target is an MPS, which may be generated from a circuit $|\psi_t\rangle = \hat{U}|0\rangle$. This is acted on by an ansatz $\hat{V}^\dagger(\vec{\theta})$, consisting of two-qubit unitaries $\hat{V}_i^\dagger(\vec{\theta}_i)$ and nonparametrized gates $\hat{S}^\dagger$. On the nth iteration, a unitary $\hat{V}_n^\dagger(\vec{\theta}_n)$ is added to one of several possible positions. (Bottom right) Of these, the one with the largest cost gradient $\|\vec{\nabla} C\|$ is chosen, pictured in solid green. (Bottom left) After adding the unitary, all parameters are optimized to minimize the cost.

Figure 2 gives an overview of ADAPT-AQC. On the left, the target quantum state $|\psi_t\rangle = \hat{U}|0\rangle$ is first transformed into its MPS representation. Notably, it is also possible to pass in the target MPS directly, without the need for an initial circuit representation. This allows for convenient interfacing of ADAPT-AQC with classical tensor network algorithms such as DMRG or time-dependent variational principle (TDVP).

With the target MPS defined, the right-hand side of Fig. 2 shows the intermediate state of circuit preparation after $n-1$ iterations of the algorithm. In the Loschmidt Echo formalism of AQC [14], the variational ansatz $\hat{V}^\dagger(\vec{\theta})$ (dashed black box) acts directly on the target such that the entire circuit produces the MPS $|\psi_{n-1}\rangle = \hat{V}^\dagger(\vec{\theta})\hat{U}|0\rangle$. The ansatz itself consists of a sequence of adaptively added unitaries $\hat{V}_i^\dagger(\vec{\theta}_i)$ and a nonparametrized circuit $\hat{S}^\dagger$. We refer to its conjugate $\hat{S}$ as the starting circuit, since ADAPT-AQC is initialized with just this block to begin with. Hence, $\hat{S}$ should ideally be a shallow quantum circuit that prepares a good starting point, if one is known. Borrowing from the findings of AQC-Tensor in Sec. II B 1, we use an initialization strategy in which $\hat{S}$ is set to the circuit that prepares a compressed $\chi = 1$ approximation of the target MPS.

As shown in Fig. 2, at the start of the nth iteration, all possible positions for the next two-qubit unitary $\hat{V}_n^\dagger(\vec{\theta}_n)$ are drawn up based on the desired connectivity of the circuit. Here, we visualize a nearest-neighbor connectivity as a sensible default, to preserve the 1D structure that allows for performant MPS simulation. Apart from the condition to not pick the same pair of qubits twice, the choice of where to place the unitary is based on properties of the state $|\psi_{n-1}\rangle$. In previous works, the primary method was to choose the pair of qubits with the highest pairwise entanglement, with the goal to disentangle the target circuit [24]. However, we find that a selection based on maximizing the cost function gradient $\|\vec{\nabla} C\|$ is more effective for the problems studied in this work. More details of the gradient-based selection scheme are given in Appendix A.

After the unitary is added to the circuit, the parameters $\vec{\theta}_n$ are optimized to minimize the cost using `rotoselect` [48] and subsequently the parameters of all adaptive blocks $\{\vec{\theta}_1, \ldots, \vec{\theta}_n\}$ are optimized using `rotosolve` [48]. This sequential optimization sweeps across the ansatz and locally updates each block, repeating until self-consistency is achieved.

Optimization requires computing the cost function, which can be achieved at scale by first simulating the circuit in Fig. 2 with the Qiskit Aer MPS simulator [32]. This produces the MPS

$$|\psi_n\rangle = \hat{S}^\dagger \hat{V}_n^\dagger(\vec{\theta}_n) \cdots \hat{V}_1^\dagger(\vec{\theta}_1)\hat{U}|0\rangle, \tag{3}$$

which is subsequently contracted with the $|0\rangle$ MPS to compute the cost function

$$C = 1 - |\langle 0|\psi_n\rangle|^2, \tag{4}$$

which when expanded is equivalent to Eq. (1). In Appendix B, we discuss steps taken to reduce the overhead of the simulation required to obtain $|\psi_n\rangle$. Finally, once a sufficient cost $C < \epsilon$ is reached, the algorithm terminates. At this point, inverting the ansatz $\hat{V}^\dagger$ gives the unitary

$$\hat{V} = \hat{V}_1(\vec{\theta}_1) \cdots \hat{V}_N(\vec{\theta}_N)\hat{S}, \tag{5}$$

TABLE I. Fidelities, CNOT depths, and CNOT counts for circuits obtained from compiling 100 random $L = 50$ MPSs of bond dimension $\chi = 2$. Values are quoted as mean ± standard deviation (minimum, maximum). Values without a standard deviation, minimum, and maximum are equal for all 100 instances.

Method	Fidelity	CNOT Depth	CNOT Count
Schön *et al.*[1] and Ran [2]	1.0	98	98
ADAPT-AQC	0.988 ± 0.008	22 ± 9 (9, 48)	208 ± 56 (109, 363)
AQC-Tensor [8]	0.9902 ± 0.0002	6	147

which prepares the state $|\tilde{\psi}_t\rangle = \hat{V}|0\rangle$. This state is an approximation of the target state $|\psi_t\rangle$, with fidelity greater than $1 - \epsilon$.

III. RESULTS

For all results, the default ADAPT-AQC and AQC-Tensor configuration in Appendix C is used unless otherwise stated.

A. Random matrix product states

We first apply AQC-Tensor and ADAPT-AQC to the task of preparing RMPSs. RMPSs are known to exhibit short-range correlations [49,50], making them a common benchmark for preparation algorithms focused on normal MPSs [18,20]. Furthermore, the preparation of RMPSs is of interest due to their correspondence with ground states of disordered locally interacting Hamiltonians [16] and as a tool to study quantum statistical mechanics [51].

We benchmark the AQC methods against the widely used Schön *et al.* [1] and Ran, [2] MPS preparation schemes as implemented in Ref. [52]. In the former, each tensor of maximum dimension χ is directly encoded into a staircase of $\log_2(\chi)$-local unitaries, exactly preparing the MPS. In the latter, the MPS is approximately prepared using successive staircases of two-qubit unitaries, each of which encodes a $\chi = 2$ decomposition of the intermediate state.

In this experiment, we specifically consider preparing 100 random instances of $L = 50$ MPSs, generated by applying random parametrized unitaries in one layer of a brickwork structure. We prepare $\chi = 2$ MPSs specifically as a minimum nontrivial MPS of this class, which has been previously used to benchmark methods [20]. The random circuits are defined by the spin layout pattern in Refs. [53,54]. For the AQC methods, each random instance is prepared with a target fidelity of 99%. Our results are summarized in Table I.

In the case of $\chi = 2$, the Schön and Ran methods are identical and exact, and the compiled circuits consist of a single staircase of 49 two-qubit unitaries. Due to the structure of the target MPSs, each unitary reduces to two CNOTs during transpilation, resulting in a CNOT count and depth of 98 for all instances. These circuits have a low CNOT count, but a large CNOT depth due to the staircase structure.

For ADAPT-AQC, all 100 instances are compiled to circuits with mean fidelity of 98.8%, mean CNOT count of 208 and a mean CNOT depth of 22. These circuits have a larger CNOT count than those obtained using the Schön and Ran methods, but a much lower CNOT depth. Importantly, this is a trade-off that is beneficial for reducing noise on current quantum hardware, as shown later in Sec. III C.

For AQC-Tensor, a solution is found with at least 99% fidelity for all 100 instances. We choose only a single ansatz layer $\mathcal{L} = 1$, since this structure maps directly to the circuits used to generate each RMPS. This means that all compiled circuits produced have a CNOT count of 147 and a CNOT depth of 6. Thus, AQC-Tensor prepares the target RMPSs with the lowest CNOT depth. In terms of the overall number of CNOT gates, the AQC-Tensor solution contains fewer on average than ADAPT-AQC. However, for 12 of the 100 instances, the AQC-Tensor solution has more CNOT gates.

Finally, we note that here we have not considered the scheme presented in Ref. [20], for which an asymptotic improvement over Schön and Ran was numerically observed for Haar-random $\chi = 2$ MPSs. Adding a comparison to this method would help establish a wider benchmark, which we leave for future work.

B. Preparing chemistry ground states with quantum nuclei

Proton-transfer reactions are fundamental to a broad array of chemical and biological processes, including acid-base chemistry, enzymatic function, and proton-coupled electron transfer in energy materials [55,56]. Due to the small mass of the proton, these processes are strongly influenced by nuclear quantum effects such as tunneling and zero-point motion—phenomena not adequately captured by classical or semiclassical treatments [57]. Accurately resolving these effects requires a fully quantum mechanical treatment of the nuclear degrees of freedom, which remains computationally demanding on classical hardware, especially for systems with complex potential energy surfaces.

Quantum computing offers a natural framework to represent and evolve such quantum states directly, but the deep circuits typically required for high-fidelity state preparation are beyond the capabilities of current devices. In this context, proton-transfer dynamics provide a well-motivated setting for exploring AQC techniques that can reduce circuit depth while preserving essential quantum features of the problem.

Here, we consider specifically the problem of simulating adiabatic proton transfer in malonaldehyde, which is a prototypical system used to investigate quantum effects in molecular proton-transfer reactions [58,59]. As the transfer is adiabatic, we apply the MPS preparation techniques already established for the task of ground-state preparation.

1. Malonaldehyde

The generalized Hamiltonian for the nuclear-electronic orbital (NEO) model, which is used to describe proton-transfer processes [60,61], is given by [62]

$$\hat{H} = \sum_{ij} h_{ij}\, \hat{a}_i^\dagger \hat{a}_j + \sum_{IJ} h_{IJ}\, \hat{a}_I^\dagger \hat{a}_J + \frac{1}{2} \sum_{ijkl} h_{ijkl}\, \hat{a}_i^\dagger \hat{a}_k^\dagger \hat{a}_l \hat{a}_j + \frac{1}{2} \sum_{IJKL} h_{IJKL}\, \hat{a}_I^\dagger \hat{a}_K^\dagger \hat{a}_L \hat{a}_J - \sum_{ijKL} h_{ijKL}\, \hat{a}_i^\dagger \hat{a}_K^\dagger \hat{a}_L \hat{a}_j$$

TABLE II. Preparing the $L = 18$ MPS corresponding to a ground state within the adiabatic proton-transfer process of the malonaldehyde molecule.

Method	Fidelity	CNOT depth	CNOT count
Schön *et al.*	1.0	634	649
Ran	0.991	45	109
ADAPT-AQC	0.990	48	96
AQC-Tensor (VQE)	0.990	608	1489
AQC-Tensor	0.973	6–84	51–688

$$+\sum_{IJ,A} h_{IJ,A}\,\hat{a}_I^\dagger \hat{a}_J - \sum_{ij,A} h_{ij,A}\,\hat{a}_i^\dagger \hat{a}_j + \frac{1}{2}\sum_{AB} \frac{Z_A Z_B}{|\boldsymbol{R}_A - \boldsymbol{R}_B|}, \tag{6}$$

where i, j, k, l denote electronic spin orbitals, I, J, K, L denote spin orbitals of any protons treated quantum mechanically, A, B denote molecular nuclei treated as classical point charges, and $\hat{a}$ and $\hat{a}^\dagger$ are the fermionic annihilation and creation operators, respectively.

We obtain the ground state of malonaldehyde from Ref. [63]. Briefly, this is computed by first reducing the dimensionality of the active space of Eq. (6) using the frozen natural orbitals (FNO) approximation [64]. Here, a set of natural orbitals are defined and, by identifying and discarding orbitals with occupation below a cutoff, a reduced active space is obtained [65]. The full Hamiltonian is then projected into the FNO space and subsequently diagonalized using complete active space configuration interaction [66].

Within this reduced space, proton transfer is modeled by defining three reference Hamiltonians H_L, H_M, and H_R, corresponding to the NEO optimized orbitals at the left minimum, barrier, and right minimum of the malonaldehyde potential energy surface. These are combined linearly via

$$\hat{H}(\alpha, \beta, \gamma) = \alpha H_L + \beta H_M + \gamma H_R, \quad \alpha + \beta + \gamma = 1. \tag{7}$$

In the adiabatic limit, the system can be considered to remain in a set of instantaneous ground states $|\Psi_0(\alpha, \beta, \gamma)\rangle$ of $\hat{H}(\alpha, \beta, \gamma)$. Thus, it is possible to map the dynamical proton motion onto a sequence of ground-state problems [62].

We focus on the particular interpolation point $(\alpha, \beta, \gamma) = (0, \frac{1}{3}, \frac{2}{3})$, which represents a state where the proton has crossed the middle barrier and is approaching the right equilibrium point. Specifically, we take the corresponding 18-qubit variational quantum eigensolver (VQE) circuit solution from Ref. [63], available at Ref. [67], and simulate it with the Qiskit Aer MPS simulator. With truncation settings such that Schmidt coefficients with a sum of squares less than 10^{-4} are discarded, the resultant MPS is of length $L = 18$ and maximum bond dimension $\chi = 7$.

2. MPS preparation

Obtaining the target MPS with the procedure above, we apply each of the MPS preparation methods from Sec. III A to the malonaldehyde ground state. The results are shown in Table II.

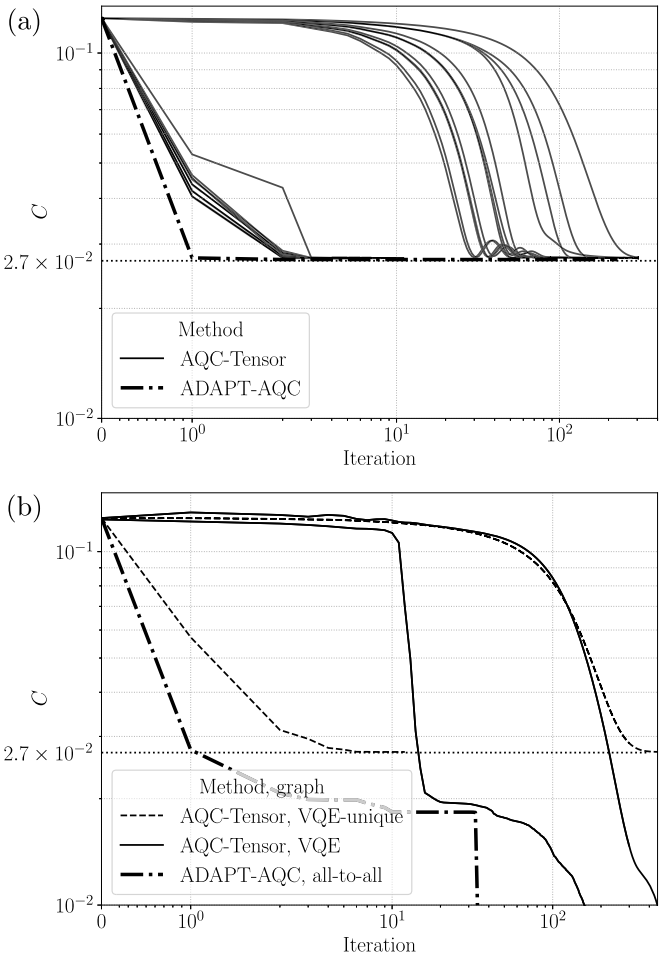


FIG. 3. Cost $C = 1 - |\langle\tilde{\psi}_t|\psi_t\rangle|^2$ as a function of compiling iteration for the AQC methods reported in Table II. (a) Both methods are restricted to nearest-neighbor connections only. Each solid line is AQC-Tensor run with a different number of layers ($\mathcal{L} = 1$ to $\mathcal{L} = 14$) and optimizer. (b) Both methods are allowed to use ansätze with longer-range interactions as defined in the text.

Here, Schön provides a baseline for preparing the state, which can be achieved with 1.0 fidelity using a circuit of 634 CNOT depth. In terms of approximate preparation, Ran is able to prepare the state to a fidelity of 0.991 with $\mathcal{L} = 4$ staircase layers. While this initially maps to a depth 60 circuit, we find it can be reduced to depth 45 by removing trivially small unitaries.

For AQC-Tensor, we find that when running with between $\mathcal{L} = 1$ to $\mathcal{L} = 14$ layers and trying several optimizers, in each instance the compilation cannot produce a solution with higher than fidelity 0.973. As shown by the solid lines in Fig. 3(a), all 28 AQC-Tensor experiments reach a cost value $C = 2.7 \times 10^{-2}$ (to three significant figures) and the algorithm exits. The solid lines grouped to the right are optimized using Adam and to the left are optimized using L-BFGS-B. This suggests the presence of a local minimum that inhibits the convergence of the algorithm.

When running ADAPT-AQC with the default settings, including restricting the adaptive process to only nearest-neighbor connectivity, the same local minimum is reached

as shown by the dash-dotted line in Fig. 3(a). However, by allowing the adaptive ansatz to pick from an all-to-all connectivity, the algorithm avoids this local minimum and converges to an $\mathcal{F} = 0.99$ solution as shown by the dash-dotted line in Fig. 3(b). Furthermore, even after transpiling the solution circuit back to 1D, through the use of SWAP operations, the circuit prepares the target state with a CNOT depth of only 48 as shown in Table II. This is partially because, while the adaptive heuristic considers all possible two-qubit pairs, in reality only a few nonlocal gates are chosen.

Finally, we consider the case where AQC-Tensor is also optimized using an ansatz with nonlocal connectivity. Unlike the other ground states studied in this work, in this setting we have access to a circuit that prepares the ground state, from the VQE solution of Ref. [63]. We use this to build two AQC-Tensor ansätze; one with a parametrized two-qubit block for each two-qubit interaction in the VQE circuit and another for each unique two-qubit connection. We call the connectivity graphs of each of these ansätze VQE and VQE unique, respectively.

From Fig. 3(b), we see that the VQE unique ansatz encounters the same local minima as the nearest-neighbor ansätze. However, the VQE ansatz converges to $\mathcal{F} = 0.99$ for both optimizers, highlighting the possible benefit of using problem-inspired ansätze. Nevertheless, the CNOT depth of the optimized VQE ansatz solution is 608, significantly deeper than when using the brickwork ansatz. Thus, in the case that we wish to prepare a high-accuracy ground state, with a shallow enough circuit to be amenable to current devices, we identify a first use case in which ADAPT-AQC produces a better solution than AQC-Tensor.

In a wider context, understanding what specific features of this ground state make it more or less amenable for each of the methods remains an important open question. Our results hint that adaptive-ansatz MPS preparation may be particularly helpful for MPSs whose effective interactions are not well captured by a 1D circuit, which may be more prevalent in quantum chemistry where Hamiltonians induce long-range couplings after mapping to qubits. However, if the number of long-range couplings grows with system size, such states may no longer be efficiently represented by MPS in the first place. Furthermore, our result here is the only application where we find that Ran produces a higher fidelity and lower depth than either of the AQC methods. This demonstrates the nuance of theory versus practice; even for MPSs with short-range correlations, it is possible for Ran to produce competitive circuits if there are sufficiently local approximations that break the staircase structure. Thus, it remains an open problem to identify ground states where adaptive-ansatz MPS preparation definitively outperforms all other methods.

C. Global quench of the 50-site XXZ Heisenberg model on a quantum computer

An important use of MPS preparation methods is to reduce the circuit depth of classically simulable subroutines in quantum algorithms that are classically intractable as a whole. This was the original motivation behind the AQC-Tensor algorithm [8], which demonstrated how short time dynamics can be represented classically by an MPS, compiled to a shallow circuit and then appended by more Trotter steps that are executed on quantum hardware [27].

A different setting in which circuit depth can be reduced via MPS preparation is a Hamiltonian quench [68–70]. In this experiment, the ground state of a locally interacting Hamiltonian is prepared, followed by time evolution under a different Hamiltonian, after which properties of the evolved state are explored. If the ground state is of a 1D gapped system, then MPS preparation methods represent an efficient way to construct the corresponding circuit. If the ground state is also unique, then it can be efficiently represented by a normal MPS [16], presenting an opportunity for shallow preparation with AQC methods.

In this work, we study initial and quenched Hamiltonians both belonging to the family of the XXZ Heisenberg spin chain. This is given by

$$H = \sum_{i=1}^{L-1} \left(S_i^x S_{i+1}^x + S_i^y S_{i+1}^y + J_z S_i^z S_{i+1}^z \right) - \sum_{i=1}^{L} h_z S_i^z, \tag{8}$$

where the spin operators S^p are related to the Pauli matrices by $S^p = \frac{1}{2}\sigma^p$, J_z is the anisotropy parameter, and h_z is the external uniform magnetic field strength.

In the following, we specifically consider a quench where the initial ground state is in the AFM phase, such that local pairs of spins are highly anticorrelated along the $\hat{Z}$ axis. This is followed by time evolution under a Hamiltonian whose ground state lies in the XY phase. Figure 4(a) gives an overview of this quench experiment.

Physically, the action of the quench causes a relaxation of the initial AFM ordering present in the ground state. A useful observable that defines the amount of AFM structure is the staggered magnetization (SM):

$$\frac{1}{L} \sum_{i=1}^{L} (-1)^i \langle S_i^z \rangle. \tag{9}$$

The complexity of the quantum many-body interactions in the quench is highly dependent on the Hamiltonian parameters chosen. A quench from the Néel product state $|\psi\rangle = |\uparrow\downarrow\uparrow\downarrow\cdots\rangle$, the ground state in the limit $J_z \to \infty$, is well studied classically [71,72]. In this case, the relaxation of the SM exhibits a simple behavior that is qualitatively reproduced by the exactly solvable XZ model [73]. Despite this, there are many quantum computing simulations of this quench [25,26] due to the ease of preparing the Néel state with only one layer of single-qubit gates. Quenching from initial states close to the phase boundary produces more intricate phenomena [73] with a unique pattern of SM relaxation. This is the setting we study in this work, for which even some modern classical methods fail to work (e.g., using the quench-action algorithm [74]).

1. *Preparing the ground state*

We first compute a ground state of Eq. (8) for $L = 50$ sites with parameters $J_z = 2.5$ and $h_z = 0.0$. This is achieved using the DMRG algorithm [33,34] in TeNPy [75] with a truncation cutoff of 10^{-4}, a maximum bond dimension of $\chi_{\max} = 100$, subspace-expansion mixer enabled, and a maximum of ten sweeps. Here, we choose a large $\chi_{\max}$ so that DMRG is effectively bound by the truncation; the largest χ explored during

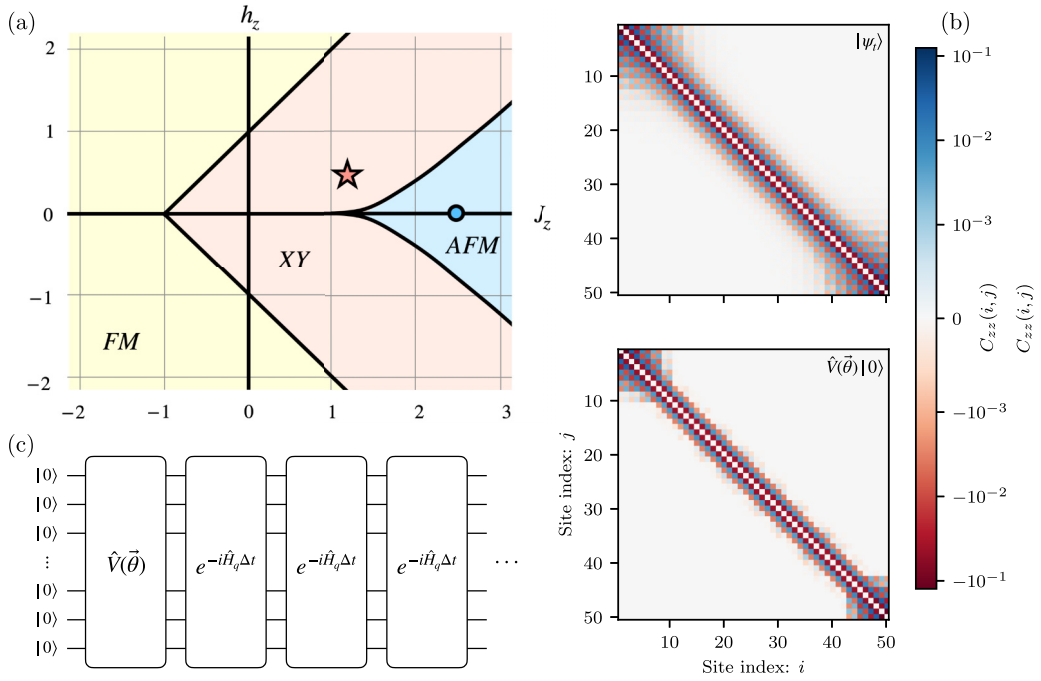


FIG. 4. Schematic of the XXZ quench experiment. (a) Parameter space of the quench, superimposed on a cartoon of the XXZ model phase diagram [76], including the gapped ferromagnetic (FM) and AFM phases and the gapless XY phase. We prepare an $L = 50$ site ground state for $J_z = 2.5$, $h_z = 0.0$ (blue dot) and perform a global quench under $J_z = 1.2$, $h_z = 0.5$ (red star). (b) The two-site correlation function $C_{zz}(i, j) = \langle S_i^z S_j^z\rangle - \langle S_i^z\rangle\langle S_j^z\rangle$ of the ground state $|\psi_t\rangle$ obtained using DMRG and of the circuit that prepares that ground state $\hat{V}(\vec{\theta})|0\rangle$, in this case using AQC-Tensor. Values at $i = j$ are left undefined. (c) For all MPS preparation methods used, the circuit that prepares the ground state $\hat{V}(\vec{\theta})|0\rangle$ is followed by Trotterized evolution under the quench Hamiltonian.

the calculation is $\chi = 14$. The algorithm converges with a final change in energy $\Delta E \sim 10^{-10}$, producing a $\chi = 14$ MPS with maximum truncation error on the order 10^{-8}. We note that due to symmetry of the Hamiltonian, this ground state is degenerate with the state with all spins flipped.

Despite its antiferromagnetic ordering, this ground state is fundamentally different from the Néel state due to the presence of entanglement. Figure 4(b) shows the two-site correlation function $C_{zz}(i, j) = \langle S_i^z S_j^z\rangle - \langle S_i^z\rangle\langle S_j^z\rangle$ of the chain for the target MPS $|\psi_t\rangle$ produced by DMRG. For the Néel state, $C_{zz} = 0$ for all pairs, since it is a classical approximation of a quantum magnet whereby entanglement is neglected. In contrast, the DMRG ground state contains short-range correlations, indicating the presence of quantum mechanical interactions that deviates away from an idealized fully ordered state. In this way, the prepared ground state is a more realistic representation of a quantum antiferromagnet.

Subsequently, we generate circuits that prepare the ground-state MPS using the different methods described in Sec. II. The fidelity, two-qubit gate count, and two-qubit gate depth are summarized in Table III.

The Schön method is the only approach that prepares the MPS exactly, with perfect fidelity between the ground state and the state produced by the circuit. However, the enormous CNOT overhead of applying a sequence of five-qubit unitaries makes this method impractically deep. Following this, we run the Ran method with between $\mathcal{L} = 1$ and $\mathcal{L} = 12$ staircase layers, producing circuits that prepare the MPS with a fidelity of 0.891 through 0.952, respectively. However, we find that the $\mathcal{L} = 5$ solution, presented in Table III, is the most layers possible without exceeding the depth limit of the IBM Quantum platform [77]. This circuit contains far fewer CNOT gates and is much shallower than the Schön solution, at the cost of a fidelity of only 0.939.

The ADAPT-AQC and AQC-Tensor algorithms both prepare the target MPS with a fidelity of 98%, producing circuits with 28 and 18 CNOT depth, respectively. While the prepared ground state is an approximation of the DMRG solution, Figure 4(b) shows how the dominant local correlations are still preserved, with only those orders of magnitude smaller being truncated. Compared to Ran, both methods produce circuits that are more accurate and significantly shallower, at the cost of up to 13% more CNOT gates.

Between the two variational methods, AQC-Tensor performs better, producing a ground state of the same accuracy with fewer operations. Moreover, AQC-Tensor and ADAPT-AQC converge with a wall-clock time of 13 and 96 h, respectively. This highlights the additional cost of adaptive methods, whereby several different ansätze are sequentially

TABLE III. Preparing the MPS corresponding to the ground state of the $L = 50$ XXZ model with $J_z = 2.5$ and $h_z = 0.0$.

Method	Fidelity	CNOT depth	CNOT count
Schön *et al.*[1]	1.0	46 157	48 407
Ran [2]	0.939	157	409
ADAPT-AQC	0.980	28	464
AQC-Tensor [8]	0.980	18	441

optimized instead of one. We stress however that since each algorithm is implemented by a different codebase, the run times are also a reflection of the efficiency of the implementation, and not purely a comparison of the underlying method. In addition, for AQC-Tensor, a minimum number of brickwork ansatz layers is not known *a priori*, so several compilations were run, with $\mathcal{L} = 3$ identified as the minimum sufficient number of layers afterward. These other runs are not factored into the 13-h run-time since they were executed in parallel.

Considering AQC-Tensor in more detail, we find that the ansatz initialization scheme described in Sec. II B 1 is essential to achieve the compiled solution. Initializing the ansatz parameters randomly, or such that the overall circuit equals the identity [78], leads to initial fidelities of 10^{-15} and 10^{-221}, respectively. In contrast, initializing the ansatz to produce the state $|\psi_{t_{\chi=1}}\rangle$ leads to an initial fidelity of 0.11, from which we have shown that convergence is achievable. In Appendix E, we analyze the performance of this scheme for preparing the same ground state with increasing system size L. While the initial fidelity decays exponentially with system size, it decreases slow enough that the absolute values may be large enough to be practically useful. For example at $L = 100$ and $L = 150$, the fidelity is $\mathcal{F} = 1.6 \times 10^{-2}$ and 2×10^{-3}, respectively. Furthermore, the exponent of the decay is over 200 times smaller than initializing the AQC-Tensor ansatz randomly or as the identity, indicating a significantly favorable scaling.

2. Quench results

With the ground-state circuits compiled, the full quench circuits are constructed by appending a second-order Suzuki-Trotter decomposition [79] of the time evolution operator $U(t) = \exp(-iH_q t)$, as visualized in Fig. 4(c). The time-evolved quench Hamiltonian H_q has parameters $J_z = 1.2$ and $h_z = 0.5$.

In this experiment, we simulate five Trotter steps of the quench Hamiltonian with step size $\Delta t = 1.0$. The corresponding circuits are first transpiled for execution on hardware, including converting the two-qubit gate set from CNOT to CZ, and then run on the *ibm_fez* quantum computer. Here, a range of error mitigation techniques are employed including twirled readout error extinction (TREX) [80], Pauli twirling [81], and zero-noise extrapolation (ZNE) [82]. More details of the error mitigation used can be found in Appendix F.

Figure 5 shows the results of the quench experiments. Due to the size of the simulation, $N = 50$ qubits, it is not possible to classically obtain an exact result with state-vector simulation. However, we compute the evolution using the TDVP algorithm [83] in TeNPy, with a truncation cutoff of 10^{-5}. Starting with the ground state, we evolve in steps of $\Delta t = 0.1$, reaching a maximum bond dimension of $\chi = 152$ by $t = 5.0$ with a truncation error on the order of 10^{-8}.

The TDVP simulated dynamics are illustrated by the solid black line. The relaxation of the SM follows an overall exponential decay, but with deviations arising from additional damped oscillatory terms, which are the hallmarks of a quench from near the phase boundary [73]. By comparison, the dashed gray line shows the TDVP evolution when considering a different quench. Starting from the Néel state, as previously studied on quantum hardware [25,26], significantly different

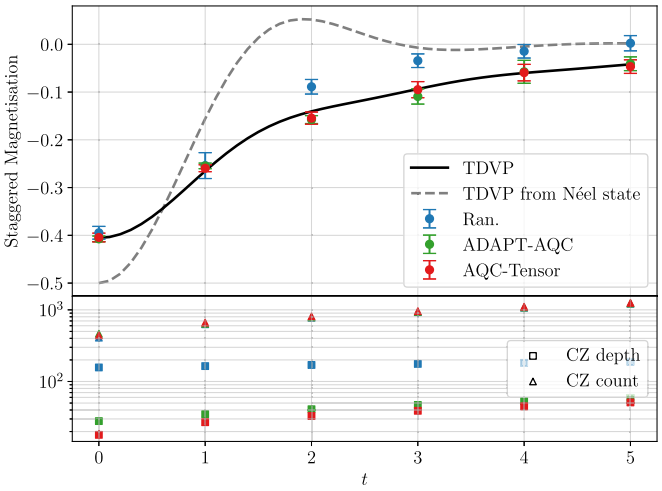


FIG. 5. (Top) Quantum computing simulation of an $L = 50$ site XXZ chain following a global quench, performed on the *ibm_fez* Heron processor. An antiferromagnetic ground state near the XY phase boundary ($J_z = 2.5$, $h_z = 0.0$) is prepared at $t = 0$ using one of the different matrix product state preparation methods. This is followed by evolution under parameters $J_z = 1.2$, $h_z = 0.5$, causing the staggered magnetization [Eq. (9)] to relax. The solid and dashed black lines show the quench dynamics when initialized in the $J_z = 2.5$, $h_z = 0.0$ ground state and Néel state, respectively, as calculated by TDVP. (Bottom) The CZ depth and CZ count of the executed circuits at each time step.

dynamics occur, whereby the SM relaxes much faster and reaches a stable equilibrium by approximately $t = 3$.

The circle markers correspond to the SM obtained on *ibm_fez*, where the ground states are prepared using the different circuits from Table III. For each marker, the error bar represents one standard deviation in the SM, computed by propagating the errors in the individual $\langle \sigma_i^z \rangle$ expectation values. These themselves are returned as an output of Qiskit's ZNE routine and are derived from shot noise, Pauli twirling spread, and uncertainty in the ZNE extrapolation fit.

The blue markers show the evolution when preparing the ground state using the Ran circuit. Despite the large CZ depth, the ground state ($t = 0$) and $t = 1$ SM are recovered within statistical errors. This is because the staircaselike structure of the circuit allows many of the qubits to delay initialization and be measured early. However, this effect becomes less pronounced with increasing Trotter steps, since each qubit interacts with an increasingly large light cone. As such, from $t = 2$ onward, it is not possible to obtain the TDVP solution, owing to the large depth of the Ran ground-state preparation. While the SM continues to tend toward 0, we note that this matches the expected behavior for systems thermalizing under random noise.

The green and red markers correspond to the SM obtained when using the ADAPT-AQC and AQC-Tensor circuits, respectively, to prepare the ground state. Due to the reduced depth of the ground-state circuits, as well as the higher fidelity, using these methods allows the quantum computing simulation to accurately obtain the signature relaxation of the SM. Physically, we observe the SM decaying from -0.404 to -0.047, capturing the process in which AFM ordering is

almost totally lost due to the quench. Both methods predict the SM accurately, within 1 standard deviation, except at $t = 2.0$ for ADAPT-AQC.

The bottom panel of Fig. 5 shows the CZ depth and CZ count at each Trotter step, the numerical values of which can be found in Table VI. The increase from $t = 0$ to $t = 1$ varies across methods, due to how the first Trotter step stacks next to each ground-state circuit. After this, both values increase equally at each time step for all methods. Most notable is the large difference in CZ depth between the AQC methods and Ran. Given the Ran, ADAPT-AQC, and AQC-Tensor methods have a respective CZ depth at $t = 5.0$ of 188, 59, and 51, our results highlight the importance of shallow ground-state preparation in order to achieve accurate quench dynamics.

IV. CONCLUSION

In this work, we have shown how to prepare instances of normal MPSs with shallow quantum circuits that can be executed accurately on current quantum computers. Key to this is the use of algorithms originally developed for AQC, whose pragmatic heuristics are able to exploit the short-range correlations of the targets. Specifically, we find that the existing AQC-Tensor algorithm [8] can be readily applied by using a parameter initialization scheme. Furthermore, we develop ADAPT-AQC, an MPS preparation algorithm in which the ansatz is dynamically built up and optimized incrementally. This enables the preparation of MPSs without the need to assume a particular ansatz structure, affording flexibility to prepare a wide range of physical systems. In particular, for quantum chemistry systems, we find early evidence that ADAPT-AQC can avoid local minima that fixed-ansatz AQC becomes stuck in.

Following this, we applied these methods to prepare a $\chi = 14$ AFM ground state of the 50-site XXZ Heisenberg spin chain. The resulting circuits of ADAPT-AQC and AQC-Tensor achieved higher fidelity and lower two-qubit gate depth compared to the Ran scheme [2], by 81% and 89%, respectively. Subsequently, we simulated a global quench on the *ibm_fez* quantum computer, evaluating circuits with up to 59 CZ depth and 1251 CZ gates. The reduced gate depth in preparing the ground state allowed us to accurately capture the signature decay of staggered magnetization for up to $t = 5$ of evolution. Our results represent a significant increase in complexity of XXZ model quenches studied on real quantum hardware, which were previously restricted to starting from unentangled ground states.

For preparing random MPSs and spin-chain ground states, AQC-Tensor produced circuits with smaller CNOT depth and fewer CNOT gates on average than ADAPT-AQC. Understanding why remains an interesting question since, in theory, the AQC-Tensor circuits are within the solution space of ADAPT-AQC, which has the freedom to pick the same structure. Furthermore, we find that even when forced to use a brickwork layout, ADAPT-AQC does not converge to the same fidelity and produces deeper solutions. This hints that the different optimization strategies are responsible; AQC-Tensor uses global gradient descent whereas ADAPT-AQC uses `Rotosolve`, a local nongradient optimizer. While the latter was designed to be beneficial when optimizing directly on quantum hardware, since ADAPT-AQC is a purely classical algorithm, a gradient-based optimizer could be more performant. Thus, an interesting immediate future direction would be to unify the two AQC approaches to see if even shallower MPS preparation can be achieved.

In Sec. III, ADAPT-AQC and AQC-Tensor are compared to widely used exact [1] and approximate [2] sequential MPS preparation schemes, but not algorithms using classical variational optimization of a staircase ansatz [3–5]. Considering the XXZ ground state specifically, if the state is of low complexity, then it may be possible to prepare it in $\lceil \log_2(\chi = 14) \rceil = 4$ layers of a staircase ansatz [3]. This would correspond to a CNOT depth of 156 and CNOT count of 588, significantly deeper than the AQC circuits. Moreover, if the MPS does not fit this condition, the preparation circuits could be significantly larger. Nevertheless, numerically verifying these estimations is an important future avenue of research, since specific solutions may transpile to much shallower circuits (e.g., where optimized two-qubit blocks become the identity). This would further enhance our understanding of how different variational MPS preparation algorithms exploit structure in normal MPSs.

Regarding scalability, the dominant cost in the AQC workflow arises from the repeated MPS simulation of the target and the ansatz circuit during optimization. Each iteration has a complexity comparable to standard MPS simulation: $O(L\,\chi^2)$ in memory and $O(L\,\chi^3)$ in time [34]. Since the target MPS is classically simulable by construction, and assuming the ansatz remains well conditioned in χ throughout, the optimization remains tractable, with total wall time governed primarily by the number of iterations to convergence. While global convergence cannot be guaranteed in general, it is important to note that the AQC schemes explored here do not impose a minimum fidelity requirement. This flexibility allows AQC to yield practically useful circuits even under aggressive truncation of χ—potentially outperforming alternative methods that either fail to converge or require significantly deeper circuits to achieve comparable fidelity.

Overall, we have demonstrated how the integration of classical tensor networks into quantum algorithms can push the boundary of what can be studied on quantum computers. In a wider scope, we are in an era where experiments on quantum computers regularly probe physics that cannot be exactly classically verified [84,85], inciting the development of tensor network techniques [86,87]. Nevertheless, enormous challenges remain in order to realize a quantum advantage with the combination of utility-scale AQC and quantum hardware. Mapping out this path is of high priority, including identifying systems of interest that exhibit bond dimensions on the order of $\chi > 10^5$ but can be efficiently prepared as quantum circuits. Whether this is possible for 1D systems, such as those exhibiting algebraic topological phases [88], or can only be realized in two or higher dimensions remains an exciting open question.

ACKNOWLEDGMENTS

We thank Niall Robertson and Sergiy Zhuk for their useful discussions and insight. We thank Sebastian Brandhofer for help with error mitigation in the quantum hardware

experiments. This work was supported by the Hartree National Centre for Digital Innovation, a collaboration between the Science and Technology Facilities Council and IBM. A.A. acknowledges support through an Investigator Award from the National Physical Laboratory's Directors' Science and Engineering Fund. L.P.L. and I.R. acknowledge support from the Engineering and Physical Sciences Research Council (Grant No. EP/Y005090/1). A.A., L.P.L, and I.R. acknowledge the support of the UK government Department for Science, Innovation and Technology through the UK National Quantum Technologies Program.

B.J. and J.C. conceptualized the work; B.J., G.P., K.M., L.W.A., and A.A. developed the algorithms; and B.J., G.P., and K.M. conducted the experiments. B.J. and G.P. wrote the original draft of the manuscript. All authors contributed to technical discussions and helped review and edit the manuscript.

DATA AVAILABILITY

The data that support the findings of this article are openly available [47].

APPENDIX A: GRADIENT PAIR SELECTION FOR ADAPT-AQC

When adding a layer, ADAPT-AQC applies a two-qubit unitary to a pair of qubits, with the choice of pair based on the properties of the current state. While previous works have used entanglement-based heuristics [24], here we use a selection process inspired by the ADAPT-VQE algorithm [22] in which the position of the next layer is determined by the gradient $\frac{\partial C}{\partial \vec{\theta}}$ of the cost with respect to the ansatz parameters.

Specifically, any two-qubit unitary $\hat{V}_j^\dagger(\vec{\theta}_j)$, as shown in the ansatz in Fig. 2, can be composed to a basis of CNOT gates and single-qubit Pauli rotation gates. For simplicity in this Appendix, let us redefine the notation of an arbitrary ansatz layer $\hat{V}_j^\dagger(\vec{\theta}_j) = \hat{A}(\vec{\theta}) = \prod \hat{A}_i(\theta_i)$, where $\hat{A}_i$ is either a CNOT gate or a single-qubit rotation of the form: $\hat{A}_i(\theta_i) = e^{-i\theta_i \hat{P}_i}$, where $\hat{P}_i$ is a Pauli gate.

Given the gradient of one of the parametrized gates

$$\frac{\partial}{\partial \theta_i}(e^{-i\theta_i \hat{P}_i})\Big|_{\theta_i=0} = -i\hat{P}_i,$$

the gradient of the ansatz layer with respect to the parameter θ_i follows

$$\begin{aligned}\frac{\partial \hat{A}(\vec{\theta})}{\partial \theta_i}\Big|_{\vec{\theta}=\vec{0}} &= -i\hat{A}_N(0)\cdots\hat{A}_{i+1}(0)\hat{P}_i\hat{A}_{i-1}(0)\cdots\hat{A}_1(0) \\ &:= -i\hat{\mathcal{A}}_i. \end{aligned} \tag{A1}$$

In order to determine which pair of qubits to pick during ADAPT-AQC, the gradient of the cost function with respect to the ansatz parameters is calculated. Rewriting Eq. (1) in terms of the candidate next ansatz layer $\hat{A}(\vec{\theta})$ and the components of Fig. 2, we have

$$C = 1 - |\langle s|\hat{A}(\vec{\theta})|\psi\rangle|^2,$$

where $|\psi\rangle$ is the state produced by the action of all the previous ansatz layers on the target and $|s\rangle = \hat{S}^\dagger|0\rangle$. Then, the gradient of C with respect to one of the variational parameters θ_i is given by

$$\begin{aligned}\frac{\partial C}{\partial \theta_i}\Big|_{\vec{\theta}=\vec{0}} &= \frac{\partial}{\partial \theta_i}(1 - |\langle s|\hat{A}(\vec{\theta})|\psi\rangle|^2)\Big|_{\vec{\theta}=\vec{0}} \\ &= -\frac{\partial}{\partial \theta_i}(\langle s|\hat{A}(\vec{\theta})|\psi\rangle\langle\psi|\hat{A}^\dagger(\vec{\theta})|s\rangle)\Big|_{\vec{\theta}=\vec{0}} \\ &= i(\langle s|\hat{\mathcal{A}}_i|\psi\rangle\langle\psi|\hat{A}^\dagger(\vec{0})|s\rangle - \langle s|\hat{A}(\vec{0})|\psi\rangle\langle\psi|\hat{\mathcal{A}}_i^\dagger|s\rangle).\end{aligned}$$

Since the second term is the complex conjugate of the first, we can combine them to give

$$\frac{\partial C}{\partial \theta_i}\Big|_{\vec{\theta}=\vec{0}} = -2\mathrm{Im}(\langle s|\hat{\mathcal{A}}|\psi\rangle\langle\psi|\hat{A}^\dagger(\vec{0})|s\rangle). \tag{A2}$$

Both of the quantities in Eq. (A2) can be calculated using the MPS simulator used in the rest of the AQC routine.

Finally, the above steps can be used to obtain the location of the next unitary in ADAPT-AQC. Given a candidate two-qubit unitary, Eq. (A2) is computed for each parametrized gate and a gradient vector is constructed

$$\vec{\nabla}C|_{\vec{\theta}=\vec{0}} = \left(\frac{\partial C}{\partial \theta_1}\Big|_{\vec{\theta}=\vec{0}}, \ldots, \frac{\partial C}{\partial \theta_N}\Big|_{\vec{\theta}=\vec{0}}\right). \tag{A3}$$

This procedure is repeated for all possible choices of pairs of qubits, and the pair with the largest gradient magnitude $\|\vec{\nabla}C|_{\vec{\theta}=\vec{0}}\|$ is selected for the layer. Intuitively, this procedure can be viewed as selecting the pair of qubits for which the steepest decrease in the cost function (around $\vec{\theta} = \vec{0}$) is available.

APPENDIX B: MPS CACHING WITH THE QISKIT AER MPS SIMULATOR

Running the ADAPT-AQC algorithm requires obtaining the MPS representation of the quantum circuits shown in Fig. 2. In this work, we specifically use the Qiskit Aer MPS simulator [32] to do this. A useful feature of this simulator is the ability to directly initialize the circuit as a specific MPS, on which gates can be appended. This hybrid MPS-circuit representation can then be evaluated at any point to obtain the final MPS corresponding to the state produced by the action of the gates on the starting MPS.

This enables a significant performance improvement in the evaluation of Fig. 2 by allowing us to locally cache the MPS representation of parts of the circuit once they will no longer be modified. The best example of this is the target circuit MPS $|\psi_t\rangle = \hat{U}|0\rangle$. By simulating the target circuit only once and then caching the resulting MPS, the computational cost of evaluating the full circuit $|\psi_n\rangle$ is significantly reduced in subsequent ADAPT-AQC iterations.

There are also situations where parts of the variational ansatz may no longer be modified during compilation. In order to reduce computational overhead, one could limit the scope of the `rotosolve` optimization step, such that after the unitary $\hat{V}_n^\dagger(\vec{\theta}_n)$ is added, only the previous $1 < i < n$ blocks are optimized. Thus, the parameters of the first $n - i$ unitaries, $\{\vec{\theta}_1, \ldots, \vec{\theta}_{n-i}\}$, would become fixed for the rest of compiling. By contracting the corresponding gates onto the already

TABLE IV. Settings used for the ADAPT-AQC results.

Setting	Value
Pair-picking method	Gradient (Appendix A)
`rotosolve` frequency	1
`rotoselect` tolerance	1×10^{-5}
`rotosolve` tolerance	1×10^{-3}
Starting circuit $\hat{S}$	$\chi = 1$ (Sec. II B 1)
MPS truncation threshold	1×10^{-6}
Coupling map	Nearest-neighbor
Layer ansatz	See Fig. 6

cached target MPS, an state $|\psi_c\rangle = \hat{V}^\dagger_{n-i}(\vec{\theta}_{n-i})...\hat{V}^\dagger_1(\vec{\theta}_1)|\psi_t\rangle$ can be cached, which includes the fixed ansatz layers.

In this work, target caching is always used for ADAPT-AQC experiments. However, we find that the reduced expressibility resulting from caching parts of the ansatz often leads to deeper compiled circuits. Thus, in order to prioritize the shallowest possible solution, ansatz caching was not used to obtain any of the results.

APPENDIX C: DEFAULT ADAPT-AQC AND AQC-TENSOR SETTINGS USED IN EXPERIMENTS

In this Appendix, we provide the configuration options used to obtain the compiled circuits in Sec. III unless otherwise stated.

The ADAPT-AQC settings are shown in Table IV. A further explanation for each setting is as follows:

1. Pair-picking method: the heuristic used for picking the pair of qubits to act on in each iteration.
2. `rotosolve` frequency: the period, defined in terms of the number of layers, between successive applications of the `rotosolve` algorithm. If the index of the layer currently being added is an integer multiple of the `rotosolve` frequency, the parameters of the entire ansatz are optimized.
3. `rotosolve` and `rotoselect` tolerances: the value by which the cost needs to decrease during each optimization cycle to continue the algorithm; see Ref. [48] for more information.
4. Starting circuit: the circuit $\hat{S}$ described in Sec. II B 2.
5. MPS truncation threshold: as defined by the Qiskit Aer MPS simulator [32]. Schmidt coefficients for which the sum of their squares is smaller than this threshold are discarded.
6. Coupling map: the allowed connections in the compiled circuit.
7. Layer ansatz: the form of the two-qubit ansatz used for each layer, represented by $\hat{V}_i^\dagger$ in Sec. II B 2.

The AQC-Tensor settings used are shown in Table V. Here, the number of layers refers to the number of repetitions of the brickwork structure, each consisting of parametrized two-qubit SU(4) unitaries. One layer was used to compile the RMPSs in Sec. III A and three layers were used to compile the XXZ ground state in Sec. III C.

TABLE V. Settings used for the AQC-Tensor results.

Setting	Value
MPS truncation threshold	1×10^{-8}
Optimization method	L-BFGS-B

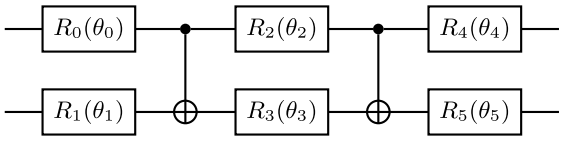


FIG. 6. The two-qubit ansatz block used by ADAPT-AQC in this work. Each $R_i(\theta_i)$ gate is either an R_x, R_y, or R_z Pauli rotation gate, determined during optimization.

APPENDIX D: ADAPT-AQC GROUND-STATE CIRCUIT DIAGRAM

Figure 7 shows the ADAPT-AQC compiled circuit diagram for the $J_z = 2.5$, $h_z = 0.0$ XXZ ground state studied in Sec. III C. The circuit exhibits an interesting structure, sharing similarities with the staircase structure of the Ran method [2] and the brickwork structure of the AQC-Tensor method [8]. Understanding precisely whether there is a link between regions of large staircaselike structures and correlations in the ground state remains as an interesting future research direction for adaptive MPS preparation methods.

APPENDIX E: SCALING OF THE $\chi = 1$ INITIALIZATION SCHEME FOR AQC

In Sec. II B 1, we introduce an initialization scheme for AQC-Tensor. Specifically, for a target MPS $|\psi_t\rangle$, we set the ansatz parameters such that the circuit produces the state $|\psi_{t_{\chi=1}}\rangle$, a $\chi = 1$ compressed approximation of the target state. We subsequently use this strategy across all the AQC-Tensor experiments in this work. Additionally, the same strategy is effectively employed in ADAPT-AQC by setting $\hat{S}^\dagger$ such that $\hat{S}^\dagger|0\rangle = |\psi_{t_{\chi=1}}\rangle$.

In this Appendix, we present analysis on the scalability of this scheme for preparing the ground state of the XXZ Heisenberg Hamiltonian. For $J_z = 2.5$ and $h_z = 0.0$, the same parameters studied in Sec. III C, we compute the ground state with DMRG for $L = 50$ through $L = 300$. For each L, we subsequently obtain $|\psi_{t_{\chi=1}}\rangle$ through calling the TeNPy `compress` function using the variational compression method.

Figure 8 shows how the fidelity $\mathcal{F} = |\langle\psi_{t_{\chi=1}}|\psi_t\rangle|^2$ changes with increasing L. Overall, we see a clear exponential decay, matching known theoretical bounds for representing the gapped phase with product states [89,90]. While this implies that our initialization scheme is not efficient in the asymptotic limit, we note that whether the absolute values are large enough for compiling to be successful is a nuanced question. Answering this fully would require running the preparation algorithms for each MPS of increasing size, a valuable future research direction beyond the scope of this work.

However, to evidence the practical effectiveness of the $\chi = 1$ initialization scheme, we repeat this analysis but now including the alternative strategies of initializing the AQC-Tensor ansatz randomly or to form the identity $\hat{\mathcal{I}}$. The results are shown in the inset of Fig. 8. For $L = 50$, we immediately note

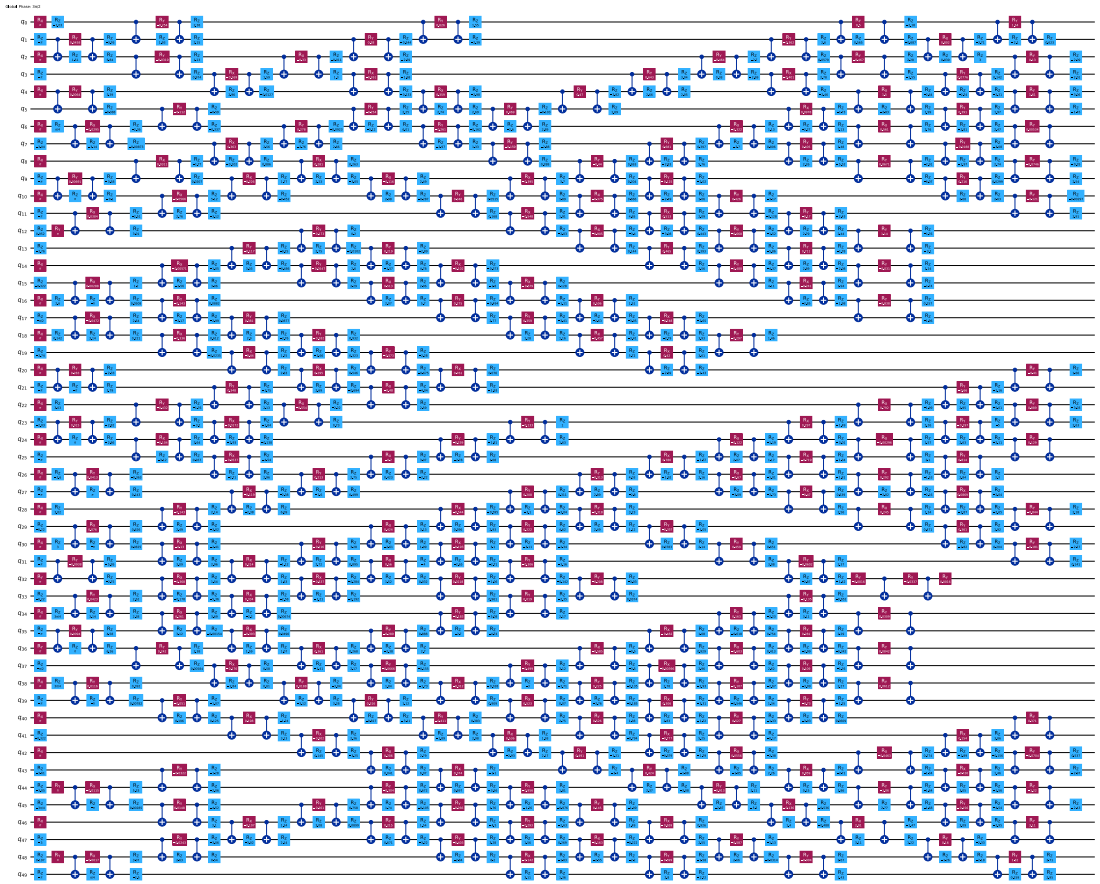


FIG. 7. Compiled $J_z = 2.5$, $h_z = 0.0$ XXZ ground-state circuit produced by ADAPT-AQC. The circuit prepares the ground state to 98.0% fidelity with a CNOT depth and count of 28 and 464, respectively.

the large separation between the $\chi = 1$, random and identity strategies, with fidelities on the order of 10^{-1}, 10^{-15}, and 10^{-221}, respectively. Furthermore, this gap only grows with increasing L. Quantifying this, we perform a linear fit and find the rate of decay for each method to be $-\partial(\log_{10}\mathcal{F})/\partial L = 0.017$, 4.213, and 3.791, respectively. Thus, while all methods decay exponentially, our $\chi = 1$ approach has a significantly favourable scaling, with an exponent over 200 times smaller than both other methods.

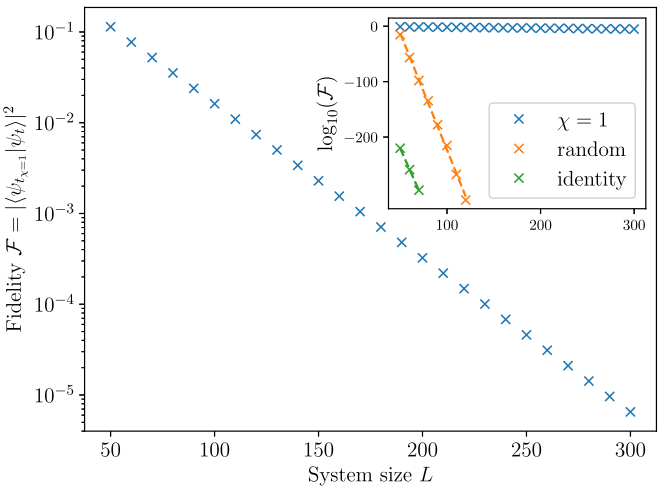


FIG. 8. Fidelity for increasing system size L between $|\psi_t\rangle$, ground state of the XXZ model with $J_z = 2.5$ and $h_z = 0.0$, and $|\psi_{t_{\chi=1}}\rangle$, the compressed $\chi = 1$ approximation of the same state. Inset shows the same data compared to the fidelity when initializing the AQC-Tensor ansatz randomly or as the identity. Dashed lines represent a linear fit of the points, which have gradient $-\partial(\log_{10}\mathcal{F})/\partial L = 0.017$, 4.213, and 3.791, respectively.

APPENDIX F: ERROR MITIGATION

For the real hardware experiments in Sec. III C, a number of error mitigation techniques were used. These were all accessed through using the `EstimatorV2` primitive of Qiskit Runtime [32].

The first two methods used are TREX [80], a form of readout error mitigation and Pauli twirling [81], which tries to reduce noise accumulated as gate errors. We direct the reader to each of their respective references for more information.

The final method employed is ZNE [82], which mitigates errors in an expectation value by purposefully amplifying noise and then extrapolating to the zero-noise limit. In Qiskit, noise amplification is achieved through gate folding, where two-qubit gates $\hat{\mathcal{U}}$ are replaced with sequences of the gate and its inverse $\hat{\mathcal{U}}\hat{\mathcal{U}}^\dagger\hat{\mathcal{U}}$. If every two-qubit gate in the circuit was substituted in this way, this would yield a noise factor of 3. Smaller noise factors and fractional noise factors can also be obtained through probabilistic insertion of the gate identity.

TABLE VI. Details of the gate counts and corresponding zero-noise extrapolation noise factors used for each quench experiment in Sec. III C. For each Trotter step, each entry is of the format (CZ depth, CZ count, [noise factors]) after transpiling for *ibm_fez*.

TS	AQC-Tensor	ADAPT-AQC	Ran
0	(18, 441, [1, 1.3, 1.6, 2])	(28, 464, [1, 1.2, 1.4, 1.6, 1.8])	(157, 409, [1, 1.125, 1.15, 1.175, 1.2])
1	(27, 663, [1, 1.2, 1.4, 1.6, 1.8])	(35, 638, [1, 1.2, 1.4, 1.6, 1.8])	(164, 629, [1, 1.125, 1.15, 1.175, 1.2])
2	(33, 810, [1, 1.2, 1.4, 1.6, 1.8])	(41, 785, [1, 1.05, 1.1, 1.15, 1.2, 1.25, 1.3])	(170, 776, [1, 1.125, 1.15, 1.175, 1.2])
3	(39, 957, [1, 1.05, 1.1, 1.15, 1.2, 1.25, 1.3])	(47, 932, [1, 1.125, 1.15, 1.175, 1.2])	(176, 923, [1, 1.125, 1.15, 1.175, 1.2])
4	(45, 1104, [1, 1.125, 1.15, 1.175, 1.2])	(53, 1079, [1, 1.125, 1.15, 1.175, 1.2])	(182, 1070, [1, 1.125, 1.15, 1.175, 1.2])
5	(51, 1251, [1, 1.125, 1.15, 1.175, 1.2])	(59, 1226, [1, 1.125, 1.15, 1.175, 1.2])	(188, 1217, [1, 1.125, 1.15, 1.175, 1.2])

In our work, we choose the observable to extrapolate to be the $\langle \sigma_i^z \rangle$ expectation value for each qubit. Once this is obtained at each noise factor, a best-fit extrapolation back to zero noise is chosen. Finally, the extrapolated individual $\langle \sigma_i^z \rangle$ values are used to compute an extrapolated SM using Eq. (9).

Choosing sensible noise factors is an important process to achieve good results with ZNE. In general, large noise factors produce better results for shallower circuits and vice versa [91], a trend that we find to be true in our experimentation. For deeper circuits, small noise factors are required, which can lead to unstable results due to the extrapolation distance being large compared to the distance between noise factors. Where we encountered this problem, additional intermediate noise factors were added to try and stabilize the fitting.

Table VI summarizes the ZNE noise factors used for each circuit, alongside the corresponding CZ depth and CZ count. This provides an overview of our general strategy, where once a good set of noise factors was found for a particular CZ depth, the same was used for all circuits of a similar depth. For example, we found that the noise factors [1, 1.125, 1.15, 1.175, 1.2] performed well for deep circuits beyond a CZ depth of approximately 45. Thus, all experiments of the Ran method were ran using this setting.

In general, the best noise factors for each circuit depth were determined through a trial and error process. Starting with the ground-state circuit produced by AQC-Tensor, we leverage its classical simulability to identify a good set of noise factors by comparing the hardware output to the DMRG solution obtained in Sec. III C 1. Importantly, for any realization of the scheme put forward in Fig. 1, benchmarking noise factors for the target circuit against tensor networks is by definition always possible. Furthermore, even for the full circuit in Fig. 1 that cannot be simulated to high accuracy by tensor networks, it is interesting to consider if a low-accuracy approximation could provide a reference point for evaluating noise factors.

The noise factors used for the AQC-Tensor ground state is then used as a starting point for further trial-and-error exploration of the 1 Trotter step circuit. Since the full quench studied in this work can be simulated by combining DMRG and TDVP, we are able to utilize this again to identify good noise factors. This process is repeated for the remaining Trotter step circuits.

For the circuits corresponding to the other MPS preparation methods, we choose their noise factors based on the corresponding AQC-Tensor circuit of similar CZ depth. For example, we note that Trotter step i of ADAPT-AQC has a similar depth to Trotter step $i-1$ of AQC-Tensor and thus uses those noise factors. This same logic is then applied to derive the Ran noise factors. Notably, for noise factors with a smaller range than [1, 1.125, 1.15, 1.175, 1.2], we find that extrapolation becomes unstable, leading to poor results. For this reason, all circuits deeper than 45 CZ depth use these noise factors.

[1] C. Schön, E. Solano, F. Verstraete, J. I. Cirac, and M. M. Wolf, Sequential generation of entangled multiqubit states, Phys. Rev. Lett. **95**, 110503 (2005).

[2] S.-J. Ran, Encoding of matrix product states into quantum circuits of one-and two-qubit gates, Phys. Rev. A **101**, 032310 (2020).

[3] 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).

[4] 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).

[5] B. Anselme Martin, T. Ayral, 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).

[6] 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).

[7] A. Smith, B. Jobst, A. G. Green, and F. Pollmann, Crossing a topological phase transition with a quantum computer, Phys. Rev. Res. **4**, L022020 (2022).

[8] N. Robertson, A. Akhriev, J. Vala, and S. Zhuk, Approximate quantum compiling for quantum simulation: A tensor network based approach, ACM Trans. Quantum Comput. **6**, 1 (2025).

[9] J. R. Garrison, K. Marshall, I. Shehzad, K. J. Sung, C. Johnson, M. Rossmannek, B. Fuller, J. R. Glick, A. Akhriev, S. Zhuk, and N. F. Robertson, Qiskit addon: Approximate quantum compilation with tensor networks (2024), https://github.com/Qiskit/qiskit-addon-aqc-tensor.

[10] M. Lubasch, J. Joo, P. Moinier, M. Kiffner, and D. Jaksch, Variational quantum algorithms for nonlinear problems, Phys. Rev. A **101**, 010301(R) (2020).

[11] A. A. Melnikov, A. A. Termanova, S. V. Dolgov, F. Neukart, and M. Perelshtein, Quantum state preparation using tensor networks, Quantum Sci. Technol. **8**, 035027 (2023).

[12] C. Mc Keever and M. Lubasch, Classically optimized Hamiltonian simulation, Phys. Rev. Res. **5**, 023146 (2023).

[13] T. Jones and S. C. Benjamin, Robust quantum compilation and circuit optimisation via energy minimisation, Quantum **6**, 628 (2022).
[14] K. Sharma, S. Khatri, M. Cerezo, and P. J. Coles, Noise resilience of variational quantum compiling, New J. Phys. **22**, 043006 (2020).
[15] D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, Matrix product state representations, Quantum Inf. Comput. **7**, 401 (2007).
[16] J. I. Cirac, D. Perez-Garcia, N. Schuch, and F. Verstraete, Matrix product states and projected entangled pair states: Concepts, symmetries, theorems, Rev. Mod. Phys. **93**, 045003 (2021).
[17] J. I. Cirac, D. Perez-Garcia, N. Schuch, and F. Verstraete, Matrix product density operators: Renormalization fixed points and boundary theories, Ann. Phys. **378**, 100 (2017).
[18] K. C. Smith, A. Khan, B. K. Clark, S. Girvin, and T.-C. Wei, Constant-depth preparation of matrix product states with adaptive quantum circuits, arXiv:2404.16083.
[19] Z.-Y. Wei, D. Malz, and J. I. Cirac, Efficient adiabatic preparation of tensor network states, Phys. Rev. Res. **5**, L022037 (2023).
[20] D. Malz, G. Styliaris, Z.-Y. Wei, and J. I. Cirac, Preparation of matrix product states with log-depth quantum circuits, Phys. Rev. Lett. **132**, 040404 (2024).
[21] R. Haghshenas, J. Gray, A. C. Potter, and G. K.-L. Chan, Variational power of quantum circuit tensor networks, Phys. Rev. X **12**, 011047 (2022).
[22] 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).
[23] L. Zhu, H. L. Tang, G. S. Barron, F. A. Calderon-Vargas, N. J. Mayhall, E. Barnes, and S. E. Economou, Adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer, Phys. Rev. Res. **4**, 033029 (2022).
[24] B. Jaderberg, A. Agarwal, K. Leonhardt, M. Kiffner, and D. Jaksch, Minimum hardware requirements for hybrid quantum–classical dmft, Quantum Sci. Technol. **5**, 034015 (2020).
[25] T. A. Chowdhury, K. Yu, M. A. Shamim, M. L. Kabir, and R. S. Sufian, Enhancing quantum utility: Simulating large-scale quantum spin chains on superconducting quantum computers, Phys. Rev. Res. **6**, 033107 (2024).
[26] L. B. Oftelie, M. Urbanek, M. Metcalf, J. Carter, A. F. Kemper, and W. A. de Jong, Simulating quantum materials with digital quantum computers, Quantum Sci. Technol. **6**, 043002 (2021).
[27] F. Jamet, L. P. Lindoy, Y. Rath, C. Lenihan, A. Agarwal, E. Fontana, F. Simkovic, B. A. Martin, and I. Rungger, Anderson impurity solver integrating tensor network methods with quantum computing, APL Quantum **2**, 016121 (2025).
[28] S. Khatri, R. LaRose, A. Poremba, L. Cincio, A. T. Sornborger, and P. J. Coles, Quantum-assisted quantum compiling, Quantum **3**, 140 (2019).
[29] M. Otten, C. L. Cortes, and S. K. Gray, Noise-resilient quantum dynamics using symmetry-preserving ansatzes, arXiv:1910.06284.
[30] S. Barison, F. Vicentini, and G. Carleo, An efficient quantum algorithm for the time evolution of parameterized circuits, Quantum **5**, 512 (2021).
[31] N. F. Berthusen, T. V. Trevisan, T. Iadecola, and P. P. Orth, Quantum dynamics simulations beyond the coherence time on noisy intermediate-scale quantum hardware by variational trotter compression, Phys. Rev. Res. **4**, 023097 (2022).
[32] 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.
[33] S. R. White, Density matrix formulation for quantum renormalization groups, Phys. Rev. Lett. **69**, 2863 (1992).
[34] U. Schollwöck, The density-matrix renormalization group in the age of matrix product states, Ann. Phys. **326**, 96 (2011).
[35] N. F. Robertson, A. Akhriev, J. Vala, and S. Zhuk, Escaping barren plateaus in approximate quantum compiling, arXiv:2210.09191.
[36] J. Zhang, J. Vala, S. Sastry, and K. B. Whaley, Geometric theory of nonlocal two-qubit operations, Phys. Rev. A **67**, 042313 (2003).
[37] C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal, Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization, ACM Trans. Math. Software **23**, 550 (1997).
[38] D. P. Kingma, Adam: A method for stochastic optimization, arXiv:1412.6980.
[39] R. Puig, M. Drudis, S. Thanasilp, and Z. Holmes, Variational quantum simulation: A case study for understanding warm starts, PRX Quantum **6**, 010317 (2025).
[40] Y. Wang, B. Qi, C. Ferrie, and D. Dong, Trainability enhancement of parameterized quantum circuits via reduced-domain parameter initialization, Phys. Rev. Appl. **22**, 054005 (2024).
[41] C.-Y. Park and N. Killoran, Hamiltonian variational ansatz without barren plateaus, Quantum **8**, 1239 (2024).
[42] N. Fitzpatrick, H. Apel, and D. M. Ramo, Evaluating low-depth quantum algorithms for time evolution on fermion-boson systems, arXiv:2106.03985.
[43] A. Haukisalmi, M. Raasakka, and I. Tittonen, Comparing resource requirements of noisy quantum simulation algorithms for the Tavis-Cummings model, arXiv:2402.16692.
[44] B. Jaderberg, L. W. Anderson, W. Xie, S. Albanie, M. Kiffner, and D. Jaksch, Quantum self-supervised learning, Quantum Sci. Technol. **7**, 035005 (2022).
[45] B. Jaderberg, A. Eisfeld, D. Jaksch, and S. Mostame, Recompilation-enhanced simulation of electron–phonon dynamics on IBM quantum computers, New J. Phys. **24**, 093017 (2022).
[46] B. Jaderberg, Solving optimisation problems on near-term quantum computers, Ph.D. thesis, University of Oxford, 2022.
[47] B. Jaderberg, G. Pennington, K. V. Marshall, L. W. Anderson, A. Agarwal, L. P. Lindoy, I. Rungger, S. Mensa, and J. Crain, Adaptive approximate quantum compiling (adapt-aqc) (2025), https://github.com/qiskit-community/adapt-aqc/.
[48] M. Ostaszewski, E. Grant, and M. Benedetti, Structure optimization for parameterized quantum circuits, Quantum **5**, 391 (2021).
[49] D. Haag, F. Baccari, and G. Styliaris, Typical correlation length of sequentially generated tensor network states, PRX Quantum **4**, 030330 (2023).
[50] C. Lancien and D. Pérez-García, Correlation length in random MPS and PEPS, Ann. Henri Poincaré **23**, 141 (2022).
[51] S. Garnerone, T. R. de Oliveira, S. Haas, and P. Zanardi, Statistical properties of random matrix product states, Phys. Rev. A **82**, 052312 (2010).

[52] D. A. Millar, G. W. Pennington, N. T. M. Siow, and S. J. Thomson, Qiskit-community/mps-to-circuit, Zenodo (2025), https://doi.org/10.5281/zenodo.14920028.

[53] L. Madden, A. Akhriev, and A. Simonetto, Sketching the best approximate quantum compiling problem, in *Proceedings of the 2022 IEEE International Conference on Quantum Computing and Engineering (QCE)* (IEEE, Broomfield, CO, USA, 2022), pp. 492–502.

[54] L. Madden and A. Simonetto, Best approximate quantum compiling problems, ACM Trans. Quantum Comput. **3**, 1 (2022).

[55] S. Hammes-Schiffer and A. A. Stuchebrukhov, Theory of coupled electron and proton transfer reactions, Chem. Rev. **110**, 6939 (2010).

[56] A. J. Kirby, Efficiency of proton transfer catalysis in models and enzymes, Acc. Chem. Res. **30**, 290 (1997).

[57] S. Hammes-Schiffer, Nuclear–electronic orbital methods: Foundations and prospects, J. Chem. Phys. **155**, 030901 (2021).

[58] T. Hammer and U. Manthe, Intramolecular proton transfer in malonaldehyde: Accurate multilayer multi-configurational time-dependent Hartree calculations, J. Chem. Phys. **134**, 224305 (2011).

[59] M. Schröder, F. Gatti, and H.-D. Meyer, Theoretical studies of the tunneling splitting of malonaldehyde using the multiconfiguration time-dependent Hartree approach, J. Chem. Phys. **134**, 234307 (2011).

[60] S. P. Webb, T. Iordanov, and S. Hammes-Schiffer, Multiconfigurational nuclear-electronic orbital approach: Incorporation of nuclear quantum effects in electronic structure calculations, J. Chem. Phys. **117**, 4106 (2002).

[61] F. Pavošević, T. Culpitt, and S. Hammes-Schiffer, Multicomponent quantum chemistry: Integrating electronic and nuclear quantum effects via the nuclear–electronic orbital method, Chem. Rev. **120**, 4222 (2020).

[62] A. Kovyrshin, M. Skogh, L. Tornberg, A. Broo, S. Mensa, E. Sahin, B. C. Symons, J. Crain, and I. Tavernelli, Nonadiabatic nuclear–electron dynamics: A quantum computing approach, J. Phys. Chem. Lett. **14**, 7065 (2023).

[63] A. Kovyrshin, D. Manawadu, E. Altamura, G. Pennington, B. Jaderberg, S. Brandhofer, A. Nykänen, A. Miller, W. Talarico, S. Knecht, F. Pavošević, A. Baiardi, F. Tacchino, I. Tavernelli, S. Mensa, J. Crain, L. Tornberg, and A. Broo, Approximate quantum circuit compilation for proton-transfer kinetics on quantum processors, arXiv:2507.08996.

[64] C. Sosa, J. Geertsen, G. W. Trucks, R. J. Bartlett, and J. A. Franz, Selection of the reduced virtual space for correlated calculations. An application to the energy and dipole moment of H_2O, Chem. Phys. Lett. **159**, 148 (1989).

[65] A. Nykänen, A. Miller, W. Talarico, S. Knecht, A. Kovyrshin, M. Skogh, L. Tornberg, A. Broo, S. Mensa, B. C. Symons, *et al.* Toward accurate post-Born–Oppenheimer molecular simulations on quantum computers: An adaptive variational eigensolver with nuclear-electronic frozen natural orbitals, J. Chem. Theory Comput. **19**, 9269 (2023).

[66] B. G. Levine, A. S. Durden, M. P. Esch, F. Liang, and Y. Shu, CAS without SCF—why to use CASCI and where to get the orbitals, J. Chem. Phys. **154**, 090902 (2021).

[67] A. Kovyrshin, D. Manawadu, E. Altamura, G. Pennington, B. Jaderberg, S. Brandhofer, A. Nykänen, A. Miller, W. Talarico, S. Knecht, *et al.*, Approximate quantum circuit compilation for proton-transfer kinetics on quantum processors (2025), https://github.com/stfc/quantum-neo-dynamics.

[68] A. Mitra, Quantum quench dynamics, Annu. Rev. Condens. Matter Phys. **9**, 245 (2018).

[69] P. Calabrese, Fabian H. L. Essler, and M. Fagotti, Quantum quench in the transverse-field Ising chain, Phys. Rev. Lett. **106**, 227203 (2011).

[70] A. Agarwal, M. Hughes, and J. Mur-Petit, Phase diagram and post-quench dynamics in a double spin-chain system in transverse fields, arXiv:2111.07831.

[71] P. Barmettler, M. Punk, V. Gritsev, E. Demler, and E. Altman, Relaxation of antiferromagnetic order in spin-1/2 chains following a quantum quench, Phys. Rev. Lett. **102**, 130603 (2009).

[72] M. Fagotti, M. Collura, F. H. L. Essler, and P. Calabrese, Relaxation after quantum quenches in the spin-$\frac{1}{2}$ Heisenberg XXZ chain, Phys. Rev. B **89**, 125101 (2014).

[73] P. Barmettler, M. Punk, V. Gritsev, E. Demler, and E. Altman, Quantum quenches in the anisotropic spin-Heisenberg chain: Different approaches to many-body dynamics far from equilibrium, New J. Phys. **12**, 055017 (2010).

[74] F. B. Ramos, A. Urichuk, I. Schneider, and J. Sirker, Power-law decay of correlations after a global quench in the massive XXZ chain, Phys. Rev. B **107**, 075138 (2023).

[75] J. Hauschild *et al.*, Tensor network python (tenpy) version 1, SciPost Phys. Codebases 41 (2024).

[76] M. V. Rakov, M. Weyrauch, and B. Braiorr-Orrs, Symmetries and entanglement in the one-dimensional spin-$\frac{1}{2}$ XXZ model, Phys. Rev. B **93**, 054417 (2016).

[77] IBM Quantum Platform (2024), https://quantum-computing.ibm.com/.

[78] E. Grant, L. Wossnig, M. Ostaszewski, and M. Benedetti, An initialization strategy for addressing barren plateaus in parametrized quantum circuits, Quantum **3**, 214 (2019).

[79] M. Suzuki, Generalized Trotter's formula and systematic approximants of exponential operators and inner derivations with applications to many-body problems, Commun. Math. Phys. **51**, 183 (1976).

[80] E. van den Berg, Z. K. Minev, and K. Temme, Model-free readout-error mitigation for quantum expectation values, Phys. Rev. A **105**, 032620 (2022).

[81] J. J. Wallman and J. Emerson, Noise tailoring for scalable quantum computation via randomized compiling, Phys. Rev. A **94**, 052325 (2016).

[82] K. Temme, S. Bravyi, and J. M. Gambetta, Error mitigation for short-depth quantum circuits, Phys. Rev. Lett. **119**, 180509 (2017).

[83] J. Haegeman, J. I. Cirac, T. J. Osborne, I. Pižorn, H. Verschelde, and F. Verstraete, Time-dependent variational principle for quantum lattices, Phys. Rev. Lett. **107**, 070601 (2011).

[84] Y. Kim, A. Eddins, S. Anand, K. X. Wei, E. Van Den Berg, S. Rosenblatt, H. Nayfeh, Y. Wu, M. Zaletel, K. Temme, *et al.*, Evidence for the utility of quantum computing before fault tolerance, Nature (London) **618**, 500 (2023).

[85] S. Anand, K. Temme, A. Kandala, and M. Zaletel, Classical benchmarking of zero noise extrapolation beyond the exactly-verifiable regime, arXiv:2306.17839.

[86] J. Tindall and M. Fishman, Gauging tensor networks with belief propagation, SciPost Phys. **15**, 222 (2023).

[87] J. Tindall, M. Fishman, E. M. Stoudenmire, and D. Sels, Efficient tensor network simulation of IBM's eagle kicked Ising experiment, PRX Quantum **5**, 010308 (2024).

[88] S. Yang, H.-Q. Lin, and X.-J. Yu, Gapless topological behaviors in a long-range quantum spin chain, Commun. Phys. **8**, 27 (2025).

[89] B. Pozsgay, Overlaps between eigenstates of the XXZ spin-1/2 chain and a class of simple product states, J. Stat. Mech.: Theory Exp. (2014) P06011.

[90] M. Brockmann and J.-M. Stéphan, Universal terms in the overlap of the ground state of the spin-1/2 XXZ chain with the Néel state, J. Phys. A: Math. Theor. **50**, 354001 (2017).

[91] R. Majumdar, P. Rivero, F. Metz, A. Hasan, and D. S. Wang, Best practices for quantum error mitigation with digital zero-noise extrapolation, in *Proceedings of the 2023 IEEE International Conference on Quantum Computing and Engineering (QCE)* (IEEE, Bellevue, WA, USA, 2023), Vol. 1, pp. 881–887.