

Automated detection of symmetry-protected subspaces in quantum simulations

Caleb Rotello^{1,2,*}, Eric B. Jones^{3,2,†}, Peter Graf^{2,‡}, and Eliot Kapit^{1,§}¹Department of Physics and Quantum Engineering Program, Colorado School of Mines, Golden Colorado 80401, USA²National Renewable Energy Laboratory, Golden Colorado 80401, USA³Infleqtion, Louisville Colorado 80027, USA

(Received 16 February 2023; accepted 7 July 2023; published 4 August 2023)

The analysis of symmetry in quantum systems is of utmost theoretical importance, useful in a variety of applications and experimental settings, and difficult to accomplish in general. Symmetries imply conservation laws, which partition Hilbert space into invariant subspaces of the time-evolution operator, each of which is demarcated according to its conserved quantity. We show that, starting from a chosen basis, any invariant, symmetry-protected subspaces which are diagonal in that basis are discoverable using transitive closure on graphs representing state-to-state transitions under k -local unitary operations. Importantly, the discovery of these subspaces relies neither upon the explicit identification of a symmetry operator or its eigenvalues nor upon the construction of matrices of the full Hilbert space dimension. We introduce two classical algorithms, which efficiently compute and elucidate features of these subspaces. The first algorithm explores the entire symmetry-protected subspace of an initial state in time complexity linear to the size of the subspace by closing local basis state-to-basis state transitions. The second algorithm determines, with bounded error, if a given measurement outcome of a dynamically generated state is within the symmetry-protected subspace of the state in which the dynamical system is initialized. We demonstrate the applicability of these algorithms by performing postselection on data generated from emulated noisy quantum simulations of three different dynamical systems: the Heisenberg-XXX model and the T_6 and F_4 quantum cellular automata. Due to their efficient computability and indifference to identifying the underlying symmetry, these algorithms lend themselves to the postselection of quantum computer data, optimized classical simulation of quantum systems, and discovery of previously hidden symmetries in quantum mechanical systems.

DOI: [10.1103/PhysRevResearch.5.033082](https://doi.org/10.1103/PhysRevResearch.5.033082)

I. INTRODUCTION

The analysis of symmetry is a central tool in physics and has enabled some of the most profound discoveries in the field. Noether's theorem famously connects the symmetries of a system's action with conservation laws to which that system's equations of motion are subject [1]. Generally, the analysis of symmetry, or the breaking thereof, allows one to constrain theories [2], solve equations of motion more efficiently [3], and identify phases of matter [4]. Applications of symmetry analysis in quantum information include, but are not limited to, quantum error correction [5], error mitigation

on quantum hardware [6–10], and quantum machine learning model design [11].

Quantum computing can efficiently simulate quantum dynamics in regimes where classical simulation becomes impossible [12]. However, current quantum processors operate in a regime severely constrained by noise, with error rates not yet sufficiently below most error correction thresholds [13]. Error mitigation will therefore be critical in the interim before fault-tolerant architectures can be scaled [14–16]. In recent work, and despite its limitations [17], the technique of *postselection* has proven useful to mitigate errors and extract useful results from quantum simulation experiments (see, e.g., [18–20]). Postselection works by identifying measured states that could only have come from error processes and excluding them from the statistics used to calculate output quantities. The most obvious example is a conserved quantity such as particle number. In such a case, any measured state that does not preserve the conserved quantity must be the result of errors. Due to the connection established by Noether, a more fundamental way to describe postselection is with respect to symmetry. According to this description, postselection works by checking the eigenvalue of the simulation's "fiducial" (i.e., initial) state under the symmetry operator against the corresponding symmetry operator eigenvalues (e.g., value of a conserved quantity) of individual measurement results in the dynamically generated output state of the simulation. If a

*caleb.rotello@nrel.gov

†eric.jones@coldquanta.com; eric.jones@infleqtion.com

‡Peter.Graf@nrel.gov

§ekapit@mines.edu

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI. Open access publication funded by the National Renewable Energy Laboratory (NREL) Library, part of a national laboratory of the U.S. Department of Energy.

particular measurement outcome registers a different eigenvalue under the symmetry operator than the fiducial state does, then the measurement is a result of error and can be discarded. This procedure is restricted in the scope of its application, as symmetries of a quantum system and their corresponding operators (i.e., conserved quantities) are typically either engineered into the dynamics “by hand” or identified by clever theoretical intuition.

While many physically relevant systems have well-characterized symmetries, for a generic quantum system the relevant symmetry operator(s) may not be obvious *a priori* and may be difficult to identify. In this paper we treat quantum cellular automata as one such case. Additionally, recent research has revealed that symmetries of Floquet or discrete time evolution can be different from symmetries in continuous time evolution of the same Hamiltonian [21–24], which further promotes the development of an automated technique for identifying such symmetries. Various attempts to discover symmetry have been pursued via density matrix methods [25], brute force algorithms such as Algorithm 2 in [26], and data-driven methods [27–29].

Being able to perform postselection in a manner that does not require explicit identification of a symmetry operator would greatly increase the technique’s applicability; this is the subject of this paper. As a corollary, such an operator-free method for error detection also enables additional applications such as more efficient classical simulation of quantum systems via computational basis state reduction. For example, particle number conservation in hardcore boson models can be used both for postselection in quantum simulation and for reducing the basis state set size from 2^n to $\binom{n}{N}$, where n is the number of lattice sites and N the number of particles, in classical simulations [30]. Interestingly, the identification of symmetry or conserved quantities in some instances can make classical simulation so efficient that it can obviate the need for quantum computation altogether [31]. Finally, in certain special cases, one may be able to infer the explicit form of a symmetry operator by inspection of the reduced basis set.

In this paper we provide algorithms to efficiently make use of symmetry in an operator-free manner. To do this, our methods create the subspace of measurement basis states which would share a conserved quantity of some commuting symmetry operator that is diagonal in that measurement basis, without needing to explicitly create that operator. We call such a space a *symmetry-protected subspace* (SPS). In the language of linear algebra, these are *invariant subspaces* of the evolution operator; they are subspaces, determined by the initial state, from which the evolution cannot escape. To reap the benefits of symmetry we need to find only the SPS of the initial state, not a conserved quantity, much less an explicit symmetry. However, naively, to find an SPS we need to actually evolve the system in the full Hilbert space, which is exponentially large in the number of qubits (particles, spins, etc.). Sections III through V describe the formulation and algorithms by which we avoid this exponential scaling, but here we provide a nontechnical overview.

First, note that most Hamiltonians and resulting unitary evolution operators are built from a number of *local* operators. For example, the Heisenberg-XXX model described below consists only of nearest-neighbor interactions. So, at some

level, we have an intuition that the dynamics, thus the SPSs, should be derivable, like the unitary operator itself, from a combination of local operations, and that local operations are inherently less computationally expensive to work with. This is indeed the case, as shown below. With this in mind, we focus on unitary operators which are composed of a tractable number of local operators.

Next, note that to say that a wave function is *in* a symmetry-protected or invariant subspace is to say that it is and remains throughout dynamic evolution a linear combination of basis states in that subspace and that subspace alone. And if we care only about finding the subspace, we do not need to keep track of the actual linear combination (i.e., both the basis vectors and their amplitudes) but only the basis vectors. This “binarization” of the evolution is critical, because it allows us to adopt a graph-theoretic framework that is vastly more efficient for finding and searching SPSs.

This also leads to an important restriction in our work; using our methods, we can automate only the discovery of symmetry which is diagonal in a chosen basis. We work in the computational Z basis throughout this paper, though extensions to other bases are of course possible by rewriting the time evolution operators in the new basis and performing the same procedure we describe below. Our automated methods should thus be viewed as a tool to find symmetry-protected subspaces within a given basis (if, of course, they exist), and to potentially improve classical and noisy quantum simulations based on those discovered subspaces. However, they still require an intelligently guessed initial basis as a starting point. For the problems we consider in this work, the computational basis is sufficient to derive novel results, though more complex choices can be required in other cases.

With this caveat in mind, once the basis is chosen we create an undirected and unweighted graph, called the state interaction graph, which describes all possible state-to-state transitions over a single application of a unitary evolution operator. The transitive closure of this graph fragments the Hilbert space of the system, represented in a particular measurement basis, into a cluster graph, whose subgraphs are each a symmetry-protected subspace.

Our main results are two classical algorithms that efficiently construct and work within these subspaces. Both algorithms require the construction of a mathematical object which we call the *basis state string edit map*, which efficiently describes the interactions of basis vectors through a unitary operator by examining only edits to the bit strings which represent those basis vectors; this procedure is described for a single local operator in Algorithm III.1. The first, Algorithm IV.1, uses “transitive closure” on local operations to explore and explicitly construct the full SPS of an initial state, which enables the partition of the Hilbert space into a set of disjoint SPSs. This algorithm scales linearly in both the number of local operators from which the global operator is constructed and the size of the SPS, which is a huge improvement over the exponential scaling of the naive “full evolution” approach. However, because the SPS itself can be exponentially large (albeit with an asymptotically smaller prefactor), the second algorithm, Algorithm V.1, finds a *path* of local operations through a set of SPS graphs from an initial to final (i.e., measured) state to determine if they lie within the *same* SPS

(and thus the final state is valid). This algorithm scales as the number of local operations raised to a small integer power (that can be tuned for accuracy and performance) times the length of the path, thus completely eliminating any exponential scaling.

This paper is structured as follows. In Sec. II we define symmetry-protected subspaces and the three quantum systems that will serve as our “exemplars” throughout. In Sec. III we outline our novel graph theoretical approach to quantum simulations; in practice, Secs. III A–III B do not need to be applied, as they are only theoretical foundations for our work, while Sec. III C describes the construction of the aforementioned basis state string edit map. Section IV discusses the algorithm for computing an entire symmetry-protected subspace. Section V provides a more efficient algorithm to verify if two states exist within the same symmetry-protected subspace. For the reader primarily interested in applying the algorithms, Sections III C, IV, and V are the most relevant. Finally, we demonstrate the effectiveness of symmetry-protected subspaces to mitigate error in quantum simulations on a classical emulator in Sec. VI.

II. PRELIMINARIES

A. Symmetry-protected subspaces

Consider a quantum system undergoing unitary evolution according to the operator $U(t)$ for a time t . The operator $U(t)$ can represent continuous time evolution, but also includes other cases, such as discrete time evolution. We say that $U(t)$ is invariant under the action of an operator S if $[S, U(t)] = 0$ for all times t . In this instance, S is a symmetry operator. For a basis in which S is diagonal, states can be labeled by their eigenvalues under the action of S : $S|s, b\rangle = s|s, b\rangle$, where b is some other (set of) label(s), which could represent, for example, a computational basis state integer encoding or a many-body eigenvalue. Suppose now that we initialize the dynamics in a state of definite s : $|\psi_0^s\rangle = \sum_b \alpha_b^s |s, b\rangle$, and evolve under $U(t)$. Given the commutativity of the symmetry and evolution operators, the action of the symmetry operator on the output state is $S[U(t)|\psi_0^s\rangle] = U(t)S|\psi_0^s\rangle = U(t)s|\psi_0^s\rangle = s[U(t)|\psi_0^s\rangle]$, indicating that the eigenvalue s is conserved under the evolution for all time. Our methods identify states that would share an eigenvalue under S without explicitly knowing S . To do this, we use the notion of a symmetry-protected subspace, also known as an invariant subspace, which we define below.

Definition 1 (Symmetry-Protected Subspace). Let \mathcal{H}_d be a Hilbert space of dimension d spanned by an orthonormal set of basis vectors, $B(\mathcal{H}_d) = \{|b\rangle\}$. A subspace $G \subseteq \mathcal{H}_d$, which is spanned by a subset of $B(\mathcal{H}_d)$, is a symmetry-protected subspace of unitary operator $U(t)$ if and only if a projection onto G , $P_G = \sum_{b \in G} |b\rangle\langle b|$, obeys the commutation relation $[P_G, U(t)] = 0$.

Note that while our definition emphasizes the connection to symmetries, an SPS, as defined above, is indeed an invariant subspace according to the usual definition [32], which is simply that $\forall |g\rangle \in G, U(t)|g\rangle \in G$, because if $|g\rangle \in G$,

$$U(t)|g\rangle = U(t)P_G|g\rangle = P_G[U(t)|g\rangle] \in G, \quad (1)$$

where we have used the commutativity of P_G and $U(t)$ and the fact that by definition the result of applying P_G to anything

is in G . Another consequence of Def. 1 is that if a particular basis state is not in G , then the transition matrix element to that state from any state in G is strictly zero.

Lemma 0.1. Let $|g\rangle \in G$ be an arbitrary element of a symmetry-protected (i.e., invariant) subspace G of an evolution operator $U(t)$ and let $|b\rangle \in B(\mathcal{H}_d)$ be a basis vector outside of G , $|b\rangle \notin G$. Then $\langle b|U(t)|g\rangle = 0$ for any time t .

Proof. Using Def. 1, $\langle b|U(t)|g\rangle = \langle b|U(t)P_G|g\rangle = \langle b|P_GU(t)|g\rangle = [P_G|b\rangle]^\dagger U(t)|g\rangle = 0$, since the projection operator onto G annihilates states outside of G .

In Sec. IV we present an algorithm that prescriptively constructs a subspace, denoted $G_{|\psi_0\rangle}$, of a particular initial state, $|\psi_0\rangle$. We also show in the corresponding theorem, Thm. 1, that subspaces so constructed share the property described in Lem. 0.1, which relies on Def. 1 for its proof. Hence, $G_{|\psi_0\rangle}$ constructed according to the procedure in Sec. IV are symmetry-protected subspaces.

Similar to that of the symmetry operator S , the commutation relation involving P_G also identifies a dynamical invariance, since for any $|g\rangle \in G$, $P_G[U(t)|g\rangle] = [U(t)|g\rangle]$. That is, the state $U(t)|g\rangle$ is an eigenvector of the projector with eigenvalue 1. Therefore, while the projector P_G does not identify conserved symmetry eigenvalues, it does indicate when such an eigenvalue exists. In Sec. III, we will show how discovering P_G , rather than S directly, empowers our algorithms to discover underlying invariances.

B. Quantum simulations

The general form for unitary evolution operators we assume is

$$U(t) = \mathcal{O}_{op} \left[\prod_{j=1}^p \prod_{i=1}^m U_i(\tau_j) \right], \quad (2)$$

where t is the total duration of evolution, \mathcal{O}_{op} denotes some operator ordering, such as time ordering, and p is the number of time steps used to evolve to t . Each $U_i(\tau_j)$ acts locally on k -qubits and is parameterized by a real time coordinate τ_j , with m k -local operators used to describe the n -qubit system. Evolution unitaries of the form in Eq. (2) can evolve over discrete time, where each τ_j is finite, or discretized-continuous time, where each τ_j is (ideally infinitesimally) small in order to minimize Trotter error.

In the instance where a k -local Hamiltonian, $H = \sum_i h_i$, is known, Eq. (2) results from the local dynamics governed by the h_i via the Trotter-Suzuki formula [33], and p corresponds to the number of Trotter steps. We refer to the operator U as the relevant quantum system and examine the dynamics and associated symmetry-protected subspaces of three exemplary systems.

1. Heisenberg-XXX

The one-dimensional Heisenberg-XXX model for n spin-1/2 particles with nearest-neighbor interactions is given by the following Hamiltonian:

$$H = \sum_{i=0}^{n-2} X_i X_{i+1} + Y_i Y_{i+1} + Z_i Z_{i+1}, \quad (3)$$

where X_i , Y_i , and Z_i are Pauli operator acting on spin i . The model conserves total spin in the Z basis, represented by the operator

$$S^z = \sum_{i=0}^{n-1} Z_i, \quad (4)$$

as well as the correspondingly defined operators S^x and S^y . Quantum simulation of the Heisenberg model on a digital quantum processor can be achieved via exponentiation and Trotterization of Eq. (3). In such quantum simulation experiments, one usually picks one or a number of qubit bases in which to measure. The symmetry operators $S^{x,y,z}$ can be used to mitigate errors in the all-qubit X , Y , Z measurement bases, respectively, via postselection. In the context of classical simulation of the XXX model, the symmetries can be used to constrain the number of basis states included in the dynamics.

2. T_6 quantum cellular automata

The one-dimensional T_6 quantum cellular automata (QCA) rule has recently come to interest within the context of quantum complexity science as a dynamical small-world mutual information network generator [20] and a QCA Goldilocks rule [34]. Its discrete-time unitary update can be derived from a parent Hamiltonian, but it is more natural to define the system by specifying the simulation unitary for a discrete time $t = p$, directly:

$$U(T_6; t) = \mathcal{O}_{op}^{time} \left[\prod_{j=1}^p \prod_{i=3,5,\dots}^{n-1} U_i(\tau_j) \prod_{i=2,4,\dots}^{n-1} U_i(\tau_j) \right],$$

$$U_i(\tau_j) = \sum_{\alpha, \beta=0}^1 P_{i-1}^{(\alpha)} \otimes (H_i)^{\delta_{\alpha+\beta,1}} \otimes P_{i+1}^{(\beta)}, \quad (5)$$

where $P_i^{(\alpha)} = |\alpha_i\rangle\langle\alpha_i|$ for $\alpha = 0, 1$ is the projection operator onto the corresponding state of qubit i , H_i is the Hadamard operator, and $\delta_{\alpha+\beta,1}$ is the Kronecker delta function. At each time step, a Hadamard is applied to a qubit only if exactly one of its neighbors is in the $|1\rangle$ state (i.e., $\alpha + \beta = 1$) and does nothing otherwise. It has a known Z basis symmetry related to domain-wall conservation:

$$S = \sum_{i=0}^n Z_i Z_{i+1}, \quad (6)$$

where it should be understood that indices $i = 1, \dots, n$ refer to dynamical, computational qubits while the indices $i = 0$ and $i = n + 1$ refer to nondynamical qubits fixed to the $|0\rangle$ state.

3. F_4 quantum cellular automata

The one-dimensional F_4 QCA with nearest- and next-nearest-neighbor connectivity is another Goldilocks rule [34]. It is also most easily specified by its simulation unitary for discrete-time duration $t = p$:

$$U(F_4; t) = \mathcal{O}_{op}^{time} \left[\prod_{j=1}^p U(\tau_j) \right], \quad (7)$$

where if the time step index, j , is even

$$U(\tau_{j=\text{even}}) = \prod_{i=2,5,8,\dots} U_i(\tau_j) \prod_{i=3,6,9,\dots} U_i(\tau_j) \prod_{i=4,7,10,\dots} U_i(\tau_j), \quad (8)$$

and if j is odd, then

$$U(\tau_{j=\text{odd}}) = \prod_{i=3,6,9,\dots} U_i(\tau_j) \prod_{i=2,5,8,\dots} U_i(\tau_j) \prod_{i=4,7,10,\dots} U_i(\tau_j). \quad (9)$$

In either case,

$$U_i(\tau_j) = \sum_{\alpha, \beta, \gamma, \omega=0}^1 P_{i-2}^{\alpha} P_{i-1}^{\beta} (H_i)^{\delta_{\alpha+\beta+\gamma+\omega,2}} P_{i+1}^{\gamma} P_{i+2}^{\omega}. \quad (10)$$

Equation (10) applies a Hadamard to a qubit if exactly two out of its neighbors or next-nearest neighbors are in the $|1\rangle$ state. There are no analytically known symmetries for this rule. As shown in Fig. 3 in Sec. IV C, our methods discover previously unknown, symmetry-protected subspaces, indicating a hitherto hidden symmetry of the system.

III. GRAPH THEORY APPROACH TO QUANTUM SIMULATIONS

In this section we show how graph theory coupled with transitive closure discovers symmetry-protected subspaces. We describe how to “binarize” interactions between basis states through the dynamical system by discarding amplitude and phases to solely highlight *if* such an interaction exists, describing these interactions as transitive relations, and finding the states connected by a transitive relation to form the SPSs of the system.

Directly, this method still requires unitary matrix multiplication in Hilbert space to establish basis state interactions through the dynamical system. Thus, to apply our methods, we need an efficient way to describe state-to-state interactions. We do this by creating a structure we call a *string edit map* \mathcal{L}_U for a unitary operator U , which relies on the observation that quantum systems are typically structured by local interactions. This string edit map returns the basis states available through any operations of a single local unitary operator on a single basis state in near-constant time complexity, allowing us to inexpensively find basis vectors available in local quantum dynamics.

This section will proceed as follows: in Sec. III A we first define the concept of a state interaction graph, whose edges indicate nonzero amplitudes on transitions between measurement basis states in the quantum simulation. Next, in Sec. III B we show how transitive closure on this graph creates a cluster graph, which we call the closed state interaction graph, whose complete subgraphs are symmetry-protected subspaces. Finally, in Sec. III C we show how this closed interaction graph can efficiently return the set of all measurement basis states seen by a unitary operator on a single state via the construction of the string edit map \mathcal{L}_U for a unitary operator U .

A. State interaction graph

We begin by defining the state interaction graph D_U for a unitary operator U over a Hilbert space basis $B(\mathcal{H}_d) = \{|b\rangle\}$

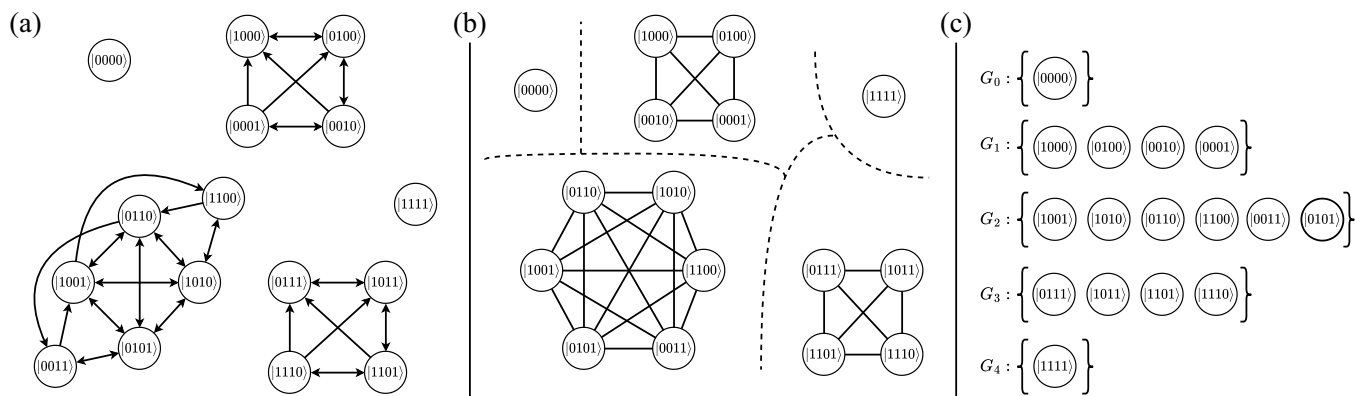


FIG. 1. Example construction of the state interaction graph $D_{U_{Trot}}$ and corresponding symmetry-protected subspaces for the one-dimensional, four-site hopping unitary in Eq. (12). Multiqubit states are ordered as $|q_0q_1q_2q_3\rangle$. One Trotter approximation for the unitary, U_{Trot} , is given in Eq. (13), where it should be understood that right-most operators act first. For graphical clarity we omit loops, with the understanding they are always implied. (a) State interaction graph $D_{U_{Trot}}$. Nodes represent the different four-bit strings, and edges occur where matrix elements of U_{Trot} are nonzero. Note the presence of five disconnected subgraphs. In practice we treat each of these edges as undirected. (b) Transitive closure of $D_{U_{Trot}}$ resulting in the closed state interaction graph, in this case corresponds exactly to the state interaction graph for the hopping unitary: $D_{U_{Hop}} = D_{U_{Trot}}^+$. Each disconnected subgraph in $D_{U_{Trot}}$ has become a disconnected complete graph in $D_{U_{Hop}}^+$. (c) Each complete graph in $D_{U_{Hop}} = D_{U_{Trot}}^+$ corresponds to a symmetry-protected subspace G_s , where s denotes that SPS's conserved quantity. In this example, the subspace indices correspond to eigenvalues under particle number conservation, $S = \sum_{i=0}^3 \frac{1}{2}(1 - Z_i)$.

(we sometimes shorten this notation to just B) and showing how to construct it. For now we leave the form of U general and will specify particular forms when necessary.

Definition 2 (State Interaction Graph). Given a basis B and a unitary operator U , define a vertex set $V \equiv B$ and an undirected edge set $E \equiv \{(|b\rangle \leftrightarrow |b'\rangle) \mid |b\rangle, |b'\rangle \in B \mid \langle b'|U|b\rangle \neq 0\}$. Then the state interaction graph is defined by the ordered tuple $D_U \equiv (V, E)$. In other words, the basis states of B are assigned to vertices (nodes) in the interaction graph and edges are created between vertex states only where the matrix element of the evolution unitary between the two states is nonzero.

Strictly speaking, D_U should be a directed graph, where an edge points from $|b\rangle$ to $|b'\rangle$ if $\langle b'|U|b\rangle \neq 0$ and from $|b'\rangle$ to $|b\rangle$ if $\langle b|U|b'\rangle \neq 0$. However, for a symmetry operator S and time-dependent simulation unitary $U(t)$, one can show that $[S, U(t)] = 0 \iff [S, U^\dagger(t)] = 0$, meaning that symmetries of the evolution operator, and their associated protected subspaces, are invariant under time reversal. To remain consistent with this observation, we treat every directed relationship $\langle b'|U|b\rangle \neq 0 \implies (|b\rangle \rightarrow |b'\rangle)$ as an undirected edge ($|b\rangle \leftrightarrow |b'\rangle$). This treatment is equivalent to assuming that the true, directed state interaction graph corresponding to Def. 2 always has a cycle that leads back to every node, such as is the case in Fig. 1(a). Formally, we assume that if there is an edge $(|b\rangle \rightarrow |b'\rangle) \in D_U$, there also exists a path $\{|b'\rangle \rightarrow \dots \rightarrow |b\rangle\} \subseteq D_U$. This global cyclicity assumption enables us to use the notion of transitive closure in Sec. III B, and subsequently, in an uncomplicated manner that respects the time-reversal invariance of the resulting subspaces. It is justified on two points: (1) most simulation unitaries have a very regular, repetitive structure so that directed, acyclic state interaction graphs are likely only to arise in extremely pathological instances and (2) the failure of the global cyclicity assumption will only ever result in the artificial enlargement of a symmetry-protected subspace, and while such a failure leads to underconstrained

subspaces, which is bad for the efficacy of, e.g., postselection, it will never result in the corruption of simulation fidelity by overconstraining or throwing out good simulation data.

To construct D_U , we use the following steps. First, choose a set of Hilbert space basis vectors $B(\mathcal{H}_d) = \{|b\rangle\}$. Any symmetry-protected subspaces that are discovered must be formed by the basis vectors of this basis. In the context of quantum simulation, B dictates the basis in which a quantum computer will be measured. For example, parallel readout in the computational Z basis will result in bit strings, $B_{\text{comp.}} = \{|0\dots 00\rangle, |0\dots 01\rangle, \dots, |1\dots 11\rangle\}$, which are Pauli Z -string eigenvectors. In the context of classical simulation, the basis furnishes a representation for the d -dimensional vector of complex amplitudes that stores the evolving many-body wave function. When the evolution operator $U(\tau)$ is applied to a basis vector $|b\rangle$ for a single time-step the resulting state $|\psi(\tau)\rangle$ has a basis vector decomposition

$$|\psi(\tau)\rangle = U(\tau)|b\rangle = \sum_{|b'\rangle \in B} |b'\rangle \langle b'|U(\tau)|b\rangle = \sum_{|b'\rangle \in B} \alpha_{b'}(\tau)|b'\rangle, \quad (11)$$

where we suppress the time-ordering subscript j in τ_j for simplicity. With this decomposition for each $|b\rangle \in B$, we create the state interaction graph D_U of the operator $U(\tau)$: for each pair $|b\rangle, |b'\rangle$ such that $\alpha_{b'}(\tau) \neq 0$ in Eq. (11), we add an edge $(|b\rangle \leftrightarrow |b'\rangle)$ to D_U . If one has a direct $d \times d$ matrix representation of $U(\tau)$ on-hand, then the adjacency matrix for D_U can be read off directly as $\mathcal{A}_U = \text{bit}[\text{bit}[U(\tau)] + \text{bit}[U^\dagger(\tau)]]$, where if an entry in $U(\tau)$ or $U^\dagger(\tau)$ becomes nonzero for any value of t , its complex value is replaced by 1 under the operation $\text{bit}[\dots]$. This adjacency matrix can be formed, for example, by directly exponentiating a $d \times d$ Hamiltonian matrix with a time parameter. In practice, however, constructing or storing an entire evolution unitary in memory is costly, since the size of Hilbert space d grows exponentially in the number of qubits n in the simulation: $d = 2^n$. Indeed, one

of the main advantages of digital quantum simulation is the ability to break global evolution unitaries into sequences of local unitaries, at the expense of introducing error, which are then implemented as quantum gates. Therefore, being able to extract symmetry-protected subspaces from consideration of local operations, rather than from the global unitary they may approximate, is of clear benefit.

Towards this end, Fig. 1(a) shows the state interaction graph $D_{U_{\text{Trot}}}$ for one, potentially very bad depending on θ , Trotter approximation to the one-dimensional, four-qubit hopping unitary

$$U_{\text{Hop}}(\theta) = e^{i\theta \sum_{i=0}^2 (X_i X_{i+1} + Y_i Y_{i+1})/2}. \quad (12)$$

We take the Trotterization to be

$$U_{\text{Trot}}(\theta) = \text{iSWAP}_{01}(\theta) \times \text{iSWAP}_{12}(\theta) \times \text{iSWAP}_{23}(\theta), \quad (13)$$

where it should be understood that rightmost operators are applied first, and multiqubit states are ordered as $|q_0 q_1 q_2 q_3\rangle$. Notice that each $\text{iSWAP}_{i,i+1}(\theta)$ is parameterized by an arbitrary θ , and as such we expect each iSWAP operation to be a fractional operation that leaves some residual state behind, i.e., the operator has an identity component. The system is small enough that the state interaction graph can be checked by hand in this case, and the main observation to be made is that it is comprised of four disjoint subgraphs, each containing only transitions between states of fixed particle number (i.e., number of $|1\rangle$ s), and that all nodes in each subgraph have a path to all other nodes in the subgraph. The zero- and four-particle states are isolated, while the one-, two-, and three-particle states form directed, incomplete, isolated subgraphs. As we will see, this ‘‘incompleteness’’ feature is a pathology of the Trotter approximation which will be rectified in Sec. III B via transitive closure. It is also worth noting that constructing the state interaction graph using the Trotter-approximated unitary is not yet useful, since on a classical computer it currently still requires the storage and evolution of a 2^n -dimensional wave function. We will demonstrate the utility of constructing state interaction graphs from component k -local operators in Sec. III C.

B. Defining symmetry-protected subspaces with transitive closure

Suppose we have states $|b\rangle$ and $|b''\rangle$, such that $\langle b''|U|b\rangle = 0$ and $\langle b''|UU|b\rangle \neq 0$. This requires a transitive relation: $\langle b''|UU|b\rangle = \sum_{b' \in B} \langle b''|U|b'\rangle \langle b'|U|b\rangle$, because $|b\rangle$ must first transition to an intermediate state $|b'\rangle$ to reach its final destination at $|b''\rangle$. Therefore, in our state interaction graph D_U , as defined in Def. 2, there are edges $(|b\rangle \rightarrow |b'\rangle)$, $(|b'\rangle \rightarrow |b''\rangle) \in D_U$ which will have the same transitive relation encoded in the path $\{|b\rangle \rightarrow |b'\rangle \rightarrow |b''\rangle\} \subseteq D_U$. We use this duality to make the assumption that if the edges $(|b\rangle \rightarrow |b'\rangle)$, $(|b'\rangle \rightarrow |b''\rangle) \in D_U$, the transition amplitude $\langle b''|UU|b\rangle \neq 0$.

There are cases where the amplitudes of the states cancel, due to destructive interference, and break this transitive property on the level of individual basis state to basis state interactions. By ignoring the amplitudes of the basis states, we run the risk of including states in the subspace which would

be removed via destructive interference. This risk comes with the benefit of efficiently knowing which states are reachable in the quantum simulation, and for our applications it does not *add* any error to a simulation, as it does not break the underlying commuting subspace projection operator P_G ; the subspaces are simply not as restrictive as they could be. See Appendix A for a proof. This also allows one to define the symmetry-protected subspaces to be for *any* parametrization of the simulation unitary.

The transitive property exists for every state $|b\rangle$ in D_U , so we can take the transitive closure of the state interaction graph to create a *closed* state interaction graph D_U^+ . The transitive closure of an edge set E is a transitively closed edge set E^+ , where every pair of states $|b\rangle, |b'\rangle \in E$ which can be associated by any transitive relation, in other words can be connected by a path $\{|b\rangle \rightarrow \dots \rightarrow |b'\rangle\} \subseteq E$, has an edge $(|b\rangle \rightarrow |b'\rangle) \in E^+$ [35].

Definition 3 (Transitively Closed State Interaction Graph). Let $D_U = (V, E)$ be a potentially nonclosed state interaction graph for unitary U and basis B . Define $V^+ \equiv V$ to be the closed state interaction graph vertex (node) set and E^+ to be the state interaction graph edge set. An edge, $(|b\rangle \leftrightarrow |b'\rangle) \in E^+$, exists in this edge set if and only if there is a path between $|b\rangle$ and $|b'\rangle$ in D_U , $\{|b\rangle \leftrightarrow \dots \leftrightarrow |b'\rangle\} \subseteq E$. The transitively closed state interaction graph is then defined as $D_U^+ \equiv (V^+, E^+)$.

Because the original interaction graph represents single-operator state-to-state transitions, any two basis states which can discover each other through the quantum evolution have an edge in D_U^+ ; in other words,

$$\exists t \text{ s.t. } \langle b'|U(t)|b\rangle \neq 0 \Rightarrow (|b\rangle \leftrightarrow |b'\rangle) \in D_U^+, \quad (14)$$

for some time or operator exponent t . The transitive closure of an undirected, unweighted graph is a cluster graph, or a set of complete subgraphs; as discussed at the end of Sec. III A, we treat the graph D_U as undirected just for this purpose. The transitive closure of the state interaction graph for the Trotterized unitary in Fig. 1(a) can be seen in Fig. 1(b), which in this case turns out to be state interaction graph for the original hopping unitary, or $D_{U_{\text{Trot}}}^+ = D_{U_{\text{Hop}}}$. As with $D_{U_{\text{Trot}}}$, within each complete subgraph, total particle number is conserved.

If there is an edge $(|b\rangle \leftrightarrow |b'\rangle) \in D_U^+$, then $|b\rangle$ and $|b'\rangle$ lie in the same cluster of nodes within D_U^+ and thus share a conserved quantity of the underlying unknown symmetry. This cluster graph structure also makes it apparent that if a wave function is initialized as a linear combination of vectors in one subgraph G of D_U^+ : $|\psi_0\rangle = \sum_{b \in G} \alpha_b |b\rangle$, it will remain in that subgraph through its time evolution:

$$|\psi(t)\rangle = U(t)|\psi_0\rangle = \sum_{b' \in G} \alpha_{b'}(t) |b'\rangle \quad \forall t. \quad (15)$$

Therefore, each complete subgraph G within D_U^+ is a symmetry-protected subspace. We will formally prove that transitive closure on the state interaction graph can give an SPS with Thm. 1 in Sec. IV A.

D_U^+ can be represented as a list of disjoint sets of nodes, where each set has implied all-to-all connectivity. This set construction can be seen for the hopping unitary in Fig. 1(c). Here the index, s , of each subset, G_s counts the number of

conserved particles. Formally, we can define the associated symmetry-protected subspaces by constructing their projection operators according to Def. 1: $P_G = \sum_{b \in G} |b\rangle\langle b|$.

C. Basis state string edit map

We now have a method to identify symmetry-protected subspaces using the language of graph theory. However, actually computing these subspaces still requires the construction and manipulation of vectors and matrices in an exponentially large Hilbert space. Recall, though, that the systems of interest are defined by Hamiltonians composed of local operations $U = \prod_i U_i$ where each U_i is k -local, meaning it involves only k of the n total qubits in the system, and where in general we will have $k \ll n$. In this section we describe our mechanism for using this fact to build up SPSs efficiently with what we call the *basis string edit map*, denoted \mathcal{L} . This map enables computation of subspaces using only k -local operations on basis state vectors, so the computational complexity of operation with this map scales with k instead of n .

Definition 4 (Basis String Edit Map). Let $U_i(t)$ be a unitary operator that acts for a time t nontrivially on k of n qubits, $Q_k(i) \subseteq \{q_0, \dots, q_{n-1}\}$. The *basis string edit map* \mathcal{L}_{U_i} maps a basis state $|b\rangle$ to the set of basis states $\{|b'\rangle\}$ to which $|b\rangle$ can evolve after an arbitrary amount of time under $U_i(t)$. Formally, $\mathcal{L}_{U_i}(|b\rangle) = \{|b'\rangle \mid \exists t \langle b' | U_i(t) | b \rangle \neq 0\}$.

We can apply this construction to any unitary operator, including any k -local Trotter decomposition. Given a unitary operator which is a product of local operators $U = \prod_i U_i$, we form the set of local operators used in the Trotter decomposition, $\{U_i\}$. Then a basis string edit map can be formed for any subset of operators from $\{U_i\}$, as long as every operator $U_i \in U$ is included in at least one string edit map. We will use the decomposition

$$\mathbb{L} \equiv \{\mathcal{L}_{U_i} : U_i \in U\}, \tag{16}$$

which has one string edit map for each local operator in the Trotter decomposition.

In Def. 4 we have deliberately left out the exact space upon which \mathcal{L}_{U_i} acts. When operating on states in a basis B with \mathcal{L}_{U_i} , when $\dim(\mathcal{L}_{U_i}) < \dim(B)$, we will call \mathcal{L}_{U_i} a “substring edit map.” When $\dim(\mathcal{L}_{U_i}) = \dim(B)$, we refer to it as a “string edit map” or “full string edit map.”

For example, in the hopping unitary U_{Hop} given in Eq. (12), we can define $\mathcal{L}_{U_{\text{Hop}}}$ which would give $\mathcal{L}_{U_{\text{Hop}}}(|0100\rangle) = G_1$ where G_1 is in Fig. 1(c). In practice, the exact exponentiation of the Hamiltonian is unavailable, so more usefully, we take the Trotterization of Eq. (12) given by Eq. (13), and define string edit maps for the local iSWAP operators, $\mathbb{L}_{\text{Trot}} = \{\mathcal{L}_{\text{iSWAP}_{0,1}(\theta)}, \mathcal{L}_{\text{iSWAP}_{1,2}(\theta)}, \mathcal{L}_{\text{iSWAP}_{2,3}(\theta)}\}$. In Eq. (13) each local operator $U_i = \text{iSWAP}_{i,i+1}(\theta)$ would generate the corresponding edit map $\mathcal{L}_{\text{iSWAP}_{i,i+1}(\theta)}$, which operate as, e.g., $\mathcal{L}_{\text{iSWAP}_{1,2}(\theta)}(|0100\rangle) = \{|0010\rangle, |0100\rangle\}$. Here we have written the operators as acting on the full 2^n -dimensional Hilbert space and highlighted in bold the qubits that are part of the 2^k -dimensional subset of this space upon which $\mathcal{L}_{\text{iSWAP}_{1,2}(\theta)}$ acts. This k -local string edit map does not require information about any states besides those at the relevant indices, 1 and 2 in this case.

Algorithm III.1 Create a string edit map \mathcal{L}_{U_i} .

Require: Local unitary U_i , orthonormal basis $B_i = \{|b\rangle\}$
Ensure: $\dim(B_i) = \dim(U_i) = 2^k$ and $\text{Col}(U_i) = \text{Col}(B_i)$

$$\mathcal{A}_{U_i} \leftarrow \text{bit}[U_i] + \text{bit}[U_i^\dagger] \quad \triangleright \text{Add } U_i^\dagger \text{ to make the adjacency matrix undirected}$$

$$\mathcal{A}'_{U_i} \leftarrow \text{bit}[\mathcal{A}_{U_i}^2]$$

while $\mathcal{A}'_{U_i} \neq \mathcal{A}_{U_i}$ **do**
 $\mathcal{A}_{U_i} \leftarrow \mathcal{A}'_{U_i}$
 $\mathcal{A}'_{U_i} \leftarrow \text{bit}[\mathcal{A}_{U_i}^2]$
end while

for $|b\rangle \in B_i$ **do**
for $|b'\rangle \in B_i$ **do**
if $\mathcal{A}_{U_i}[|b\rangle, |b'\rangle] = 1$ **then**
 $\mathcal{L}_{U_i}(|b\rangle) \leftarrow \mathcal{L}_{U_i}(|b\rangle) \cup |b'\rangle$
end if
end for
end for
return \mathcal{L}_{U_i}

Algorithm III.1 creates \mathcal{L}_{U_i} by taking transitive closure of the adjacency matrix \mathcal{A}_{U_i} of D_{U_i} via Boolean matrix multiplication [36]. This algorithm requires $O(2^{3k})$ time to compute \mathcal{L}_{U_i} , $O(2^k)$ space to store it, and $O(1)$ time to use \mathcal{L}_{U_i} .

Algorithm III.1 is computationally trivial to compute for small unitary operators ($k = 2$ in the iSWAP example), but very expensive for the large unitary operators encountered in quantum simulations. Throughout the rest of this paper, we will use \mathcal{L}_{U_i} defined on small k to compute our subspaces in order to keep a small overhead (thus, in our terminology, we will always be talking about “substring edit maps”).

The object \mathbb{L} lets us create a version of the state interaction graph from Def. 2 which we call the *string interaction graph* $D_{\mathbb{L}}$; this construction compactly shows the state-to-state interactions in the algorithms presented in Sec. IV and V. The mapping itself is defined as $\mathbb{L}(|b\rangle) \equiv \{\mathcal{L}_i(|b\rangle) : \mathcal{L}_i \in \mathbb{L}\}$.

Definition 5 (String Interaction Graph). Let $D_{\mathbb{L}}$ be a graph defined by an ordered tuple $D_{\mathbb{L}} \equiv (V_{\mathbb{L}}, E_{\mathbb{L}})$ and $\mathbb{L} \equiv \{\mathcal{L}_{U_i} : U_i \in U\}$ for some Trotterized unitary $U = \prod_i U_i$. The vertex set is then given by $V_{\mathbb{L}} \equiv B(\mathcal{H}_d)$, and the edge set is given by $E_{\mathbb{L}} \equiv \{(|b\rangle \leftrightarrow |b'\rangle) : |b\rangle, |b'\rangle \in B \text{ and } |b'\rangle \in \mathbb{L}(|b\rangle)\}$.

While edges in D_U can capture the action of multiple operators at once, each edge in $D_{\mathbb{L}}$ is the action of only a single operator; therefore, if $D_U \equiv (V, E)$ is the state interaction graph and $D_{\mathbb{L}} \equiv (V, E_{\mathbb{L}})$ is the string interaction graph, then $E_{\mathbb{L}} \subseteq E$. Later we will show with the proof in Appendix B for Thm. 1 that despite this inequality, the transitive closure of the graph $D_{\mathbb{L}}^+$ is equivalent to the transitively closed state interaction graph $D_U^+ \equiv D_U^+$.

IV. ALGORITHM: CREATING THE SYMMETRY-PROTECTED SUBSPACE OF AN INITIAL STATE

We now turn to the construction of the entire symmetry-protected subspace of a given initial state using k -local substring edit maps and transitive closure, a construction consistent with the observation that if a symmetry exists locally everywhere in a quantum circuit, then it will also manifest

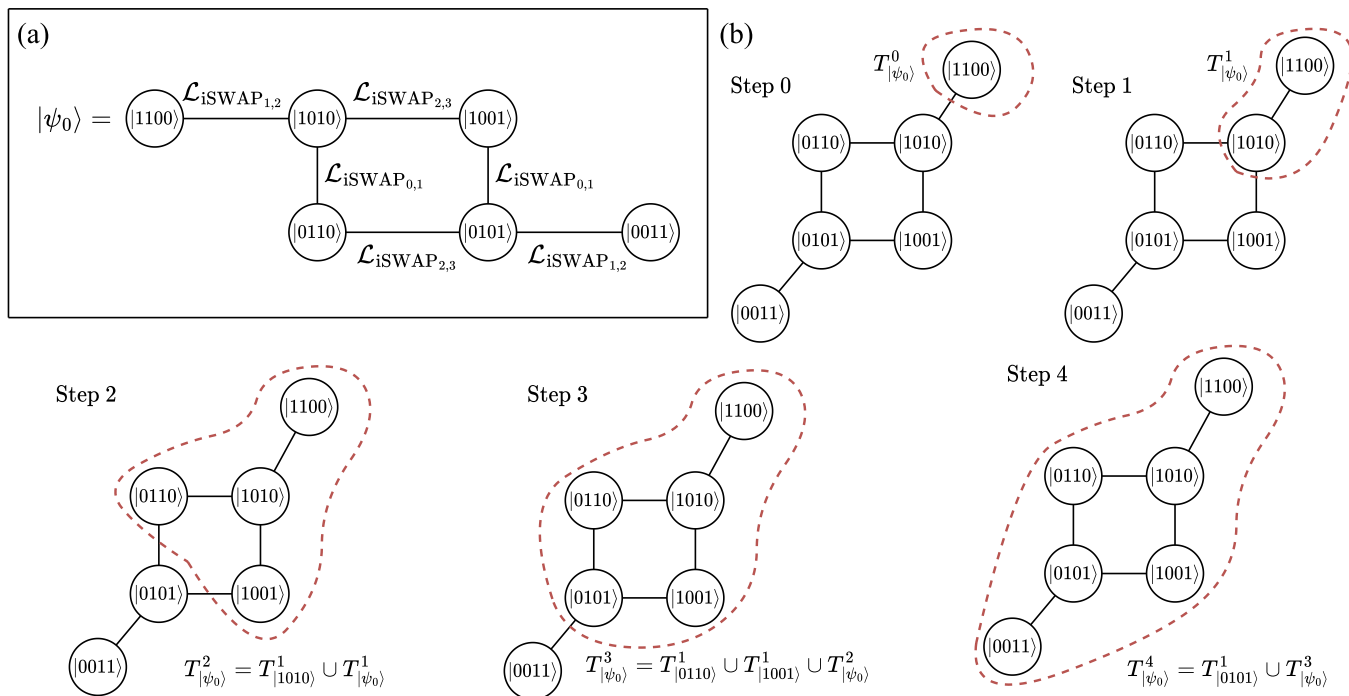


FIG. 2. This figure shows how we can incrementally build the symmetry-protected subspace for $U = \prod_{i=0}^2 \text{iSWAP}_{i,i+1}(\theta)$. (a) A single unclosed symmetry-protected subspace, or a subgraph of $D_{\mathbb{L}}$ for an iSWAP network. All operations $\mathbb{L}_{\text{Trot}} = \{\mathcal{L}_{\text{iSWAP}_{0,1}(\theta)}, \mathcal{L}_{\text{iSWAP}_{1,2}(\theta)}, \mathcal{L}_{\text{iSWAP}_{2,3}(\theta)}\}$ which are nonidentity on each node are shown as an edge. Notice the similarities between this graph and the $s = 2$ subgraph of Fig. 1(a): both have the same vertex set, but the edge set of $D_{\mathbb{L}}$ is a subset of the edge set in D_U . (b) Follow the recursion relation in Eq. (17) to iteratively build the symmetry-protected subspace $G_{|1100\rangle}$, starting from $|\psi_0\rangle = |1100\rangle$. Even though the graph in (a) is not equivalent to the graph in Fig. 1(a), their transitively closed graphs are equivalent, as can be seen by the vertices covered by the red line.

globally [37]. The algorithm by which we do so works by establishing a recurrence relation for computing new states, associated by symmetry protection to the initial state, in the simulation. This recurrence relation furnishes an efficient way to compute the transitive closure of select subgraphs of the entire Hilbert space, with no extraneous information. For a unitary simulation operator decomposed into enumerated local operations, $U(\tau_j) = \prod_{i=1}^m U_i(\tau_j)$, where each $U_i(\tau_j)$ is k -local, we will show that a symmetry-protected subspace $G_{|\psi_0\rangle}$ of the initial state $|\psi_0\rangle$ costs $O((m+1) \times |G_{|\psi_0\rangle}|)$ to compute, where m is the number of k -local operators in a single discrete time-step of the simulation, and $O(|G_{|\psi_0\rangle}|)$ to store with a breadth-first search [38]. We will give the algorithm for the case where the initial state $|\psi_0\rangle$ is a single measurement basis state (i.e., a product state). If $|\psi_0\rangle$ is a linear combination of measurement basis vectors, the algorithm can be repeated for each basis vector in the sum; this does not impact the asymptotic performance of the algorithm, as it only makes the computed subspace bigger.

We enumerate every state in a symmetry-protected subspace by transitively closing subgraphs created by the local basis substring edit maps established in Sec. III C, which return a set of basis states evolved to by their corresponding unitary operators in $O(1)$ when operating on a single basis state. We recursively build the subspace by checking the set of substring edit maps $\mathbb{L} \equiv \{\mathcal{L}_{U_i} : U_i \in U\}$ on each new state, until none are added. This process can be seen as the transitive closure of a subgraph of the graph $D_{\mathbb{L}}$. For U_{Trot} in Eq. (13), Fig. 2(a) shows an example of the subgraph of $D_{\mathbb{L}}$

corresponding to the action of each $\mathcal{L}_{\text{iSWAP}_{i,i+1}(\theta)} \in \mathbb{L}_{\text{Trot}}$ starting from the initial state $|\psi_0\rangle = |1100\rangle$ (self-edges are ignored as elsewhere in the paper).

A. Recurrence relation

To find the symmetry-protected subspace of an initial state, we begin by computing the string edit map for each k -local unitary, $\mathbb{L} \equiv \{\mathcal{L}_{U_i} : U_i \in U\}$. Next, we check the set of measurement basis strings generated by operating with each substring edit map on the initial state, notated $\mathbb{L}(|\psi_0\rangle) \equiv \{\mathcal{L}_i(|\psi_0\rangle) : \mathcal{L}_i \in \mathbb{L}\}$. We define the set $T_{|\psi_0\rangle}^1 \equiv \{|\psi_0\rangle\} \cup \mathbb{L}(|\psi_0\rangle)$. Then, for each *new* state $|\phi\rangle \in \mathbb{L}(|\psi_0\rangle)$, we check operations under the substring edit maps: $T_{|\psi_0\rangle}^2 = \mathbb{L}(T_{|\psi_0\rangle}^1) \equiv \{\mathbb{L}(|\phi\rangle) : |\phi\rangle \in \mathbb{L}(|\psi_0\rangle)\}$. This process repeats until no new states are found through the following recurrence relation:

$$\begin{cases} T_{|\psi_0\rangle}^1 \leftarrow \{|\psi_0\rangle\} \cup \mathbb{L}(|\psi_0\rangle) & \text{base case} \\ T_{|\psi_0\rangle}^{i+1} \leftarrow T_{|\psi_0\rangle}^i \cup \mathbb{L}(T_{|\psi_0\rangle}^i) & \text{recursive case} \\ T_{|\psi_0\rangle}^{i+1} = T_{|\psi_0\rangle}^i & \text{stop condition} \end{cases} \quad (17)$$

The stop condition activates if no new states are found, that is, when additional operations drawn from \mathbb{L} do not unveil any new states. Steps 0–4 in Fig. 2(b) show how the recurrence relation manifests for the input state $|\psi_0\rangle = |1100\rangle$ and the set of substring edit maps generated from the Trotterization in Eq. (13). Once Eq. (17) reaches the stop condition $T_{|\psi_0\rangle}^{i+1} = T_{|\psi_0\rangle}^i$ and returns, we define the symmetry-protected

subspace $G_{|\psi_0\rangle}$ to which the state $|\psi_0\rangle$ belongs via

$$G_{|\psi_0\rangle} \equiv T_{|\psi_0\rangle}^{i+1}, \quad (18)$$

where the definition “ \equiv ” in Eq. (18) should be taken to mean “all basis states in $T_{|\psi_0\rangle}^{i+1}$ viewed as nodes in a complete graph.” We can create the Kleene Closure of the set of substring edit maps, denoted \mathbb{L}^* , which is the set of all finite concatenations of substring edit maps, including the identity. Any arbitrary string of substring edit maps \mathbb{L}^* applied to $|\psi_0\rangle$ will result in a state in $G_{|\psi_0\rangle}$, by its definition. Hence, one can write

$$\mathbb{L}^*(|\psi_0\rangle) = G_{|\psi_0\rangle}. \quad (19)$$

We now state our main result.

Theorem 1. Let $U(t) = \mathcal{O}_{op}[\prod_{j=1}^p \prod_{i=1}^m U_i(\tau_j)]$ be a quantum simulation unitary of duration t acting on n qubits as in Eq. (2), divided into p time steps, where each $U_i(\tau_j)$ is k -local and is time step parameterized by τ_j , and some operator ordering (such as time ordering) is specified. Let \mathcal{L}_{U_i} be the string edit map corresponding to any available parametrization of $U_i(\tau_j)$ and the set of such maps $\mathbb{L} \equiv \{\mathcal{L}_{U_i} : i \in \{1, \dots, m\}\}$. Let $B(\mathcal{H}_{2^n}) \equiv \{|b\rangle\}$ be the basis in which computations (measurements) are being performed classically (quantumly). Then, given an input state $|\psi_0\rangle$, expressed in the basis B , if $|b\rangle \notin G_{|\psi_0\rangle}$, where $G_{|\psi_0\rangle}$ is constructed according to Eqs. (17) and (18), then $\langle b|U(t)|\psi_0\rangle = 0$.

For a proof, see Appendix B. Note that for $\langle b_f|U(t)|\psi_0\rangle$ to vanish under these conditions, $G_{|\psi_0\rangle}$ must satisfy Def. 1 as demonstrated in Lem. 0.1. In other words, for $G_{|\psi_0\rangle}$ to be able to exclude particular basis states for arbitrary evolution times, it must be a symmetry-protected subspace. Note that Thm. 1 immediately provides two corollaries.

Corollary 1 (Postselection). For simulation on an idealized, noise-free quantum computer, if $|b_f\rangle \notin G_{|\psi_0\rangle}$ then $\|\langle b_f|U(t)|\psi_0\rangle\|^2 = 0$. Hence, if the state $|b_f\rangle$ is measured in the output of a noisy quantum device, it can be assumed that the state arose as a result of error and may be discarded.

Corollary 2 (Global Subspace). We assumed a Trotterized form for $U(t)$ in the statement and proof of Theorem 1. However, we can formally recover the corresponding global simulation unitary by taking the limit $p \rightarrow \infty$ where $\tau_j = jt/p$ in the time-ordered case and $\tau_j = t/p \forall j$ when time ordering is unnecessary (such as when the Hamiltonian is time-independent). Nothing in the proof of Theorem 1 relies upon the finiteness of p or discreteness of the corresponding time differential t/p . Therefore, our result holds for global simulation unitaries as well. This implies that one can reduce the resource requirements in classical simulations of $U(t)$ by only evolving basis states $|b\rangle \in G_{|\psi_0\rangle}$.

B. Pseudocode

With an understanding of the recurrence relation in Eq. (17) and how it can compute symmetry-protected subspaces, we present an algorithm which can enumerate these subspaces using a breadth-first search. Breadth-first search to enumerate an entire graph (V, E) has computational complexity $O(|V| + |E|)$. In our implementation, there are $|G_{|\psi_0\rangle}|$ vertices and we check for m edges at each vertex, giving $O(m \times |G_{|\psi_0\rangle}|)$ edges in the entire graph. Thus, our

Algorithm IV.1 Enumerate symmetry-protected subspace $G_{|\psi_0\rangle}$ with a breadth-first search.

```

Require: String edit maps  $\mathbb{L}$  of the simulation operator,
initial state  $|\psi_0\rangle$ 
 $G \leftarrow \{|\psi_0\rangle\}$ 
Let  $Q$  be a first-in-first-out queue
 $Q.enqueue(|\psi_0\rangle)$ 
while  $Q \neq \emptyset$  do
   $|b\rangle \leftarrow Q.dequeue()$ 
  for  $|b'\rangle \in \{\mathcal{L}_{U_i}(|b\rangle) : \mathcal{L}_{U_i} \in \mathbb{L}\}$  do
    if  $|b'\rangle \notin G$  then
       $G \leftarrow G \cup |b'\rangle$ 
       $Q.enqueue(|b'\rangle)$ 
    end if
  end for
end while
return  $G_{|\psi_0\rangle} \leftarrow G$ 

```

breadth-first search to enumerate the symmetry-protected subspace is $O(|G_{|\psi_0\rangle}| + m \times |G_{|\psi_0\rangle}|) = O((m+1) \times |G_{|\psi_0\rangle}|)$

Algorithm IV.1 uses the set G , which is eventually the symmetry-protected subspace, to track which states have already been added to the queue Q during the runtime of the algorithm and prevent them from being checked more than once. If this set uses the hash of the basis state’s bit strings, insertion and search will be average case $O(1)$. Getting the set of single-operator transitions $T_{|b\rangle}^1$ is $O(m)$, where m is the number of unitary operators in the system, for a single state $|b\rangle$; while not all m substring edit maps will provide an edge, as many might act as identity on the state $|b\rangle$, each string edit map must still be checked. This set is computed for each state discovered, and each state discovered is never added to the queue Q more than once, which confirms our original complexity analysis of $O((m+1) \times |G_{|\psi_0\rangle}|)$.

This algorithm computes a set equivalent to that described by Eq. (17). See Appendix B for a proof that this set is a symmetry-protected subspace.

C. Usage and limits

As stated in Sec. I, postselection for quantum simulations is the aim of our methods. In order to perform postselection with the algorithm from this section, the entire symmetry-protected subspace $G_{|\psi_0\rangle}$ must be known, and each measurement $|b_f\rangle$ is verified via

$$\begin{cases} |b_f\rangle \in G_{|\psi_0\rangle} & \text{assumed no error} \\ |b_f\rangle \notin G_{|\psi_0\rangle} & \text{known error.} \end{cases} \quad (20)$$

Because this requires the computation of the entire subspace, it can still become computationally intractable. As mentioned in Sec. I, we benchmark this algorithm against the Heisenberg-XXX, T6 QCA, and F4 QCA quantum simulations. The computationally limiting factor of this algorithm is the size of the symmetry-protected subspace, as it has time complexity $O((m+1) \times |G_{|\psi_0\rangle}|)$ and spatial complexity $O(|G_{|\psi_0\rangle}|)$. See Fig. 3 for a depiction of the size of the subspaces, up to 17 qubits. Figure 3(c) is especially significant because, as alluded to in Sec. II B 3, this model previously had no known conservation laws, but these data show the

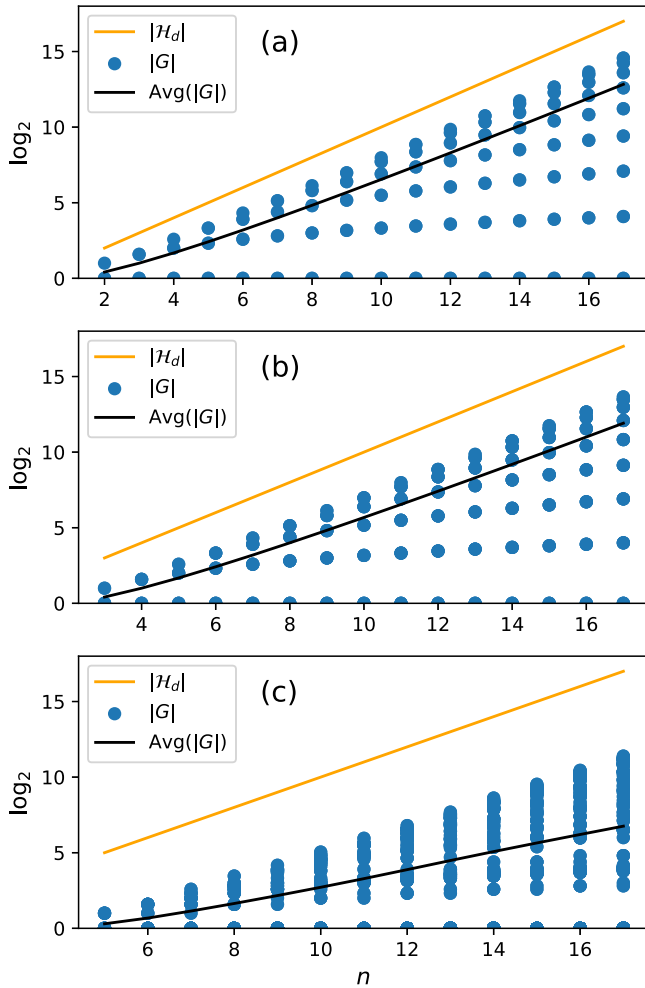


FIG. 3. The number of states in symmetry-protected subspaces for each model with n qubits (implying Hilbert space dimension $d = 2^n$) up to $n = 17$ qubits on a \log_2 scale. Each blue dot is the size of an individual subspace, and the black line is the average SPS dimension for the current system size. (a) Heisenberg-XXX. (b) T_6 QCA. (c) F_4 QCA, which has no known symmetry operator.

partitioning of Hilbert space into symmetry-protected subspaces. By examining Fig. 3, we can see that the worst case of each subspace size, $\max(|G|)$, is $\log_2(\max(|G|)) \approx \log_2(|\mathcal{H}_d|) - k$ when the model is comprised of k -local operations. The important thing to see is that the size of symmetry-protected subspaces still scales exponentially in the worst and average cases; while smaller than the full Hilbert space, they are only linearly smaller. We address this step to postselection with another algorithm in Sec. V.

V. ALGORITHM: VERIFICATION OF A SHARED SYMMETRY-PROTECTED SUBSPACE

Because the worst-case size of a symmetry-protected subspace is still exponential in the number of qubits, Alg. IV.1 is only applicable at relevant system sizes for simulations where $G_{|\psi_0\rangle}$ is not exponentially large. Thus, to generalize the usability of symmetry-protected subspaces to any simulation, we present an alternative algorithm in this section that uses an

efficient, but greedy, heuristic. Instead of computing the entire exponentially large SPS, this method performs an efficient search for a path of substring edit maps to connect the initial state $|\psi_0\rangle$ and a measured state $|b_f\rangle$ in the graph $D_{\mathbb{L}}$. The heuristic nature of this algorithm means that it produces only approximate results (albeit with a degree of approximation that can be continuously improved at the cost of more time complexity), and since its runtime scales favorably, it can be used to check the output of quantum simulations well beyond the scale where other classical methods become intractable.

Naive search algorithms in a graph traditionally use a breadth-first search from an initial vertex [39], which is what we described in Alg. IV.1 to enumerate the symmetry-protected subspace through transitive closure; as stated, this is too computationally expensive in many cases.

Our greedy algorithm works as follows: Following all edges (application of the set of substring edit maps \mathbb{L}) from a state generates a set of possible states accessible from that state. Working both forward from $|\psi_0\rangle$ and backward from $|b_f\rangle$, we have found a *path* when steps from both directions lead to a shared element. To detect whether this has happened, we use a simple observation about ordered sets: two sets are the *same* set if they have the same minimal (or maximal) element. A natural order for sets of quantum computational basis states (i.e., binary strings) is just the integer they encode. So our greedy algorithm works by building two sets (one starting from the initial state and one from the measured state) and comparing them; these sets are subsets of the symmetry-protected subspace corresponding to the initial and final states. If these sets have any common elements (which we check in constant time by looking at their *minimal* elements), then we have found a path connecting the initial and measured state, and they occupy the same SPS.

In our methods, postselection's accept or reject decision is made by checking for a measurement result in the symmetry-protected subspace of the initial state. Using the rationale outlined above, we can declare with certainty that two states share a SPS when their paths collide. On the other hand, if the two paths terminate in dissimilar minima we *assume* the initial and measured states inhabit disjoint subspaces. We say "assume" because the accuracy of our heuristic differs depending on the simulation; when the locally minimal choice at each step from either state does not build a path to the true minimum element, the two states may build distinct paths while occupying the same SPS.

A. Searching in the string interaction graph

This algorithm builds a single branch, following local minima in depth-limited breadth-first searches, from a single starting vertex in the $D_{\mathbb{L}}$ graph, where each vertex $|b\rangle$'s edge set is given by $\mathbb{L}(|b\rangle)$.

Throughout this section we use the integers encoded by the bit strings b of basis states $|b\rangle \in B(\mathcal{H}_{2^n})$, which provides a natural ordering to states. Let the function $\min(A)$ on a set of states A return the state with the smallest encoded integer in that set:

$$\min(A) = |b\rangle \text{ s.t. } b \leq b', \forall |b'\rangle \in A. \quad (21)$$

To find the minimal element of the symmetry-protected subspace, we start with an element $|b_0\rangle = |\psi_0\rangle$ or $|b_f\rangle$ and build a set towards the minimum of $G_{|b_0\rangle}$, notated $\min(G_{|b_0\rangle})$, with locally optimal decisions. At each step $|b_{\text{curr}}\rangle$ in the search, we compute $T_{|b_{\text{curr}}\rangle}^\mu$ from Eq. (17), which is a breadth-first search to depth μ , or every state in $\mathbb{L}^\mu(|b_{\text{curr}}\rangle)$. Then the starting point for the next step, $|b_{\text{next}}\rangle$, is the state with the smallest binary encoded integer in $T_{|b_{\text{curr}}\rangle}^\mu$, notated $\min(T_{|b_{\text{curr}}\rangle}^\mu)$. Repeat until the set $T_{|b_{\text{curr}}\rangle}^\mu$ does not offer a state smaller than $|b_{\text{curr}}\rangle$. Let this process be represented by the recursive function χ :

$$\chi(|b_{\text{curr}}\rangle, \mu) = \begin{cases} |b_{\text{curr}}\rangle & \text{if } |b_{\text{curr}}\rangle = \min(T_{|b_{\text{curr}}\rangle}^\mu) \\ \chi(\min(T_{|b_{\text{curr}}\rangle}^\mu), \mu) & \text{otherwise} \end{cases} \quad (22)$$

To reiterate, because each step is given by applications of the substrings edit maps, $\chi(|b_0\rangle, \mu)$ and $|b_0\rangle$ must share a symmetry-protected subspace, i.e., $\chi(|b_0\rangle, \mu) \in G_{|b_0\rangle}$. Therefore, when $\chi(|\psi_0\rangle, \mu) = \chi(|b_f\rangle, \mu)$ there *must* be a sequence of substrings edit maps between $|\psi_0\rangle$ and $|b_f\rangle$; i.e., there is a path $\{|\psi_0\rangle \leftrightarrow \dots \leftrightarrow |b_f\rangle\} \subseteq D_{\mathbb{L}}$ and $|b_f\rangle \in \mathbb{L}^*(|\psi_0\rangle)$, which is an equivalent statement to $|b_f\rangle \in G_{|\psi_0\rangle}$.

When both searches conclude in the true minimal element of their corresponding symmetry-protected subspaces, $\chi(|\psi_0\rangle, \mu) = \min(G_{|\psi_0\rangle})$ and $\chi(|b_f\rangle, \mu) = \min(G_{|b_f\rangle})$, we can conclude with certainty that the states do or do not inhabit the same subspace if $\chi(|\psi_0\rangle, \mu) = \chi(|b_f\rangle, \mu)$ or $\chi(|\psi_0\rangle, \mu) \neq \chi(|b_f\rangle, \mu)$. However, if either search does *not* conclude in their targeted minimal state, our conclusions can be wrong. Suppose $|b_f\rangle \in G_{|\psi_0\rangle}$, which means that $\min(G_{|\psi_0\rangle}) = \min(G_{|b_f\rangle})$, but the search result $\chi(|b_f\rangle, \mu)$ finds a false minima of $G_{|b_f\rangle}$, meaning $\chi(|b_f\rangle, \mu) \neq \min(G_{|b_f\rangle})$. When postselecting with our procedure under these conditions, state $|b_f\rangle$ would be wrongly rejected because the underlying assumption for this heuristic, that $\chi(|b_0\rangle, \mu) = \min(G_{|b_0\rangle})$, is wrong. In practice, the true minimal element of the symmetry-protected subspace is unknown; thus, the assertion that χ finds the minimum is always an assumption.

Consequently, if the result of the two searches is a collision, $\chi(|\psi_0\rangle, \mu) = \chi(|b_f\rangle, \mu)$, we *know* the two states must share a symmetry-protected subspace, even if the searches are at a false minima. If the two searches do not find a common element, we *assume* the initial and measured states occupy separate symmetry-protected subspaces. As such, measurement outcomes that lie within the initial state's symmetry-protected subspace can be rejected. Thus, each measurement $|b_f\rangle$ is verified with

$$\begin{cases} \chi(|b_f\rangle, \mu) = \chi(|\psi_0\rangle, \mu) & \text{assumed no error} \\ \chi(|b_f\rangle, \mu) \neq \chi(|\psi_0\rangle, \mu) & \text{assumed error} \end{cases} \quad (23)$$

Formally, we can state that the protected subspace formed by χ around an initial state $|\psi_0\rangle$ is

$$P'_G = \sum_b |b\rangle\langle b| \text{ s.t. } \chi(|b\rangle, \mu) = \chi(|\psi_0\rangle, \mu), \quad (24)$$

which is approximately equal to the true symmetry-protected subspace, $P_G \approx P'_G$.

The confidence of this assumption depends on the quantum system being studied and the depth μ of the local breadth-first searches. Thus, we present arguments to support it for our three exemplar systems: Heisenberg-XXX, T_6 QCA, and F_4

QCA as they are outlined in Sec. II B. For the Heisenberg-XXX model this assumption is always correct at $\mu = 1$: systems whose substrings edit maps \mathbb{L} are isomorphic to nearest-neighbor SWAP substrings edit maps \mathbb{L}_{SWAP} , such as the Heisenberg-XXX model, provably find $\min(G_{|b_0\rangle})$ using $\chi(|b_0\rangle, \mu = 1)$, as shown in Appendix C 1. We also numerically find that $\chi(|b_0\rangle, \mu = 2)$ is exact for the T_6 QCA model. Out of our three studied systems the F_4 QCA demonstrates the worst accuracy: $\chi(|b_0\rangle, \mu = 9)$ still returns false minima, which potentially causes a false rejection. In Sec. V C 1 we analyze these properties of the T_6 and F_4 QCAs more closely. In Sec. VI C we show that postselection is still effective when χ causes a false rejection to occur.

The computational performance of this method is also model-dependent. Let *Depth* denote the depth of our search, or the number of intermediate states traversed to [calls to the second line in Eq. (22)] before finding a locally optimal vertex. This means the sequence of substrings edit maps between $|b_0\rangle$ and $\chi(|b_0\rangle, \mu)$ is length $O(\mu \times \text{Depth})$, and the computational complexity of Eq. (22) is $O(m^\mu \times \text{Depth})$.

Because the reliability and computational complexity of this postselection method varies, before applying the algorithm to a given quantum simulation, benchmarks should be performed on a small version of the system to understand the local search depth μ required to obtain reliable results.

B. Pseudocode

Here we describe the algorithm, Alg. V.1, to find the minimum binary encoded state in the symmetry-protected subspace of a given basis state. This algorithm starts at a

Algorithm V.1. Greedy pathfinding to the minimum binary encoded state.

```

Require: Path starting state  $|b_0\rangle$ , substrings edit maps  $\mathbb{L}$ ,
search limiting integer  $\mu$ 
 $|b_{\text{next}}\rangle \leftarrow |b_0\rangle, |b_{\text{curr}}\rangle \leftarrow \text{null}$ 
while  $|b_{\text{curr}}\rangle \neq |b_{\text{next}}\rangle$  do
     $T_{|b_{\text{curr}}\rangle} \leftarrow \{|b_{\text{curr}}\rangle, |b_{\text{next}}\rangle\}$ 
     $|b_{\text{curr}}\rangle \leftarrow |b_{\text{next}}\rangle$ 
    Let  $Q$  be a queue
     $Q.\text{enqueue}(|b_{\text{curr}}\rangle, 0)$   $\triangleright$  Each element in  $Q$  is tuple
    representing (state, depth)
    while  $Q \neq \emptyset$  do  $\triangleright$  Breadth-first search centered
    around  $|b_{\text{curr}}\rangle$ 
         $|b'\rangle, \eta \leftarrow Q.\text{dequeue}()$ 
        if  $\eta = \mu$  then break  $\triangleright$  Reached the depth limit
        else
            for  $|b''\rangle \in \{\mathcal{L}_{U_i}(|b'\rangle) : \mathcal{L}_{U_i} \in \mathbb{L}\}$  do
                if  $|b''\rangle \notin T_{|b_{\text{curr}}\rangle}$  then
                     $Q.\text{enqueue}(|b''\rangle, \eta + 1)$ 
                     $T_{|b_{\text{curr}}\rangle} \leftarrow T_{|b_{\text{curr}}\rangle} \cup |b''\rangle$ 
                end if
            end for
        end if
    end while
     $|b_{\text{next}}\rangle \leftarrow \min(T_{|b_{\text{curr}}\rangle})$ 
end while
return  $\chi(|b_0\rangle, \mu) = |b_{\text{curr}}\rangle$ 
    
```

state $|b_0\rangle$ and follows the local minima of limited breadth-first searches in an attempt to find the set's minimal element, resulting in $\chi(|b_0\rangle, \mu)$.

Notice that the innermost *while* loop is borrowed from Alg. IV.1 to compute each $T_{|b_{\text{curr}}\rangle}^\mu$, where $|b_{\text{curr}}\rangle$ is the current best minimum state. This time the queue Q uses the ordered tuple $(|b'\rangle, \eta)$ as its elements, where $|b'\rangle$ is the current, unchecked, state in the breadth-first search and η is the number of edges (i.e., depth) that $|b'\rangle$ is away from the current optimal solution $|b_{\text{curr}}\rangle$. Each node checked in this loop has $O(m)$ new edges, and we check a maximum depth of μ , giving the inner loop computational complexity $O(m^\mu)$. The set $T_{|b_{\text{curr}}\rangle}^\mu$ is checked for every node traversed to in the search, and our search traverses to *Depth* elements, which is how we arrive at an overall complexity of $O(m^\mu \times \text{Depth})$ to compute $\chi(|b_0\rangle, \mu)$.

To postselect with Alg. V.1, run it once for $|\psi_0\rangle$ to get $\chi(|\psi_0\rangle, \mu)$, again for each measurement $|b_f\rangle$ to get $\chi(|b_f\rangle, \mu)$, and check the results in Eq. (23).

As an aside, if all states computed in the search, $T_{|b_{\text{curr}}\rangle}^\mu$, are cached with the result of their search, subsequent executions of the algorithm can be preempted with their previously calculated result if a state is found inside the cache.

C. Benchmarking data for Algorithm V.1

This subsection presents benchmark data for Alg. V.1 that demonstrates its practical applicability for our example systems. These benchmarks include heuristic reliability and search depth. We supplement these data with proofs or explicit analytical forms when possible. When this is not possible, we instead rely on extrapolation from the data to inform the asymptotics of these quantities. We find that for the Heisenberg-XXX model, the heuristic is provably exact (Appendix C 1) and the search depth has an explicit equation (Appendix C 2), while the T_6 and F_4 QCA models rely on data-driven intuition for these quantities.

1. Search reliability

If searching for the minimal element from $|b_0\rangle = |\psi_0\rangle$ or $|b_f\rangle$ fails, because $\chi(|b_0\rangle, \mu) \neq \min(G_{|b_0\rangle})$, then it is possible to falsely assume that two states occupy disjoint subspaces. As such, we examine the reliability of the searching function χ on each exemplar system.

In the Heisenberg-XXX model, the base case for χ , as described in Eq. (22), activates only at a state which is always the minimum binary-encoded state in the symmetry-protected subspace. This is because the set of substring edit maps for the Heisenberg-XXX system is isomorphic to the set of substring edit maps for the nearest-neighbor SWAP network: $\mathbb{L}_{\text{HeisXXX}} = \mathbb{L}_{\text{SWAP}}$. Appendix C 1 proves that χ at $\mu = 1$ is exact for \mathbb{L}_{SWAP} .

Because the T_6 QCA and F_4 QCA do not have SWAP-isomorphic substring edit maps, we do not have any available proofs for their exactness, and instead rely on simulating χ for these models. The results of this are in Fig. 4, where for the T_6 and F_4 QCA models we plot the proportion of states $|b\rangle$ in the computational Z basis $B_{\text{comp}}(\mathcal{H}_d)$ such that $\chi(|b\rangle, \mu) \neq \min(G_{|b\rangle})$ for each $n \in [k, 17]$.

First, notice Fig. 4(a), which shows search failures for the T_6 QCA model; while $\mu = 1$ causes most searches to fail as

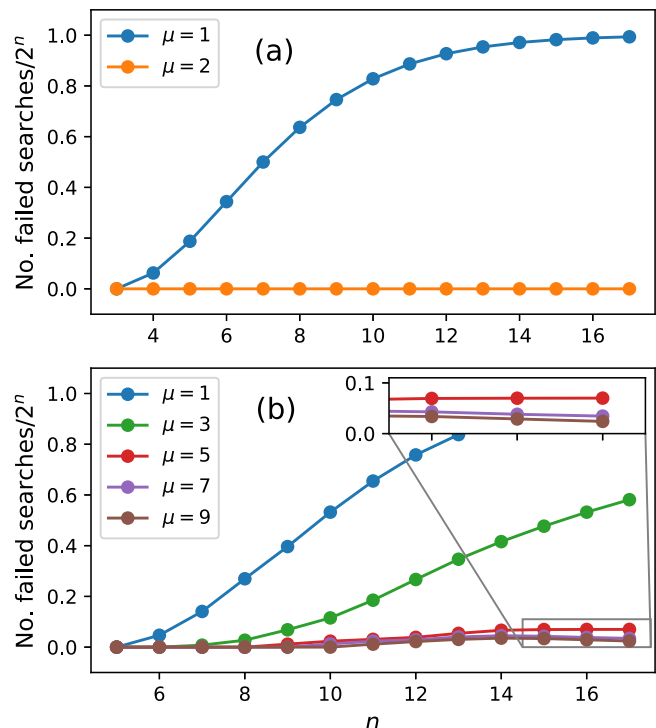


FIG. 4. The proportion of states in the computational Z basis $|b\rangle \in B_{\text{comp}}(\mathcal{H}_d)$, where $\chi(|b\rangle, \mu) \neq \min(G_{|b\rangle})$ when computed with Alg. V.1, called “failed searches,” at different system sizes n , for each μ used. (a) T_6 QCA. At $\mu = 2$, Alg. V.1 becomes exact for this system. (b) F_4 QCA. High μ can still fail, but does asymptotically better for $\mu \geq 5$.

n increases, $\mu = 2$ causes *every* search to succeed for every n in the domain. Thus, it we can assume that the heuristic for Alg. V.1 is accurate for this model and that $\chi(|b\rangle, \mu = 2) = \min(G_{|b\rangle})$, $\forall |b\rangle \in B$.

Next, notice Fig. 4(b), which shows search failures for the F_4 QCA model; we use $\mu = 1, 3, 5, 7, 9$, and find that χ can fail even with high μ . We see that $\mu = 1, 3$ sees majority failures over the domain, that $\mu = 5, 7, 9$ has bounded failure ≈ 0.05 , and that no μ completely removes failures.

This metric determines only the reliability of χ , not the reliability of postselecting with χ . This is because, as discussed previously in Sec. V A, two searches can fail and obtain the same false minima, $\chi(|\psi_0\rangle, \mu) = \chi(|b_f\rangle, \mu) \neq \min(G_{|\psi_0\rangle}) = \min(G_{|b_f\rangle})$, i.e., two failed searches can still successfully identify that the states share a conserved quantity. Therefore, the search reliability shown in Fig. 4 should be seen as an *upper bound* on the error caused by the heuristic.

Sections VIB and VIC explore this reliability further in the context of postselection on noisy quantum data. In the context of postselection, we find that even small $\mu = 3$ is acceptable for the F_4 QCA, as the most likely measurement outcomes also share false minima with the initial state. We discuss this observation more with the data that support it in Sec. VIC.

2. Search depth

The time performance of Alg. V.1 is heavily dependent on the length of the path taken. The asymptotic worst case of

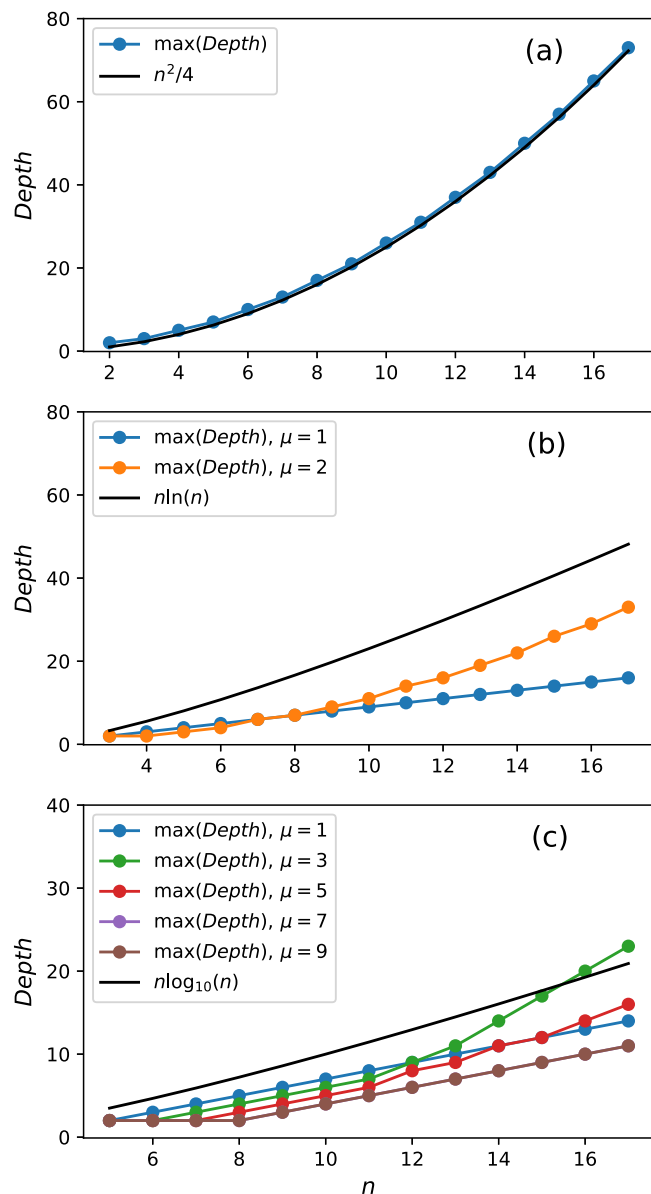


FIG. 5. The longest path, denoted $Depth$, taken at each system size n for the models (a) Heisenberg-XXX. (b) T_6 QCA. (c) F_4 QCA.

the search depth is system-dependent, being identified as $n^2/4$ for the Heisenberg-XXX model in Appendix C 2, and closer to $O(n \ln(n))$ for the T_6 QCA and $O(n \log_{10}(n))$ for the F_4 QCA by observing the largest search depth for the algorithm at small n , shown in Fig. 5. In this figure we find the largest path length at each system size for each relevant μ . For the T_6 model, shown in Fig. 5(b), search depth becomes longer as the parameter μ increases, indicating that false minima are avoided. For the F_4 model, shown in Fig. 5(c), the search depth becomes shorter as μ increases; this is because higher μ means more edges are traversed at each step in the search, and thus less depth is needed in the search overall.

These path lengths, combined with the μ found in Sec. V C 1, results in the computational complexity of Alg. V.1 on a single state being Heisenberg-XXX is $O(mn^2)$, T_6 QCA is $O(m^2 n \ln(n))$, and F_4 QCA is $O(m^\mu n \log_{10}(n))$. The

μ exponent is left in the F_4 QCA computational complexity because, as shown in Sec. V C 1, we find no value of μ that guarantees that a search finds the minimal element, and thus expect μ to vary depending on the symmetry-protected subspace occupied for the simulation. As we can see in Fig. 4(b), $\mu \geq 5$ is mostly sufficient.

VI. POSTSELECTION IN EMULATED NOISY SIMULATIONS

To demonstrate the power of our algorithm, we perform automated postselection on simulated quantum computations with noise. We perform discretized time evolutions with our exemplar systems, the Heisenberg-XXX, T_6 QCA, and F_4 QCA models, and show that our postselection methods restructure noisy measurement probability distributions to be closer to the one found in an ideal simulation. We first run the simulation without errors via $U(t)|\psi_0\rangle$, constructing the measurement distribution at each time step. Next, we repeated the simulations with depolarizing noise injected after each layer of gates with independent probability $\epsilon/3 = 0.005, 0.01, 0.02, 0.05$ of X_i, Y_i, Z_i on each qubit i . See Appendix D 1 for more noise model details. We use Kullback-Liebler divergence [40] to quantify the distance between the ideal and noisy measurement distributions constructed at each measurement layer, with and without postselection using Alg. V.1 and the symmetry check in Eq. (23).

We observe a significant increase in accuracy of the data when using our postselection methods without adding circuit runs, even in simulation subspaces which have imperfect pathfinding.

A. Methods

We simulate discrete time evolution, and for each discrete time step a sequence of measurements is used to construct a probability distribution. Let $P(p)$ be the measurement distribution of the wave function after p Trotter steps, such that $P(p) \approx |||\psi(p)\rangle|^2$ up to shot noise. We compare an ideal simulation measurement distribution $P_{\text{ideal}}(p)$, a noisy simulation without postselection $P_{\text{raw}}(p)$, and a noisy simulation with postselection using Alg. V.1 at a given μ , $P_{\text{ps},\mu}(p)$. Let $P(p)$ be a normalized probability distribution constructed by a sequence of M measurements after p Trotter steps, and $P(p, b) \approx ||\langle b|\psi(p)\rangle|^2$, which is the probability amplitude of state $|b\rangle$ at that layer p , up to shot noise. We use Kullback-Liebler divergence [40], defined for arbitrary probability distributions P and Q as

$$\mathcal{D}(P(p), Q(p)) \equiv \sum_{b \in \mathcal{B}(\mathcal{H}_d)} P(p, b) \ln \left[\frac{P(p, b)}{Q(p, b)} \right], \quad (25)$$

which measures the distance between two probability distributions, and use the following equation from [41] as a simulation's fidelity:

$$\mathcal{F}(P_{\text{sim}}(p), P_{\text{ideal}}(p)) \equiv 1 - \frac{\mathcal{D}(P_{\text{sim}}(p), P_{\text{ideal}}(p))}{\mathcal{D}(P_{\text{IRN}}, P_{\text{ideal}}(p))}, \quad (26)$$

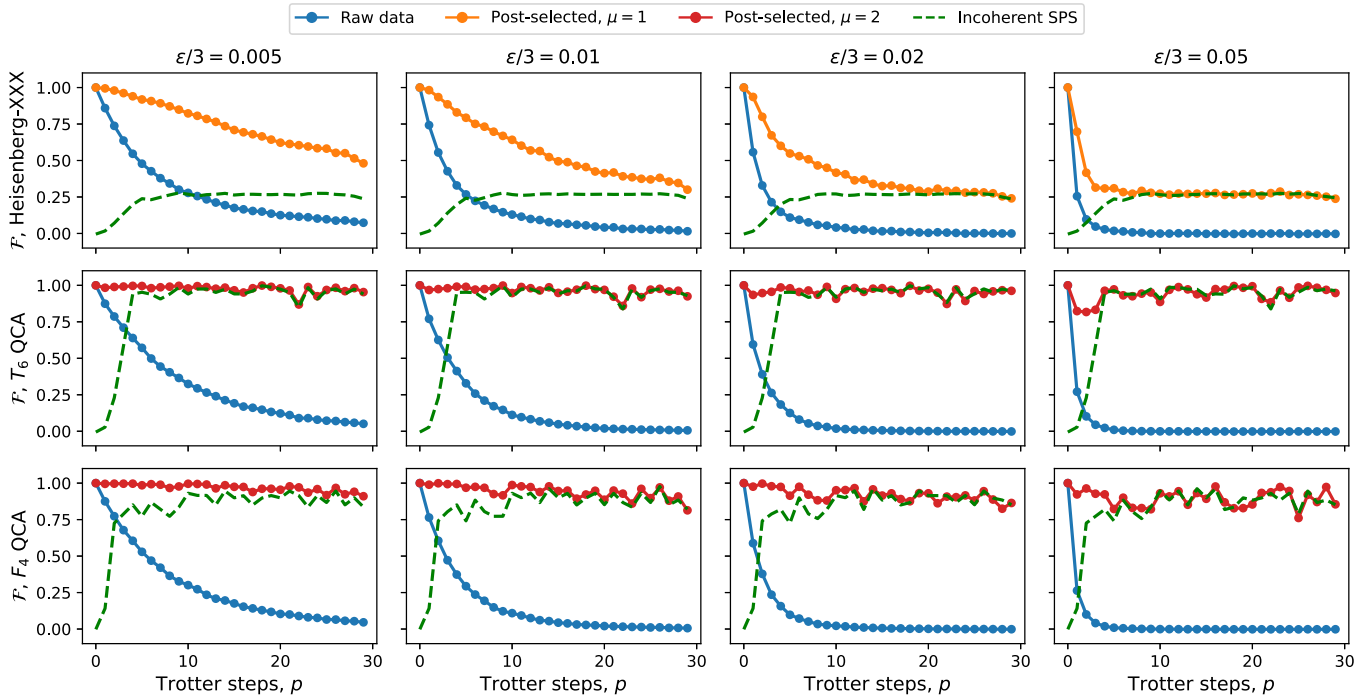


FIG. 6. The fidelities $\mathcal{F}(P_{\text{sim}}(p), P_{\text{ideal}}(p))$, $P_{\text{sim}} = P_{\text{raw}}, P_{\text{ps}, \mu=1}, P_{\text{ps}, \mu=2}, P_{\text{ISPS}}$ of each model (rows) at errors $\epsilon/3 = 0.005, 0.01, 0.02, 0.05$ (columns). There are 30 000 measurements at each Trotter step. The μ used is the smallest required to reach perfect shared subspace verification under Alg. V.1. This figure shows that with postselection, model fidelity remains above random noise at least as long as $p = 2 * n$.

where P_{IRN} is the incoherent random noise probability distribution

$$P_{\text{IRN}}(b) \equiv \frac{1}{d} \quad \forall |b\rangle \in B(\mathcal{H}_d), \quad (27)$$

which we expect to measure once noise has proliferated in the computation. Equation (26) will return 1 if $P_{\text{sim}} = P_{\text{ideal}}$, and 0 if $P_{\text{sim}} = P_{\text{IRN}}$. We compute Eq. (26) at each measurement layer in the computation, with P_{sim} being either the raw data P_{raw} or the postselected data P_{ps} .

We postselect by removing measurement results that fail under Alg. V.1 and Eq. (23) and renormalizing the probability distribution. If $P_{\text{raw}}(p)$ is the measurement distribution with noise at step p of M measurements, $P_{\text{ps}}(p)$ is the postselected measurement distribution, given by checking each state in $P_{\text{raw}}(p)$ in Alg. V.1, at the same Trotter step p with measurements $M_{\text{kept}} \leq M$.

B. Postselection with perfect symmetry-protected subspaces

We run a 15 qubit simulation of each of the three exemplar models, using initial conditions seen in the literature: The antiferromagnetic state $|\psi_0\rangle = |1010\dots 101\rangle$ for the Heisenberg-XXX model, a single bit flip on the middle qubit $|\psi_0\rangle = |0\dots 1\dots 0\rangle$ for the T_6 QCA [20,34], and two bit flips neighboring the center qubit $|\psi_0\rangle = |0\dots 101\dots 0\rangle$ for the F_4 QCA [34]. No false minima are seen past $\mu = 2$ in any of the simulations for these initial conditions, meaning postselection functions as intended. The data use $M = 30\,000$ measurements at each measurement layer and do not add any measurements to replace error-victim circuit runs.

Results can be seen in Fig. 6. We observe that postselected data approaches incoherent noise in the SPS of that simulation's initial state, which we define as

$$P_{\text{ISPS}}(b) \equiv \begin{cases} \frac{1}{|G|} & |b\rangle \in G \\ 0 & |b\rangle \notin G \end{cases}. \quad (28)$$

Because our simulation's raw data converges to random noise, which is a uniform probability distribution over the Hilbert space, and postselection only removes measurement outcomes, we should expect that the result of postselection is still a uniform probability distribution, just over the symmetry-protected subspace instead. In other words, we should observe that

$$\lim_{p \rightarrow \infty} \mathcal{F}(P_{\text{ps}}(p), P_{\text{ideal}}(p)) \approx \mathcal{F}(P_{\text{ISPS}}(p), P_{\text{ideal}}(p)) \quad (29)$$

given enough measurements.

C. Postselection with incorrect symmetry-protected subspaces

We show that even in subspaces of the F_4 QCA model where Alg. V.1 has failures at high μ , which is equivalent to false-positives mislabeling some basis vectors as *outside* the symmetry-protected subspace, our methods mitigate more errors than they introduce and thus still show promise for NISQ quantum simulations.

We found 18 symmetry-protected subspaces of the F_4 QCA that had false minima at $n = 15$ with $\mu \geq 9$ (henceforth called incomplete subspaces) and ran simulations to $p = 29$ with $M = 10\,000$ measurements at each cycle, initialized in one state from each rocky subspace. See Appendix D 4b for a list of these initial states. In these conditions, Alg. V.1 is highly

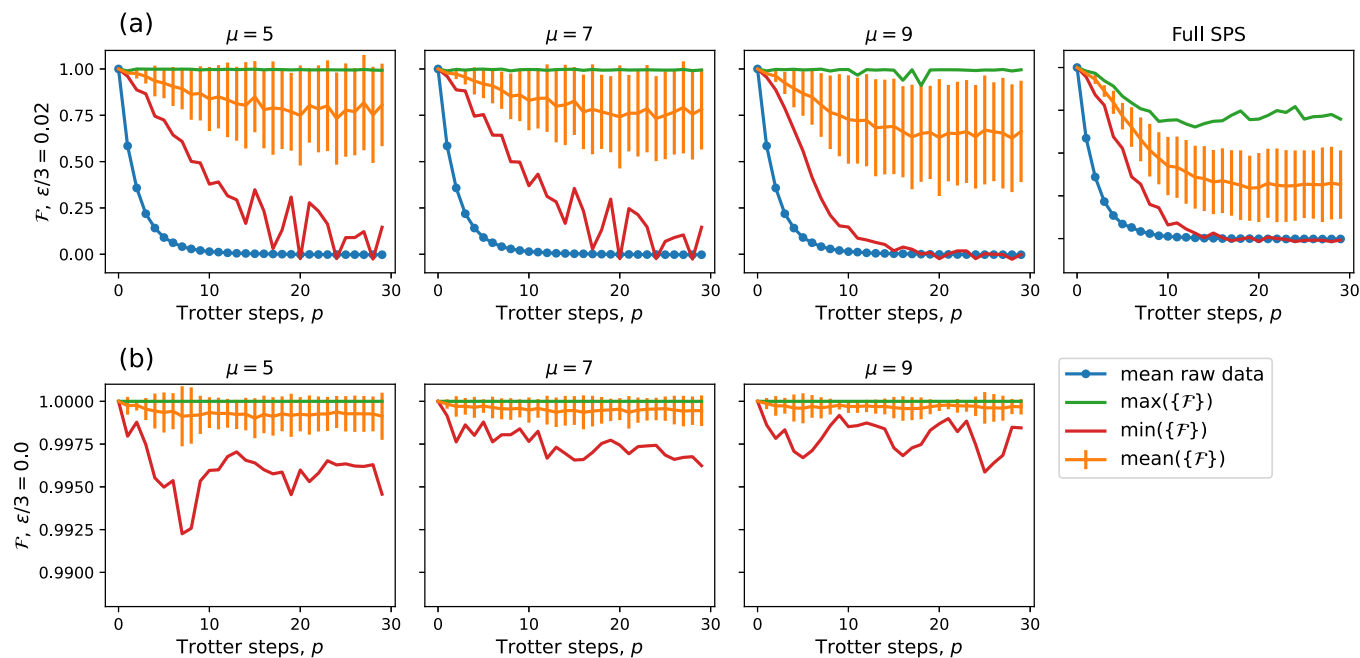


FIG. 7. The maximum, minimum, mean, and standard deviation of the set of fidelities, $\{F\}$, generated by simulating the F_4 QCA for an initial condition in each of its incomplete subspaces at $\mu = 9$ with postselected distributions $P_{\text{sim}} = P_{\text{ps},\mu=5}, P_{\text{ps},\mu=7}, P_{\text{ps},\mu=9}, P_{\text{SPS}}$. Each Trotter step has $M = 10\,000$ measurements. The error rate is (a) $\epsilon/3 = 0.02$ and (b) $\epsilon/3 = 0.0$ or an ideal simulation.

likely to meet false minima and reject measurements that were in the symmetry-protected subspace of the initial state. We calculate $\mathcal{F}(P_{\text{ps},\mu}(p), P_{\text{ideal}}(p))$ for each simulation, and run our postselection method with $\mu = 5, 7, 9$. See Fig. 7(a) for the set of fidelities $\{F\}$ at $\epsilon/3 = 0.02$, compared to postselection with the “Full SPS,” which is postselection using the full symmetry-protected subspace, instead of the heuristic; in the average case our postselection keeps the simulation data above incoherent noise, even with lower μ . The worst-case fidelity (in red) drops to the fidelity of incoherent noise, which indicates that our methods offer improvement only on particular simulations. However, the mean (orange) and standard deviation (orange bars) remain reliably higher than incoherent noise, with the maximum fidelity keeping close to the ideal simulation.

The other important observation is that our postselection, which is using an inaccurate symmetry-protected subspace, obtains a higher mean and maximum fidelity than postselection with a well-defined symmetry-protected subspace, with fidelity lowering as μ increases. We theorize the reason for this is as follows: if two states share a false minima with χ , they are likely close to each other in the string interaction graph $D_{\mathbb{L}}$, and thus have a higher probability of transitions between each other in the simulation. On the other hand, if a measured state does *not* share a false minima with the initial state, it is less likely to be observed in a simulation, but a noisy simulation will artificially amplify the probability of that state. The result is that postselection with the true SPS will accept these artificially amplified results, where our false SPS will coincidentally reject states that can possibly have this happen.

We also examine our postselection in a simulation absent any error; this is to see how much a false rejection can degrade a perfect computation. See Fig. 7(b) for postselection with an

error-free simulation. It can be seen that most fidelities remain above ≈ 0.999 , meaning most of the time this error-prone postselection method will have minimal impact even on an ideal simulation. The worst-case degraded fidelity is ≈ 0.992 .

VII. DISCUSSION AND CONCLUSION

We introduce a graph theory interpretation of quantum time evolution, which provides a theoretical framework through which symmetry-protected subspaces can be constructed via transitive closure. We identify that these invariant subspaces are an operator-free method for characterizing symmetry in the system, by indicating a conserved quantity of an initial state as it is manipulated by the dynamical system without needing to explicitly identify it. Along these lines, our approach complements recent work in open quantum systems where symmetry-protected density matrices were constructed without the knowledge of explicit symmetry operators [25], and other work to identify symmetry operators from brute force algorithms such as Algorithm 2 in [26], and machine learning methods [27–29].

We observe that a symmetry-protected subspace can be used to provide a smaller computational space, postselect noisy quantum simulation data, or be analyzed to deduce a symmetry operator. We identify postselection as a pertinent application and introduce two main classical algorithms that elucidate the features of a quantum system’s symmetry-protected subspaces. These algorithms employ a *basis string edit map*, which is an efficient construction to provide the local dynamics of an operator by focusing on the presence or absence of basis vectors through its action.

The first algorithm uses transitive closure, calculated with breadth-first search of these basis string edit maps, to enu-

merate every state within the symmetry-protected subspace of the initial state. Because postselection with this subspace requires constructing the entire SPS, which can still be exponentially difficult in the number of spins, we provide a second, polynomial-scaling algorithm. The second algorithm attempts to find the smallest basis state (by binary-encoded integer) in an SPS by following a locally optimal heuristic, and establishing that a path of basis string edit maps exists between an initial and final state if their respective searches collide at any elements.

We conclude by demonstrating postselection using our second algorithm, which shows that even when the raw simulation data degrade to incoherent noise, our methods effectively recover a probability distribution closer to an ideal computation. Our methods are compatible with any other error-mitigation technique compatible with postselection, such as zero noise extrapolation, and thus present a further addition to a growing array of techniques to improve noisy quantum computation [42].

We identify a few obvious extensions to this project. First, if Alg. IV.1 concludes with certain classical resources, this implies that the wave function can be stored with constant overhead on those same resources; this could shrink the computational memory requirements from the quantum regime to the classical regime. Second, we speculate that more reliable algorithms than Alg. V.1 may fulfill the same function; a string-matching algorithm similar to Needleman-Wunsch [43] with the basis string edit maps, instead of insert/delete/shift edits, may fulfill this function.

This work also presents interesting results on postselection that are worth further exploration. The first is the convergence to incoherent noise *within* a symmetry-protected subspace; this shows that, as expected, postselection will not converge to an ideal computation, regardless of the additional circuit runs to replace error-victim runs detected, as error will still have proliferated within the the SPS. However, such incoherent distributions still have significant complexity and nontrivial structure (see, e.g., [20]), making them particularly interesting in cases where the underlying symmetry is not analytically known. The second is the observation that misidentifying measurements within the SPS paradoxically results in a more accurate computation than what results from full knowledge of the SPS. The exact cause of this is up to speculation, and is likely a problem-dependent effect, but is interesting in its own right.

Finally, since the algorithms presented herein are specified with respect to a particular simulation or measurement basis, our methods are currently constrained to discovering subspaces that are protected either by a single symmetry generator in that basis or by a collection of Abelian generators, which are diagonal (or have diagonal representations) in the chosen basis. However, constraining many-body dynamics by sets of non-Abelian generators often yields rich physics as has been noted with respect to non-Abelian thermal states [44], bipartite entanglement entropy growth [45], eigenstate thermalization [46], and thermalization in finite-size quantum simulators [47,48]. As such, extending our subspace detection framework to subspaces that are protected by multiple non-Abelian generators would be a fruitful direction for future research.

Code is available at the public GitHub repository [49]. Data are available upon reasonable request to the corresponding authors.

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Superconducting Quantum Materials and Systems Center (SQMS) under Contract No. DE-AC02-07CH11359, and by National Science Foundation Grant No. PHY-1653820. This work was also authored in part by the National Renewable Energy Laboratory (NREL), operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. This work was supported by the Laboratory Directed Research and Development (LDRD) Program at NREL. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

APPENDIX A: PROOF OF ACCEPTABLE MAXIMAL COVERAGE IN SYMMETRY-PROTECTED SUBSPACES

Take the operator $P_G = \sum_{b \in G_{|\psi_0\rangle}} |b\rangle\langle b|$ to be the projection operator for the symmetry-protected subspace of $|\psi_0\rangle$:

$$[P_{G_{|\psi_0\rangle}}, U] = 0 \quad (\text{A1})$$

and suppose it has been constructed such that for each $|b\rangle \in G_{|\psi_0\rangle}$, $\exists t$ s.t. $\langle b|U(t)|\psi_0\rangle \neq 0$. Now suppose that the basis vector $|k\rangle$ is never seen by time evolution of U , such that

$$\langle k|U(t)|\psi_0\rangle = 0 \quad \forall t. \quad (\text{A2})$$

If we include $|k\rangle$ and the symmetry-protected subspace of $|k\rangle$ in our protected subspace, such that $G' = G_{|\psi_0\rangle} \cup G_{|k\rangle}$, then $P_{G'} = P_{G_{|\psi_0\rangle}} + \sum_{j \in G_{|k\rangle}} |j\rangle\langle j|$, and we get

$$[P_{G'}, U] = [P_{G_{|\psi_0\rangle}} + P_{G_{|k\rangle}}, U] \quad (\text{A3})$$

$$= [P_{G_{|\psi_0\rangle}}, U] + [P_{G_{|k\rangle}}, U] \quad (\text{A4})$$

$$= 0 \quad (\text{A5})$$

by Def. 1 of a symmetry-protected subspace for both $G_{|\psi_0\rangle}$ and $G_{|k\rangle}$. This shows that if a state which is never seen by the time evolution is included in the final symmetry-protected subspace, the projection operator still commutes, and thus the subspace still follows Def. 1.

APPENDIX B: PROOF OF THEOREM 1

We take $|\psi_0\rangle \in B$ to be a product state. Due to linearity, generalizing the proof to superposition states is trivial, given that one interprets $G_{|\psi_0\rangle}$ as being the union of all symmetry-protected subspaces of each of which corresponds to one (or more) basis states in the superposition. For notational

simplicity, and without loss of generality, we also assume an implicit operator ordering \mathcal{O}_{op} . Consider the transition amplitude

$$\begin{aligned} \langle b_f | U(t) | \psi_0 \rangle &= \langle b_f | \prod_{j=1}^p \prod_{i=1}^m U_i(\tau_j) | \psi_0 \rangle \\ &= \langle b_f | \prod_{j=1}^p \prod_{i=1}^m \sum_{|b_{j,i}\rangle \in B} |b_{j,i}\rangle \langle b_{j,i}| U_i(\tau_j) | \psi_0 \rangle, \end{aligned} \quad (\text{B1})$$

where in the second line of Eq. (B1) we have inserted the identity operator $\mathbb{1} = \sum_{|b_{j,i}\rangle \in B} |b_{j,i}\rangle \langle b_{j,i}|$ after each k -local unitary operator. Without loss of generality, we assume an operator ordering to expand and rearrange the right-hand side of Eq. (B1),

$$\begin{aligned} \langle b_f | U(t) | \psi_0 \rangle &= \sum_{|b_{p,m}\rangle \in B} \langle b_f | b_{p,m} \rangle \sum_{|b_{p,m-1}\rangle \in B} \langle b_{p,m} | U_m(\tau_p) | b_{p,m-1} \rangle \cdots \\ &\cdots \sum_{|b_{1,1}\rangle \in B} \langle b_{1,2} | U_2(\tau_1) | b_{1,1} \rangle \langle b_{1,1} | U_1(\tau_1) | \psi_0 \rangle. \end{aligned} \quad (\text{B2})$$

Next, we note that by the definition of the string edit map, Def. 4, and the presence of the local transition amplitude $\langle b_{1,1} | U_1(\tau_1) | \psi_0 \rangle$ in Eq. (B2), we can rewrite

$$\begin{aligned} &\sum_{|b_{1,1}\rangle \in B} \langle b_{1,2} | U_2(\tau_1) | b_{1,1} \rangle \langle b_{1,1} | U_1(\tau_1) | \psi_0 \rangle \\ &= \sum_{|b_{1,1}\rangle \in \mathcal{L}_{U_1}(|\psi_0\rangle)} \langle b_{1,2} | U_2(\tau_1) | b_{1,1} \rangle \langle b_{1,1} | U_1(\tau_1) | \psi_0 \rangle \\ &= \sum_{|b_{1,1}\rangle \in T_{|\psi_0}^1} \langle b_{1,2} | U_2(\tau_1) | b_{1,1} \rangle \langle b_{1,1} | U_1(\tau_1) | \psi_0 \rangle, \end{aligned} \quad (\text{B3})$$

where in the third line of Eq. (B3) we have used the fact that $\mathcal{L}_{U_1}(|\psi_0\rangle) \subseteq T_{|\psi_0}^1 \subseteq B$, which is to say, we can expand the summation to run over $T_{|\psi_0}^1$ by summing over terms where the local transition amplitude is zero. By the same logic, we can rewrite the second summation as

$$\begin{aligned} \sum_{|b_{1,2}\rangle \in B} (\cdots) &= \sum_{|b_{1,2}\rangle \in \mathcal{L}_{U_2}(|b_{1,1}\rangle) : |b_{1,1}\rangle \in \mathcal{L}_{U_1}(|\psi_0\rangle)} (\cdots) \\ &= \sum_{|b_{1,2}\rangle \in T_{|\psi_0}^2} (\cdots), \end{aligned} \quad (\text{B4})$$

where again, we have simplified notation by noting that $\{\mathcal{L}_{U_2}(|b_{1,1}\rangle) : |b_{1,1}\rangle \in \mathcal{L}_{U_1}(|\psi_0\rangle)\} \subseteq T_{|\psi_0}^2$ and summing over all additional states in $T_{|\psi_0}^2$ to which the transition amplitude is zero. Generally, one can rewrite summation $(j-1)m+i$ as

$$\sum_{|b_{j,i}\rangle \in B} (\cdots) = \sum_{|b_{j,i}\rangle \in T_{|\psi_0}^{(j-1)m+i}} (\cdots). \quad (\text{B5})$$

We therefore write the full transition amplitude as

$$\begin{aligned} \langle b_f | U(t) | \psi_0 \rangle &= \sum_{|b_{p,m}\rangle \in T_{|\psi_0}^{pm}} \langle b_f | b_{p,m} \rangle \sum_{|b_{p,m-1}\rangle \in T_{|\psi_0}^{pm-1}} \langle b_{p,m} | U_m(\tau_p) | b_{p,m-1} \rangle \\ &\cdots \sum_{|b_{1,1}\rangle \in T_{|\psi_0}^1} \langle b_{1,2} | U_2(\tau_1) | b_{1,1} \rangle \langle b_{1,1} | U_1(\tau_1) | \psi_0 \rangle. \end{aligned} \quad (\text{B6})$$

There are now two cases to consider. Let $T_{|\psi_0}^\ell$ be the iteration at which the stop condition in Eq. (17) is activated. Either $T_{|\psi_0}^{pm} = T_{|\psi_0}^\ell \equiv G_{|\psi_0}$, or else $T_{|\psi_0}^{pm} \subset T_{|\psi_0}^\ell$. In either case, $|b_f\rangle \notin G_{|\psi_0} \Rightarrow |b_f\rangle \notin T_{|\psi_0}^{pm}$, which means that $\langle b_f | b_{p,m} \rangle = 0 \forall |b_{p,m}\rangle \in T_{|\psi_0}^{pm}$ and the full transition amplitude must vanish. ■

APPENDIX C: ANALYSIS OF ALGORITHM V.1

1. Proof of exactness for \mathbb{L}_{SWAP} isomorphic systems

We will show that for a system U which has $\mathbb{L}_U = \mathbb{L}_{\text{SWAP}}$, iterative path creation with Eq. (22) (which is what Alg. V.1 computes) has only one stop condition when using $\mu = 1$, which is the binary encoded minima of the entire symmetry-protected subspace.

Assume our simulation of n qubits has substring edit maps $\mathbb{L}_{\text{SWAP}} \equiv \{\mathcal{L}_{\text{SWAP}_{0,1}}, \mathcal{L}_{\text{SWAP}_{1,2}}, \dots, \mathcal{L}_{\text{SWAP}_{n-2,n-1}}\}$. We take $|b_0\rangle$ to be the first state in the path, and it is a bitstring $|b_0\rangle = |0, 1\rangle^n$ such that $\sum_{i=0}^{n-1} a_i^\dagger a_i |b_0\rangle = s |b_0\rangle$. The smallest binary encoding belonging to a state in the symmetry-protected subspace of $|b_0\rangle$, which is the SPS encoding particle conservation symmetry, will belong to the bit string $\min(G_{|b_0}) = |1^s 0^{n-s}\rangle$. The bits are ordered left-to-right as least-to-most significant.

The string edit map $\mathcal{L}_{\text{SWAP}_{i,i+1}}$ acting on its local basis set $B_i = \{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ has the following mappings:

$$\begin{aligned} \mathcal{L}_{\text{SWAP}_{i,i+1}}(|00\rangle) &= \{|00\rangle\}, \\ \mathcal{L}_{\text{SWAP}_{i,i+1}}(|01\rangle) &= \{|10\rangle, |01\rangle\}, \\ \mathcal{L}_{\text{SWAP}_{i,i+1}}(|10\rangle) &= \{|01\rangle, |10\rangle\}, \\ \mathcal{L}_{\text{SWAP}_{i,i+1}}(|11\rangle) &= \{|11\rangle\}. \end{aligned} \quad (\text{C1})$$

When applied to a n -qubit state represented as an n -character bit string b , the mapping operates as, e.g.,

$$\begin{aligned} \mathcal{L}_{\text{SWAP}_{i,i+1}}(|b[0, i-1]; 01; b[i+2, n-1]\rangle) \\ = |b[0, i-1]; 10; [i+2, n-1]\rangle, \end{aligned} \quad (\text{C2})$$

where the “;” symbol means string concatenation and “ $b[i, j]$ ” is the substring of b from characters at index i through j (inclusive).

Given an incomplete search with $\chi(|b_0\rangle, \mu = 1)$, where $|b_j\rangle = \min(T_{|b_{j-1}}^1)$ is the most recent progression in the search, the next state $|b_{j+1}\rangle$ in the search is the smallest bit string in $\mathbb{L}_{\text{SWAP}}(|b_j\rangle)$, given by $|b_{j+1}\rangle = \min(T_{|b_j}^1)$; this will be $\mathcal{L}_{\text{SWAP}_{i,i+1}}(|b_j\rangle)$, where $|b_j[i+1]\rangle$ is the *most* significant bit equal to $|1\rangle$ such that $|b_j[i]\rangle = |0\rangle$.

In other words, our heuristic dictates that we will only choose $\mathcal{L}_{\text{SWAP},i,i+1}(|b_j\rangle)$ when $|b_j[i, i + 1]\rangle = |01\rangle$. Therefore, Alg. V.1 will continue until there is no substring “01” in b_j . For a bit string, the only configuration of bits that meets this condition is $b = 1^s 0^{n-s}$, which is the state $\min(G_{|b_0})$. Therefore, if Alg. V.1 halts, it must have found the minimum binary encoded state for that symmetry-protected subspace. ■

2. Max depth of search in \mathbb{L}_{SWAP}

Here we calculate the worst-case search depth for Alg. V.1 when using string edit maps \mathbb{L}_{SWAP} . Each time a new state $|b_{j+1}\rangle$ is traversed to with $|b_{j+1}\rangle = \min(T_{|b_j}^1)$ from Eq. (22), a “1” bit is swapped with a left-neighboring “0” bit, assuming bits are least-to-most significant left to right. We count each of these swaps as 1 step in the search, and this subsection will show an analytical form for the *most* steps ever required for an arbitrary bit string $|b_0\rangle$ of length n to reach the stop condition $|1^s 0^{n-s}\rangle$, as defined at the beginning of Appendix C 1. That section also shows that for the SWAP-isomorphic model we always get $\chi(|b_0\rangle, 1) = |1^s 0^{n-s}\rangle$. Note that, because $\mu = 1$ is used, the path length of edges in $D_{\mathbb{L}}$ between $|b_0\rangle$ and $|1^s 0^{n-s}\rangle$ is equal to the search depth, because $|Path| = \mu \times Depth = Depth$.

Suppose we have a symmetry-protected subspace of s many “1” bits and $n - s$ many “0” bits. The target state $\min(G_{|b_0})$ for a search with $\chi(|b_0\rangle, 1)$ is $\min(G_{|b_0}) = |1^s 0^{n-s}\rangle$. The longest path will start with $|b_0\rangle = |0^{n-s} 1^s\rangle$, since every other state is fewer SWAPs away from $\min(G_{|b_0})$.

Each $|1\rangle$ state in $|b_0\rangle$ requires $n - s$ swaps to get to its position in $|1^s 0^{n-s}\rangle$; therefore, in a system of s “|1” states, the search depth obeys the summation

$$Depth = \sum_{i=1}^s n - s, \tag{C3}$$

which simplifies to

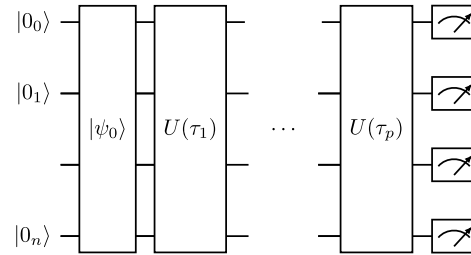
$$Depth = s(n - s). \tag{C4}$$

This quantity is maximized at $s = n/2$, which results in $\max(Depth) = \frac{n^2}{4}$.

APPENDIX D: SIMULATION INFORMATION

Our simulations all follow the same template: we take an initial state $|\psi_0\rangle$, which is a Z basis vector prepared by single-qubit Pauli-X gates. In our ideal simulations, we follow this with $p \in [0, 29]$ gate layers of gates to encode the dynamical system, followed by a parallel readout of all qubits in the Z basis with M measurements. This is shown in the circuit diagram below, where $|\psi_0\rangle$ is a computational basis state prepared by a set of X gates, each $U(\tau_j)$ is a single layer of the Trotterized circuit, at layer j , with parameters τ_j . Once

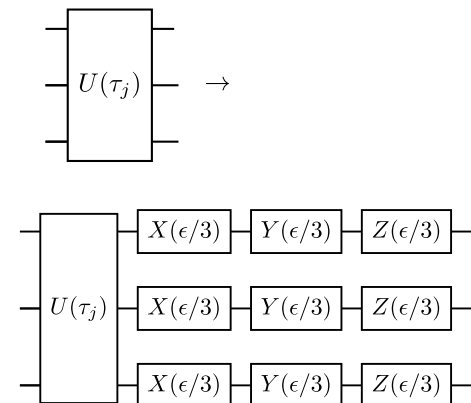
the system has been simulated up to t layers, we measure M times, and repeat for each $p \in [0, 29]$:



For a simulation with noise, we also added an “error layer” E after each $U(\tau_j)$, such that $U \rightarrow UE$. It is detailed below in Appendix D 1.

1. Simulation noise model

We use symmetric single-qubit depolarizing noise with probability $\epsilon/3$ after each Trotter layer in the circuit. The circuit diagram for this process at an arbitrary Trotter step j is



where a gate $X(\epsilon/3)$ is an X gate applied with probability $\epsilon/3$ and identity $\mathbb{1}$ applied with probability $1 - \epsilon/3$. In practice, we simulate this with density matrices through the following steps: if we have the wave function’s density matrix ρ from the last application of the Trotterized circuit layer $U(\tau_j)$, we apply the symmetric depolarizing noise with the equation

$$\rho \rightarrow (1 - \epsilon)\rho + \frac{\epsilon}{3}X_i\rho X_i + \frac{\epsilon}{3}Y_i\rho Y_i + \frac{\epsilon}{3}Z_i\rho Z_i \tag{D1}$$

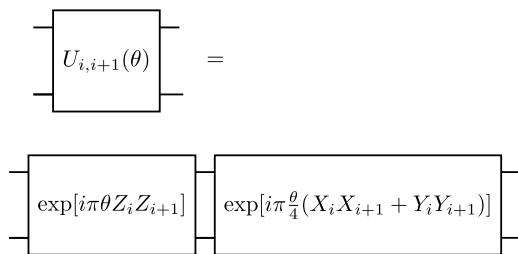
on each qubit i . As stated in Sec. VI, we use $\epsilon/3 = 0.005, 0.01, 0.02, 0.05$.

2. Heisenberg-XXX

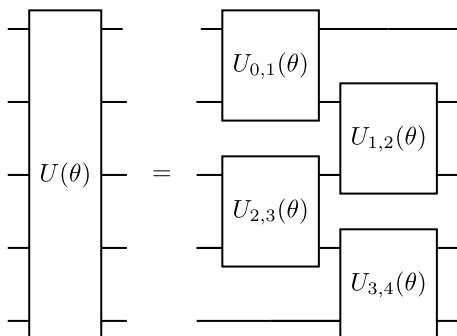
Here we give the simulation details for the Heisenberg-XXX model, with Hamiltonian given in Sec. II B 1 and simulation results in Sec. VI B.

a. Circuit implementation

The k -local unitary used ($k = 2$) is $U_{i,i+1} = \text{iSWAP}_{i,i+1}(\theta)Z_iZ_{i+1}(\theta)$, or



where i is the complex coefficient when multiplied and the qubit when indexed, using $\theta = 0.1$. We then do an even- and odd-layered Trotterization for one discrete layer of the simulation. This Trotterization on $n = 5$ is



for a single discrete time step. We use the initial condition $|\psi_0\rangle = |1010101010101\rangle$.

b. String edit maps

The string edit maps $\mathbb{L}_{\text{HeisXXX}}$ for this simulation are SWAP-isomorphic, with each local string edit map $\mathcal{L}_{\text{HeisXXX},i,i+1}$ having the behavior

$$\begin{aligned} \mathcal{L}_{\text{HeisXXX},i,i+1}(|00\rangle) &= \{|00\rangle\}, \\ \mathcal{L}_{\text{HeisXXX},i,i+1}(|01\rangle) &= \{|01\rangle, |10\rangle\}, \\ \mathcal{L}_{\text{HeisXXX},i,i+1}(|10\rangle) &= \{|01\rangle, |10\rangle\}, \\ \mathcal{L}_{\text{HeisXXX},i,i+1}(|11\rangle) &= \{|11\rangle\}. \end{aligned} \quad (\text{D2})$$

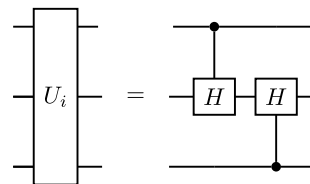
Each string edit [row in Eq. (D2)] obviously conserves $S = \sum_{j=i}^{i+1} Z_j$.

3. T_6 quantum cellular automata

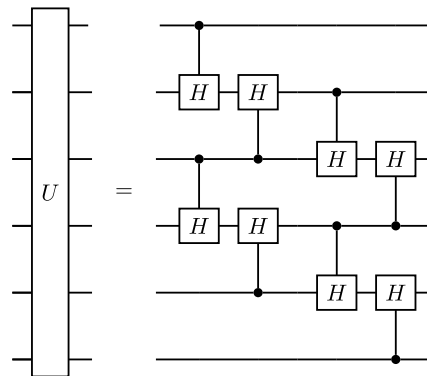
Here we give the simulation details of the T_6 quantum cellular automata, with model details in Sec. II B 2 and simulation results in Sec. VIB.

The k -local unitary used, ($k = 3$), is $U_i = P_{i-1}^{(1)}H_iP_{i+1}^{(0)} + P_{i-1}^{(0)}H_iP_{i+1}^{(1)} + P_{i-1}^{(1)}\mathbb{1}_iP_{i+1}^{(1)} + P_{i-1}^{(0)}\mathbb{1}_iP_{i+1}^{(0)} = \sum_{\alpha,\beta=0}^1 P_{i-1}^{(\alpha)} \otimes (H_i)^{\delta_{\alpha+\beta,1}} \otimes P_{i+1}^{(\beta)}$, where H_i is the Hadamard gate. This unitary

is implemented with the circuit



We implement this as a global unitary, which is applied once at each time step in the simulation by doing a parallel layer of all gates U_i where i is even, then a layer of all gates where i is odd.



We use only the initial condition $|\psi_0\rangle = |000000010000000\rangle$.

a. String edit maps

The string edit maps \mathbb{L}_{T_6} conserve the domain wall, and each local string edit map $\mathcal{L}_{T_6,i-1,i,i+1}$ has the behavior

$$\begin{aligned} \mathcal{L}_{T_6,i-1,i,i+1}(|000\rangle) &= \{|00\rangle\}, \\ \mathcal{L}_{T_6,i-1,i,i+1}(|001\rangle) &= \{|001\rangle, |011\rangle\}, \\ \mathcal{L}_{T_6,i-1,i,i+1}(|010\rangle) &= \{|010\rangle\}, \\ \mathcal{L}_{T_6,i-1,i,i+1}(|011\rangle) &= \{|001\rangle, |011\rangle\}, \\ \mathcal{L}_{T_6,i-1,i,i+1}(|100\rangle) &= \{|100\rangle, |110\rangle\}, \\ \mathcal{L}_{T_6,i-1,i,i+1}(|101\rangle) &= \{|101\rangle\}, \\ \mathcal{L}_{T_6,i-1,i,i+1}(|110\rangle) &= \{|100\rangle, |110\rangle\}, \\ \mathcal{L}_{T_6,i-1,i,i+1}(|111\rangle) &= \{|111\rangle\}, \end{aligned} \quad (\text{D3})$$

which conserves the symmetry operator $S = \sum_{j=i-2}^{i+1} Z_j Z_{j+1}$.

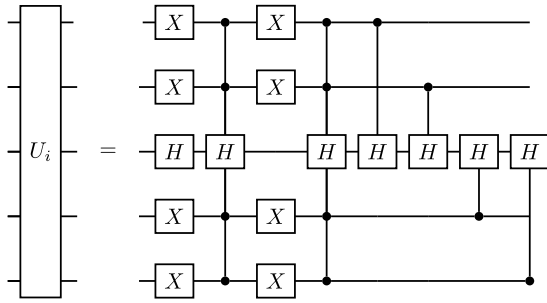
4. F_4 quantum cellular automata

Here we give the simulation details of the F_4 quantum cellular automata, with model details in Sec. II B 3 and simulation results in Sec. VIB and Sec. VIC.

The k -local unitary used ($k = 5$) is built on the local unitary

$$U_i = \sum_{\alpha,\beta,\gamma,\omega=0}^1 P_{i-2}^{\alpha} P_{i-1}^{\beta} (H_i)^{\delta_{\alpha+\beta+\gamma+\omega,2}} P_{i+1}^{\gamma} P_{i+2}^{\omega}, \quad (\text{D4})$$

which we implement through quantum gates with the circuit



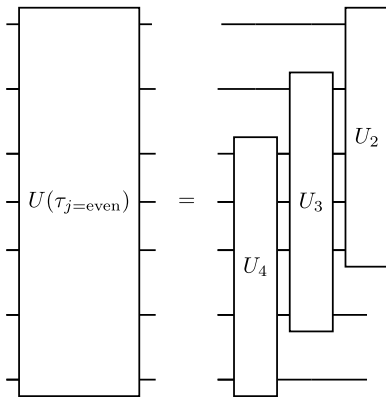
This local circuit is combined into a unitary applied at even discrete time steps

$$U(\tau_{j=\text{even}}) = \prod_{i=2,5,8,\dots} U_i(\tau_j) \prod_{i=3,6,9,\dots} U_i(\tau_j) \prod_{i=4,7,10,\dots} U_i(\tau_j), \tag{D5}$$

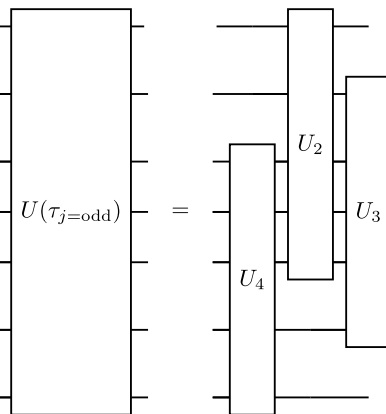
and a unitary applied at odd discrete time steps

$$U(\tau_{j=\text{odd}}) = \prod_{i=3,6,9,\dots} U_i(\tau_j) \prod_{i=2,5,8,\dots} U_i(\tau_j) \prod_{i=4,7,10,\dots} U_i(\tau_j). \tag{D6}$$

These unitaries on, for example, seven qubits have the circuit diagrams

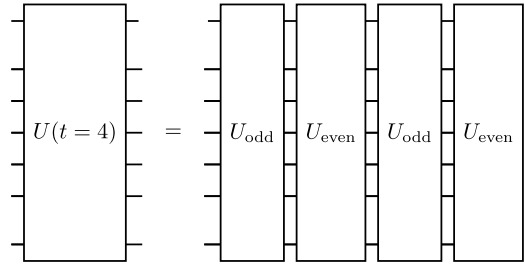


and



Notice that gates in the same product in Eq. (D5) and (D6), for example, U_4 and U_7 , commute because only their controls

overlap. To get to, for example, $t = 4$, our circuit would look like



a. String edit maps

Previously, no symmetry was known for the F_4 QCA. Below we outline the nonidentity behavior of each local string edit map $\mathcal{L}_{F_{4i}}$:

$$\begin{aligned} \mathcal{L}_{F_{4i}}(|00011\rangle) &= \{|00011\rangle, |00111\rangle\}, \\ \mathcal{L}_{F_{4i}}(|00111\rangle) &= \{|00011\rangle, |00111\rangle\}, \\ \mathcal{L}_{F_{4i}}(|01001\rangle) &= \{|01001\rangle, |01101\rangle\}, \\ \mathcal{L}_{F_{4i}}(|01101\rangle) &= \{|01001\rangle, |01101\rangle\}, \\ \mathcal{L}_{F_{4i}}(|01010\rangle) &= \{|01010\rangle, |01110\rangle\}, \\ \mathcal{L}_{F_{4i}}(|01110\rangle) &= \{|01010\rangle, |01110\rangle\}, \\ \mathcal{L}_{F_{4i}}(|11000\rangle) &= \{|11000\rangle, |11100\rangle\}, \\ \mathcal{L}_{F_{4i}}(|11100\rangle) &= \{|11000\rangle, |11100\rangle\}, \\ \mathcal{L}_{F_{4i}}(|10010\rangle) &= \{|10010\rangle, |10110\rangle\}, \\ \mathcal{L}_{F_{4i}}(|10110\rangle) &= \{|10010\rangle, |10110\rangle\}, \\ \mathcal{L}_{F_{4i}}(|10001\rangle) &= \{|10001\rangle, |10101\rangle\}, \\ \mathcal{L}_{F_{4i}}(|10101\rangle) &= \{|10001\rangle, |10101\rangle\}. \end{aligned} \tag{D7}$$

Any state $|b\rangle$ which does *not* appear above will have the behavior $\mathcal{L}_{F_{4i}}(|b\rangle) = \{|b\rangle\}$. It should also be understood that i is the middle qubit in the ordering above.

b. Initial conditions for simulations

We use a variety of initial conditions to test the imperfect path finding of Alg. V.1. Our initial condition $|\psi_0\rangle = |000000101000000\rangle$, tested in Sec. VIB, has no false rejections (in other words perfect path finding) with $\mu = 2$.

We also gather the set of symmetry-protected subspaces which have failed paths at $\mu = 9$, their average, best, and worst fidelities are shown in Sec. VIC. We chose an arbitrary state from each of these subspaces to be the initial condition; the states are

$$\begin{aligned} |\psi_0\rangle \in \{ & |001000011111111\rangle, |01110000001000\rangle, \\ & |011110101011111\rangle, |011110000010001\rangle, |011101000001000\rangle, \\ & |011000000101010\rangle, |001000011111111\rangle, |01110000001000\rangle, \\ & |011110101011111\rangle, |011110000010001\rangle, |011101000001000\rangle, \\ & |011000000101010\rangle, |111111101111100\rangle, |11011111110101\rangle, \\ & |111111101010111\rangle, |111100010111110\rangle, |110111111010001\rangle, \\ & |110101000000100\rangle\}. \end{aligned} \tag{D8}$$

- [1] E. Noether, Invariant variation problems, *Trans. Theo. and Stat. Phys.* **1**, 186 (1971).
- [2] M. K. Gaillard, P. D. Grannis, and F. J. Sciulli, The standard model of particle physics, *Rev. Mod. Phys.* **71**, S96 (1999).
- [3] V. V. Kozlov, Integrability and non-integrability in Hamiltonian mechanics, *Russian Math. Surve.* **38**, 1 (1983).
- [4] L. D. Landau, On the theory of phase transitions, *Zh. Eksp. Teor. Fiz.* **7**, 19 (1937).
- [5] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, *Phys. Rev. A* **86**, 032324 (2012).
- [6] X. Bonet-Monroig, R. Sagastizabal, M. Singh, and T. E. O'Brien, Low-cost error mitigation by symmetry verification, *Phys. Rev. A* **98**, 062339 (2018).
- [7] R. Sagastizabal, X. Bonet-Monroig, M. Singh, M. A. Rol, C. C. Bultink, X. Fu, C. H. Price, V. P. Ostroukh, N. Muthusubramanian, A. Bruno *et al.*, Experimental error mitigation via symmetry verification in a variational quantum eigensolver, *Phys. Rev. A* **100**, 010302(R) (2019).
- [8] J. R. McClean, M. E. Kimchi-Schwartz, J. Carter, and W. A. de Jong, Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states, *Phys. Rev. A* **95**, 042308 (2017).
- [9] Z. Cai, Multi-exponential error extrapolation and combining error mitigation techniques for NISQ applications, *npj Quantum Inf.* **7**, 80 (2021).
- [10] J. R. McClean, Z. Jiang, N. C. Rubin, R. Babbush, and H. Neven, Decoding quantum errors with subspace expansions, *Nat. Commun.* **11**, 636 (2020).
- [11] M. Larocca, F. Sauvage, F. M. Sbah, G. Verdon, P. J. Coles, and M. Cerezo, Group-invariant quantum machine learning, *PRX Quantum* **3**, 030341 (2022).
- [12] A. J. Daley, I. Bloch, C. Kokail, S. Flannigan, N. Pearson, M. Troyer, and P. Zoller, Practical quantum advantage in quantum simulation, *Nature (London)* **607**, 667 (2022).
- [13] R. Acharya, I. Aleiner, R. Allen, T. I. Andersen, M. Ansmann, F. Arute, K. Arya, A. Asfaw, J. Atalaya, R. Babbush *et al.*, Suppressing quantum errors by scaling a surface code logical qubit, [arXiv:2207.06431](https://arxiv.org/abs/2207.06431).
- [14] Z. Cai, R. Babbush, S. C. Benjamin, S. Endo, W. J. Huggins, Y. Li, J. R. McClean, and T. E. O'Brien, Quantum error mitigation, [arXiv:2210.00921](https://arxiv.org/abs/2210.00921).
- [15] Y. Quek, D. S. França, S. Khatri, J. J. Meyer, and J. Eisert, Exponentially tighter bounds on limitations of quantum error mitigation, [arXiv:2210.11505](https://arxiv.org/abs/2210.11505).
- [16] Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. D. Sawaya *et al.*, Quantum chemistry in the age of quantum computing, *Chem. Rev.* **119**, 10856 (2019).
- [17] R. Takagi, S. Endo, S. Minagawa, and M. Gu, Fundamental limits of quantum error mitigation, *npj Quantum Inf.* **8**, 114 (2022).
- [18] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, A. Bengtsson, S. Boixo, M. Broughton, B. B. Buckley *et al.*, Observation of separated dynamics of charge and spin in the Fermi-Hubbard model, [arXiv:2010.07965](https://arxiv.org/abs/2010.07965).
- [19] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, S. Boixo, M. Broughton, B. B. Buckley, D. A. Buell *et al.*, Hartree-Fock on a superconducting qubit quantum computer, *Science* **369**, 1084 (2020).
- [20] E. B. Jones, L. E. Hillberry, M. T. Jones, M. Fasihi, P. Roushan, Z. Jiang, A. Ho, C. Neill, E. Ostby, P. Graf *et al.*, Small-world complex network generation on a digital quantum processor, *Nat. Commun.* **13**, 4483 (2022).
- [21] D. V. Else, B. Bauer, and C. Nayak, Floquet Time Crystals, *Phys. Rev. Lett.* **117**, 090402 (2016).
- [22] T. Iadecola and T. H. Hsieh, Floquet Supersymmetry, *Phys. Rev. Lett.* **120**, 210603 (2018).
- [23] Y. Peng, Floquet higher-order topological insulators and superconductors with space-time symmetries, *Phys. Rev. Res.* **2**, 013124 (2020).
- [24] D. J. Yates and A. Mitra, Strong and almost strong modes of Floquet spin chains in Krylov subspaces, *Phys. Rev. B* **104**, 195121 (2021).
- [25] J. Thingna and D. Manzano, Degenerated Liouvillians and steady-state reduced density matrices, *Chaos* **31**, 073114 (2021).
- [26] T. Prosen, Chaos and complexity of quantum motion, *J. Phys. A: Math. Theor.* **40**, 7881 (2007).
- [27] K. Desai, B. Nachman, and J. Thaler, Symmetry discovery with deep learning, *Phys. Rev. D* **105**, 096031 (2022).
- [28] R. Bondesan and A. Lamacraft, Learning symmetries of classical integrable systems, [arXiv:1906.04645](https://arxiv.org/abs/1906.04645) [physics.comp-ph].
- [29] Z. Liu and M. Tegmark, Machine Learning Hidden Symmetries, *Phys. Rev. Lett.* **128**, 180201 (2022).
- [30] F. Hébert, G. G. Batrouni, R. T. Scalettar, G. Schmid, M. Troyer, and A. Dorneich, Quantum phase transitions in the two-dimensional hardcore boson model, *Phys. Rev. B* **65**, 014513 (2001).
- [31] E. R. Anschuetz, A. Bauer, B. T. Kiani, and S. Lloyd, Efficient classical algorithms for simulating symmetric quantum systems, [arXiv:2211.16998](https://arxiv.org/abs/2211.16998).
- [32] H. Radjavi and P. Rosenthal, *Invariant Subspaces* (Springer, Berlin, Heidelberg, 2003).
- [33] M. Suzuki, General theory of fractal path integrals with applications to many-body theories and statistical physics, *J. Math. Phys.* **32**, 400 (1991).
- [34] L. E. Hillberry, M. T. Jones, D. L. Vargas, P. Rall, N. Y. Halpern, N. Bao, S. Notarnicola, S. Montangero, and L. D. Carr, Entangled quantum cellular automata, physical complexity, and goldilocks rules, *Quantum Sci. Technol.* **6**, 045017 (2021).
- [35] R. Lidl, *Applied Abstract Algebra*, 2nd ed., Undergraduate Texts in Mathematics (Springer, New York, 1998).
- [36] M. J. Fischer and A. R. Meyer, Boolean matrix multiplication and transitive closure, in *IEEE 54th Annual Symposium on Foundations of Computer Science, Los Alamitos, CA* (IEEE Computer Society, Piscataway, NJ, 1971), pp. 129–131.
- [37] I. Marvian, Locality and conservation laws: How, in the presence of symmetry, locality restricts realizable unitaries, *Nat. Phys.* **18**, 283 (2022).
- [38] S. S. Skiena, Graph traversal, in *The Algorithm Design Manual*, edited by W. Wheeler and A. Wylde (Springer, London, 2008), pp. 145–184.
- [39] C. Y. Lee, An algorithm for path connections and its applications, *IRE Trans. Elect. Comput.* **EC-10**, 346 (1961).
- [40] S. Kullback and R. A. Leibler, On information and sufficiency, *Ann. Math. Stat.* **22**, 79 (1951).

- [41] E. Kapit, P. Roushan, C. Neill, S. Boixo, and V. Smelyanskiy, Entanglement and complexity of interacting qubits subject to asymmetric noise, *Phys. Rev. Res.* **2**, 043042 (2020).
- [42] 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).
- [43] S. B. Needleman and C. D. Wunsch, A general method applicable to the search for similarities in the amino acid sequence of two proteins, *J. Mol. Biol.* **48**, 443 (1970).
- [44] N. Yunger Halpern, P. Faist, J. Oppenheim, and A. Winter, Microcanonical and resource-theoretic derivations of the thermal state of a quantum system with noncommuting charges, *Nat. Commun.* **7**, 12051 (2016).
- [45] S. Majidy, A. Lasek, D. A. Huse, and N. Yunger Halpern, Non-abelian symmetry can increase entanglement entropy, *Phys. Rev. B* **107**, 045102 (2023).
- [46] C. Murthy, A. Babakhani, F. Iniguez, M. Srednicki, and N. Y. Halpern, Non-Abelian Eigenstate Thermalization Hypothesis, *Phys. Rev. Lett.* **130**, 140402 (2023).
- [47] N. Yunger Halpern, M. E. Beverland, and A. Kalev, Noncommuting conserved charges in quantum many-body thermalization, *Phys. Rev. E* **101**, 042117 (2020).
- [48] F. Kranzl, A. Lasek, M. K. Joshi, A. Kalev, R. Blatt, C. F. Roos, and N. Y. Halpern, Experimental observation of thermalisation with noncommuting charges, *PRX Quantum* **4**, 020318 (2023).
- [49] <https://github.com/NREL/SymmetryDetectionForQuantumSim>.