

Sequential quantum circuits as maps between gapped phases

Xie Chen^{1,2,3}, Arpit Dua^{2,3}, Michael Hermele⁴, David T. Stephen^{2,3,4}, Nathanan Tantivasadakarn,^{1,2}
Robijn Vanhove,² and Jing-Yu Zhao⁵

¹Walter Burke Institute for Theoretical Physics, Caltech, Pasadena, California 91125, USA

²Department of Physics, Caltech, Pasadena, California 91125, USA

³Institute for Quantum Information and Matter, Caltech, Pasadena, California 91125, USA

⁴Department of Physics and Center for Theory of Quantum Matter, University of Colorado, Boulder, Colorado 80309, USA

⁵Institute for Advanced Study, Tsinghua University, Beijing 100084, China



(Received 26 September 2023; accepted 16 January 2024; published 8 February 2024)

Finite-depth quantum circuits preserve the long-range entanglement structure in quantum states and map between states within a gapped phase. To map between states of different gapped phases, we can use *sequential quantum circuits*, which apply unitary transformations to local patches, strips, or other subregions of a system in a sequential way. The sequential structure of the circuit, on the one hand, preserves entanglement area law and hence the gappedness of the quantum states. On the other hand, the circuit has generically a linear depth, hence, it is capable of changing the long-range correlation and entanglement of quantum states and the phase they belong to. In this paper, we systematically discuss the definition, basic properties, and prototypical examples of sequential quantum circuits that map product states to Greenberger-Horne-Zeilinger states, symmetry-protected topological states, intrinsic topological states, and fracton states. We discuss the physical interpretation of the power of the circuits through connection to condensation, Kramers-Wannier duality, and the notion of foliation for fracton phases.

DOI: [10.1103/PhysRevB.109.075116](https://doi.org/10.1103/PhysRevB.109.075116)

I. INTRODUCTION

Understanding the nature of entanglement in ground-state wave functions has led to significant developments in the theoretical understanding and classification of quantum phases of matter. In particular, for zero-temperature gapped phases, it was understood that gapped ground states connected by a finite-depth local unitary quantum circuit (quantum circuits composed of a finite number of layers of nonoverlapping local unitaries) have the same “long-range entanglement” structure and are hence in the same phase [1]. This understanding has been helpful in the classification or systematic construction of gapped phases in various dimensions.

To map between ground states of different gapped phases, what kind of quantum circuits do we need? It has been proven in Ref. [2] that, to map from a product state (ground state of a trivial phase) to either the Greenberger-Horne-Zeilinger (GHZ) state (ground state of a symmetry-breaking phase) or a topological state, a quantum circuit of at least linear depth is needed. The intuition is simply that we need an “effort” or, more precisely, time that grows linearly with the total system size in order to establish the long-range correlation in the GHZ state or the long-range entanglement in the topological states. On the other hand, however, applying a generic linear depth circuit to a product state could easily lead to too much correlation and entanglement that cannot be accommodated in gapped ground states. Therefore, the question becomes as follows: Which subset of linear (or higher) depth circuits can map between gapped phases and not beyond?

The *sequential quantum circuit* [3–6] (SQC) is such a subset. As the name suggests, sequential quantum circuits are

local unitary quantum circuits that act sequentially. As shown in Fig. 1, a sequence of local unitary quantum gates or finite-depth quantum circuits can be applied to local patches, strips, annuli, or other lower-dimensional subregions in the system one at a time. To cover the whole system, the circuit depth scales as the number of subregions, which can be constant, linear, or higher in the linear size of the system. As the action of each layer is restricted to one subregion, it ensures that each local region in the system gets acted upon by only a finite number of local unitaries in the whole circuit. We use this as the defining feature of a sequential quantum circuit. That is, a *quantum circuit (composed of local unitary gates) is called a sequential quantum circuit if each local degree of freedom is only acted upon by a finite number of gates in the circuit.*

Finite-depth circuits are hence examples of sequential quantum circuits, although only a small subset. Sequential quantum circuits have been used to propose efficient generation methods for matrix product states and a large class of tensor product states are extensively discussed [3–6]. They also play an important role in various quantum protocols. See, for example, Refs. [7–9].

A direct consequence of the sequential structure is that if the initial quantum state satisfies the entanglement area law, the final state also satisfies the entanglement area law. Therefore, gapped systems remain gapped under the action of SQC. (When we say a state is gapped, it means the state is the gapped ground state of a local Hamiltonian.) On the other hand, the linear depth gives SQC the power to generate or change the long-range correlation and long-range entanglement in the quantum state, hence mapping between different quantum phases.

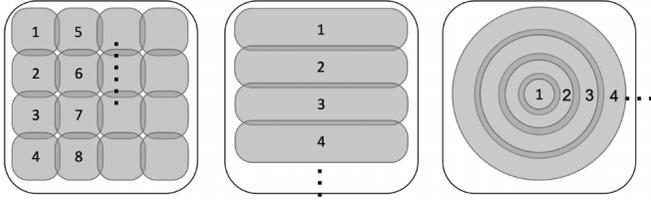


FIG. 1. Examples of sequential quantum circuits in 2D where (a) a sequence of local unitary gates are applied to local patches, or a sequence of finite depth quantum circuits are applied to (b) strips or (c) annuli.

In this paper, we discuss the sequential quantum circuit that generates various gapped quantum phases, including symmetry-breaking phases (Sec. II), symmetry-protected topological phases (Sec. III), (2 + 1)D topological phases (Sec. IV), (3 + 1)D topological phases and Walker-Wang models (Sec. V), and fracton phases (Sec. VI). In Sec. VII, we show that all locality-preserving unitary operators, i.e., quantum cellular automata, can be realized as sequential quantum circuits. We discuss circuits acting in different dimensions and in the presence of different symmetries. We note that some sequential circuits in the context of generating gapped phases of matter have been discussed in previous literature [3,10–13]. We interpret the power of the SQC by connecting the circuit action to Kramers-Wannier duality between symmetric and symmetry-breaking phases, condensation on gapped boundaries of topological and Walker-Wang models, the foliation structure of fracton models, etc. Note that to demonstrate that a sequential quantum circuit can connect a generic state in one gapped phase to a generic state in another gapped phase, we only need to show how to map between particular chosen states (usually fixed-point states) in the two phases. The mapping from a generic state to the fixed-point state in the same phase can be accomplished by an additional finite-depth circuit.

II. MAP TO SYMMETRY-BREAKING PHASES

In this section, we consider the mapping between symmetric and symmetry-breaking phases. We will first focus on the prototypical example of the (1 + 1)-dimensional [(1 + 1)D] transverse field Ising model before generalizing it to all dimensions and all finite-symmetry groups. We show that the circuit preserves the global Z_2 symmetry of the (1 + 1)D transverse field Ising model, but does not preserve locality.

The transverse field Ising model in (1 + 1)D,

$$H = -J \sum_i Z_i Z_{i+1} - B \sum_i X_i, \quad (1)$$

has a symmetric phase ($B > J > 0$) and a symmetry-breaking phase ($J > B > 0$) with respect to the global Z_2 symmetry $\prod_i X_i$. The fixed-point wave functions of the symmetry-preserving phase and the symmetry-breaking phase are

$$\begin{aligned} |\psi_{\text{SP}}\rangle &= |++++\dots\rangle, \\ |\psi_{\text{SB}}\rangle &= \frac{1}{\sqrt{2}}|00\dots 0\rangle + \frac{1}{\sqrt{2}}|11\dots 1\rangle, \end{aligned} \quad (2)$$

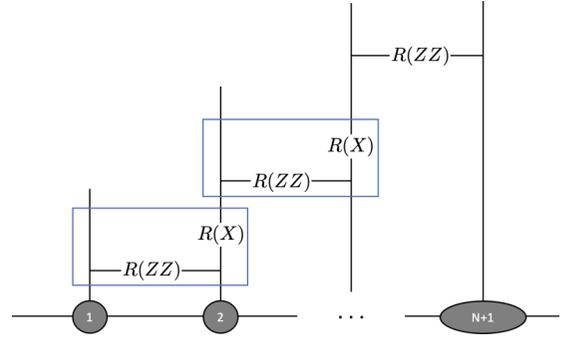


FIG. 2. Sequential circuit that maps from the symmetric state to the symmetry-breaking GHZ state of the (1 + 1)D transverse field Ising model. Each blue box represents the unitary $U_{i,i+1}$ in Eq. (8). Site $N + 1$ is the same as site 1.

where $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and we have chosen the symmetrized GHZ state to represent the symmetry-breaking ground space. To motivate the sequential circuit, it is insightful to map the above Hamiltonian to fermions using the Jordan-Wigner transformation

$$Z_i Z_{i+1} \rightarrow i\tilde{\gamma}_i \gamma_{i+1}, \quad (3)$$

$$X_i \rightarrow i\gamma_i \tilde{\gamma}_i, \quad (4)$$

where $\gamma_i = c_i + c_i^\dagger$ and $\tilde{\gamma}_i = -i(c_i - c_i^\dagger)$. Under such a mapping, the symmetric and symmetry-breaking phases map to the trivial and nontrivial Kitaev chain phases, respectively. It is now clear that the sequential circuit that maps between these two phases is composed of Majorana swaps [11],

$$\mathcal{U}_F = e^{\frac{\pi}{4}\tilde{\gamma}_N \gamma_1} \prod_{i=N-1}^1 e^{\frac{\pi}{4}\gamma_{i+1}\tilde{\gamma}_{i+1}} e^{\frac{\pi}{4}\tilde{\gamma}_i \gamma_{i+1}}. \quad (5)$$

Here, the ordering is chosen such that from right to left, the product goes from 1 to $N - 1$. Mapping back to the spin operators and defining,

$$R(\mathcal{O}) \equiv e^{-\frac{i\pi}{4}\mathcal{O}}, \quad (6)$$

the desired circuit is, therefore,

$$\mathcal{U} = R(Z_1 Z_N) \prod_{i=N-1}^1 U_{i,i+1}, \quad (7)$$

where

$$U_{i,i+1} = R(X_{i+1})R(Z_i Z_{i+1}), \quad (8)$$

as shown in Fig. 2.

We note that a convenient property for $R(\mathcal{O})$ is that for Pauli operators P and Q ,

$$R(Q)PR(Q)^\dagger = \begin{cases} P, & [P, Q] = 0 \\ iPQ, & \{P, Q\} = 0. \end{cases} \quad (9)$$

Using the above property, one can immediately see that conjugation by $U_{i,i+1}$ sends X_{i+1} to $Z_i Z_{i+1}$.

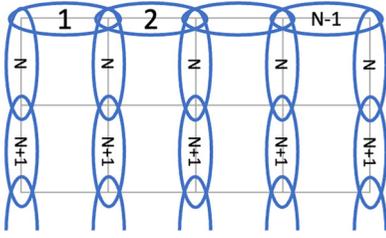


FIG. 3. Sequential circuit that maps from symmetric state to the symmetry-breaking GHZ state of the $(2 + 1)$ D transverse field Ising model on a $N \times M$ lattice. The blue ovals correspond to the gate set in the blue boxes in Fig. 2. The numbers in the ovals indicate the layer numbers in the sequential circuit. The horizontal gates act on the top layer only.

The complete mapping of operators using the unitary (7) is

$$\begin{aligned}
 X_1 &\rightarrow X_1 X_2 \dots X_{N-1} X_N Z_1 Z_N, \\
 X_i &\rightarrow Z_{i-1} Z_i, \quad i = 2, \dots, N \\
 X_1 X_2 \dots X_N &\rightarrow X_1 X_2 \dots X_N, \\
 Z_1 &\rightarrow Z_1, \\
 Z_i &\rightarrow X_i X_{i+1} \dots X_N Z_1, \quad i = 2 \dots N \\
 Z_i Z_{i+1} &\rightarrow X_i, \quad i = 2 \dots N \\
 Z_1 Z_2 &\rightarrow X_2 \dots X_N.
 \end{aligned} \tag{10}$$

Note that in the $\prod_i X_i = 1$ subspace, the mapping sends $X_i \rightarrow Z_{i-1} Z_i$ and $Z_i Z_{i+1} \rightarrow X_i$ as desired. A similar circuit has been discussed in Ref. [12].

We want to point out some important features of this circuit.

- (i) $|\psi_{SP}\rangle$ maps to $|\psi_{SB}\rangle$ and vice versa.
- (ii) In the bulk of the system, the transverse field term maps to the Ising term and vice versa. Near the two end points, the correspondence is lost, and some terms map to nonlocal terms.
- (iii) Applying the circuit twice generates translation by one site in the bulk of the system.
- (iv) The circuit is of linear depth, which saturates the lower bound required to generate long-range correlation in the GHZ state from a product state.
- (v) The circuit is symmetric. That is, the circuit is composed of local gates (gate sets in the blue boxes) that commute with the global Z_2 symmetry $\prod X_i$.
- (vi) The circuit is not locality preserving, as the operator X_1 maps to $Z_1 Z_N X_1 X_2 \dots X_N$. This is also necessary because it is known that locality-preserving unitaries in one spatial dimension are either finite-depth circuits or translation; neither is able to generate long-range correlation and map from symmetric to symmetry-breaking phases.¹

In Appendix A, we generalize the circuit to all finite groups. Higher-dimensional versions of the circuit can be built starting from the $(1 + 1)$ D version. For example, in $(2 + 1)$ D, to map from the symmetric state to the symmetry-breaking state with Z_2 symmetry, we can use the circuit shown in Fig. 3.

¹If we perform the Jordan-Wigner transformation and map the transverse field Ising model to the Majorana chain, the mapping between the two phases can be realized by translation by a single Majorana mode.

III. MAP TO SYMMETRY-PROTECTED TOPOLOGICAL PHASES

In this section, we construct symmetric, linear-depth SQCs that generate fixed points of symmetry-protected topological (SPT) phases from symmetric product states. We focus on the bosonic SPT phases described by group cohomology [14].

We begin with 1D SPT phases, which are described by an element $[\omega]$ of the second cohomology class $H^2(G, U(1))$ of the protecting symmetry G [15–17]. We can construct fixed-point states realizing such phases by picking a representative cocycle $\omega(g, h)$ and constructing a d -dimensional projective representation $V(g)$ such that $V(g)V(h) = \omega(g, h)V(gh)$ [15]. Define the d^2 -dimensional linear representation $U(g) = V(g) \otimes V(g)^*$ where $*$ is complex conjugation. Trivial and nontrivial representative fixed-point states with this symmetry can be defined on a chain of N sites, where the Hilbert space of each site has dimension d^2 and consists of two d -dimensional particles. In the trivial state, the two particles within each site are in the maximally entangled state $|\Omega\rangle = \sum_{i=0}^{d-1} |ii\rangle$, so the state is a product state. In the nontrivial state, the right particle of one site is maximally entangled with the left particle of the next site.² This is pictured in Fig. 4(a). These states are both symmetric under the symmetry $U(g)^{\otimes N}$. It was shown in Ref. [11] that these two states cannot be related by a finite-depth symmetric circuit, but that they can be related by a linear-depth symmetric circuit of SWAP gates, as pictured in Fig. 4(a). Each SWAP gate commutes with the symmetry $U(g)^{\otimes N}$, as only particles that transform under the representation $V(g)^*$ are swapped, so this is a linear-depth symmetric SQC mapping a trivial SPT state to a nontrivial SPT state.

It is interesting to consider the effect of truncating this SQC. Namely, if we stop applying gates at some point in the circuit, what state is left over? This is illustrated in Fig. 4(b), which shows that after applying some of the gates in Fig. 4(a), the resulting state is a nontrivial 1D SPT on periodic boundary conditions (PBC). This is necessary in order to ensure that the symmetry is preserved after each step. Each gate in the SQC serves to extend the region of the lattice occupied by the nontrivial SPT, and the circuit ends when this region is the entire lattice. This will contrast with the circuits we construct for topological phases in Sec. IV, in which truncating the SQC can result in a droplet of topological order with open boundaries and a particular gapped boundary to vacuum.

Now, we consider 2D SPT phases, focusing on one example and giving the general case in Appendix C. We consider the CZX model, which is a fixed-point representative of a 2D SPT phase with \mathbb{Z}_2 symmetry [18] and is defined on a square lattice with four qubits per site, as pictured in Fig. 4(c). Similar to the 1D fixed-point states, the qubits are divided into a tensor product of four-qubit entangled states $\frac{1}{\sqrt{2}}(|0000\rangle + |1111\rangle)$ which are contained within one site in the trivial SPT fixed-point and shared between four sites in the

²Note that even though the only difference in the two states is the choice of the unit cell, we care about the relative difference between the two states: if one is declared as a trivial product state, then the other is said to be nontrivial because the two cannot be related by a finite-depth quantum circuit.

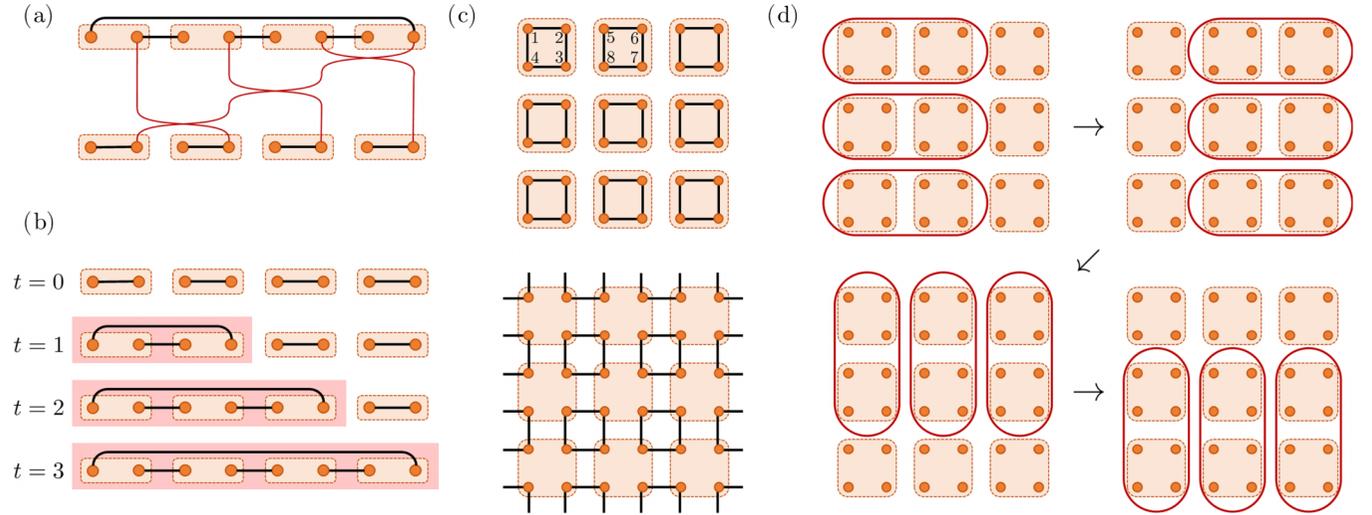


FIG. 4. (a) The trivial (bottom) and nontrivial (top) fixed-point states for 1D SPTs on a ring (periodic boundary condition in the 1D chain shown) of length 8. Circles indicate qubits, and two circles connected by a black line are in the maximally entangled state $|\Omega\rangle$. The red lines indicate the linear-depth circuit of SWAPs, which map between the two states. (b) The 1D states created after applying each of the three gates in (a) to the initial trivial state. The red region indicates the region spanned by the nontrivial SPT with periodic boundaries, which grow with each applied gate. (c) The trivial (upper) and nontrivial (lower) fixed-point 2D SPT states. The shaded regions depict single sites; solid lines indicate which qubits are entangled. (d) The linear-depth circuit which maps between the trivial and nontrivial 2D SPT states. Each ellipse indicates one application of SWAP^{CCZ}.

nontrivial SPT fixed point [see Fig. 4(c)]. The symmetry acts on each site as $U_{CZX} = CZ_{12}CZ_{23}CZ_{34}CZ_{41}X_1X_2X_3X_4$ where the four qubits within the site are numbered clockwise. It is straightforward to check that $U_{CZX}^2 = 1$ and that the trivial and nontrivial states are both symmetric under $U_{CZX}^{\otimes N}$ where the tensor product is over all N sites. Therefore, U_{CZX} defines a \mathbb{Z}_2 global symmetry of the models.

As these two states are in different phases with respect to the \mathbb{Z}_2 symmetry [18], there is no finite-depth symmetric circuit that relates them. To construct a linear-depth SQC, we might first try to mimic the 1D case and use SWAP gates to shift the four-qubit entangled states from within the sites to between the sites. This will map between the two states. However, in this case, the SWAP gates do not commute with the \mathbb{Z}_2 symmetry since they drag around the CZ gates involved in U_{CZX} . To remedy this, we can dress the SWAP gates with additional operations that restore the symmetry without affecting the action of the gate on the fixed-point states. Define the dressed SWAP gate acting on two adjacent sites as

$$\text{SWAP}^{CCZ} = \text{SWAP}_{26}\text{SWAP}_{37} \times \text{CCZ}_{265}\text{CCZ}_{261}\text{CCZ}_{378}\text{CCZ}_{374}, \quad (11)$$

where the qubits are numbered as in Fig. 4(c) and CCZ is the controlled-controlled- Z gate which acts on three qubits as $CCZ|ijk\rangle = (-1)^{ijk}|ijk\rangle$. These CCZ are placed such that the additional phase factors created by commuting the SWAP's in SWAP^{CCZ} past the CZ 's in U_{CZX} are canceled by phase factors created by commuting the CCZ 's in SWAP^{CCZ} past the X 's in U_{CZX} . One can indeed check that SWAP^{CCZ} acting on any neighboring pair of sites commutes with U_{CZX} , as is shown for a more general case in Appendix C.

Now we can use these symmetric SWAP^{CCZ} gates to map between the two states. Beginning with the trivial state, we first apply the gates sequentially between each pair of adjacent columns, noting that all gates within a given column can be performed in parallel such that the depth of this step is linear. Then, we apply 90°-rotated versions of the gates sequentially to each row, which also has linear depth. This results in the nontrivial SPT states. It is important that, at each step, the extra CCZ gates included in SWAP^{CCZ} cancel out pairwise when acting on the state, such that they do not affect the final state. This procedure is pictured in Fig. 4(d).

The general construction of dressing SWAP operators with additional phases to make symmetric gates is described in Appendix C for arbitrary 2D SPT phases. From that construction and our general understanding of fixed-point SPT states in all dimensions [14], we should be able to construct similar circuits for SPT phases within the group cohomology classification in all dimensions. Indeed, the construction of symmetric SQCs for quantum cellular automata (QCA), described in Sec. VII, can be applied to obtain symmetric SQCs for all SPT states that can be created via a (nonsymmetric) finite-depth quantum circuit (FDQC), which includes all group cohomology SPTs [14] and some beyond cohomology SPTs [19].

IV. MAP TO (2 + 1)D TOPOLOGICAL PHASES

In this section, we discuss the mapping from product states to (2 + 1)D topological states with gappable boundary: the string-net states [20]. We first discuss the circuit for generating toric code (TC) ground states and then generalize to all string nets. A key feature of our general construction is that we can write SQCs, which generate string-net models with arbitrary

gapped boundaries to vacuum, as well as models with periodic boundary conditions.

A. (2 + 1)D toric code

We start with the ground state of the (1 + 1)D toric code [21]. The model has qubit DOFs on edges of a square lattice, and the Hamiltonian is

$$H_{2\text{DTC}} = - \sum_{\text{vertex } e \ni \text{vertex}} \prod Z_e - \sum_{\text{plaquette } e \in \text{plaquette}} \prod X_e. \quad (12)$$

We discuss two distinct SQCs in detail. It is convenient to define the following notation for a unitary gate:

$$\begin{array}{c} \uparrow \\ \swarrow \quad \searrow \\ \downarrow \end{array} = \begin{array}{c} \circ \\ \downarrow \end{array} \quad (13)$$

where the arrows represent CNOT gates acting on the qubits on the edges of the lattice, with the arrow pointing from the control qubit to the target qubit. This gate will be generalized for string-net models in the next section.

The first circuit closely follows the construction given in Ref. [10], and is pictured in Fig. 5(a). The circuit consists of two distinct parts. First, we apply parallel gates to rows of the lattice sequentially. Each gate generates one plaquette term of $H_{2\text{DTC}}$. After each row of gates is applied, we are left with a TC state on a cylinder with periodic boundaries in the x direction and smooth [22] boundaries to vacuum (i.e., the qubits that are still in a product state) on the top and bottom edges. In other words, each row of gates pushes the gapped boundary one row further into the vacuum, expanding the region spanned by the topological order. The second part of the circuit involves the gates acting on the final row. We act on each plaquette in the final row (except the last) sequentially from left to right. This has the effect of “zipping” the smooth boundaries on the top and bottom of the cylinder, resulting

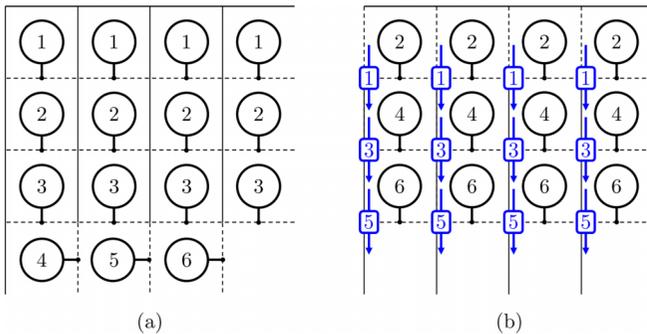


FIG. 5. Quantum circuits for creating toric code ground states with $L_x = L_y = 4$. The lattice has periodic boundary conditions in both directions. Each circle represents the gate defined in Eq. (13), while the blue arrows represent CNOT gates coupling two adjacent vertical edges. Qubits on solid (dashed) edges are initialized in the state $|0\rangle$ ($|+\rangle$), then the gates are applied in the order shown. (a) A circuit that leaves a smooth boundary when truncated. The gates in the bottom row “zip” the smooth boundaries together to create the periodic boundary conditions. (b) A circuit which leaves periodic boundaries when truncated.

in a TC ground state on the torus. The specific ground state that is obtained is the +1 eigenstate of the Z -type Wilson loops winding in both directions. The overall depth of the SQC scales like $O(L_x + L_y)$ where L_x, L_y are the lengths in the x and y directions. We can use a similar circuit on the dual lattice to prepare the eigenstate X -type Wilson loops in both directions, such that truncating the circuit results in a rough [22] boundary to vacuum.

The second circuit is pictured in Fig. 5(b). The key difference compared to the first circuit is that, after each step in the circuit, we are left with a TC state with periodic boundaries in both directions. That is, there is never a gapped boundary to vacuum. Instead, the first two steps of the circuit prepare a TC state on a thin torus, and each subsequent row of gates serves to extend the region over which the periodic toric code state is defined, similar to the circuits for SPT states defined in the previous section. Interestingly, because the final zipping step is not required (since the periodic boundaries in the vertical direction are present from the start), the depth of this circuit scales only as $O(L_y)$. Accordingly, the ground state of the TC that this circuit prepares is the +1 eigenstate of both the X - and Z -type Wilson loops that wrap around the torus in the y direction: the minimal entangled states on the x -direction cylinder [23]. We remark that the circuit in Fig. 5(b) can also be run such that all blue controlled-NOTs (CNOTs) are first applied in sequence, and then the plaquette operators in Eq. (13) are applied in sequence. The first step prepares a stack of 1D GHZ states, and the second step merges them into the TC ground state, with each step having depth $\sim L_y$.

B. Levin-Wen string-net models

In this section, we show how the construction in the previous section for the 2D TC can be generalized to arbitrary Levin-Wen string-net states [20], writing a sequential circuit that generates the bulk string-net ground state on a cylinder and on the torus. In the first case, the state is created row by row from a product state, creating and moving arbitrary gapped boundaries in the process like in the previous section. We generalize the unitary circuit that generates the “smooth” boundary at the ends of the cylinder constructed in Ref. [10] to arbitrary gapped boundaries. We show the general framework here and present the technical details in Appendix B. The general construction follows from Kitaev and Kong [22] and bears many resemblances to the (finite-depth) quantum circuit written in Ref. [24], mapping between different Morita equivalent string nets.

The Hilbert space of string-net models consists of configurations on a hexagonal lattice, with the edges labeled by simple objects a, b, \dots of a unitary fusion category \mathcal{C} . The simple objects obey fusion constraints: $a \times b = \sum_c N_{ab}^c c$, which are assumed to be multiplicity free here ($N_{ab}^c \in \{0, 1\}$). There exists a unit object $\mathbf{1}$, such that $\forall a, \mathbf{1} \times a = a$. The string-net Hamiltonian is a sum of commuting projectors, generalizing Eq. (12):

$$H_{\text{SN}} = - \sum_v A_v - \sum_p B_p. \quad (14)$$

In the ground state, the vertex terms (A_v) enforce the fusion rules at every vertex and the plaquette terms (B_p) give

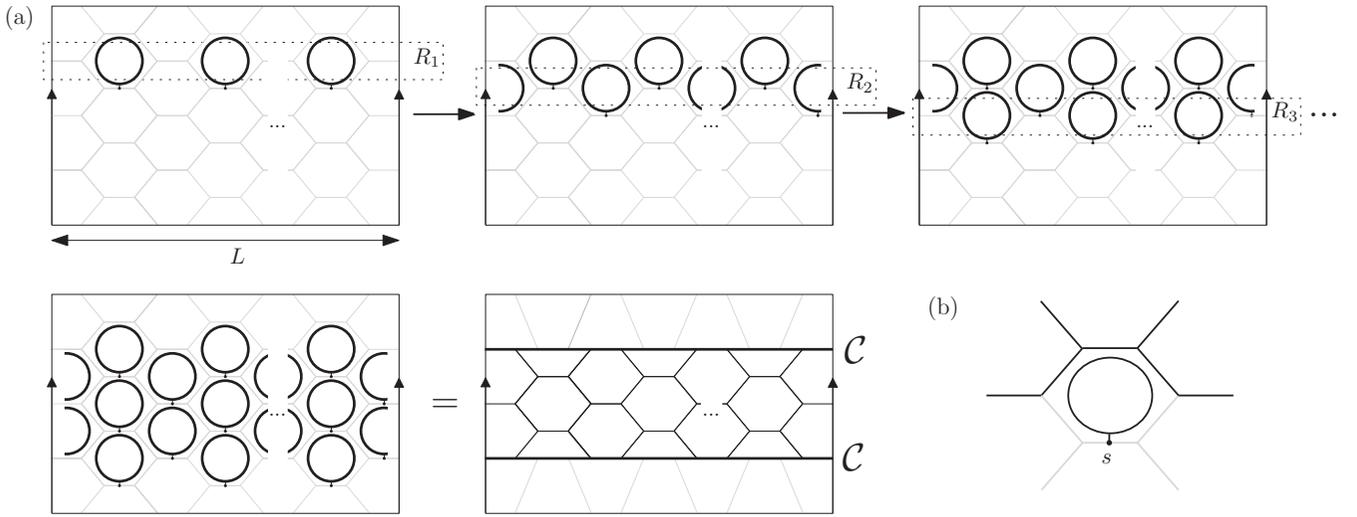


FIG. 6. (a) We start with a honeycomb lattice with periodic boundary conditions in the horizontal direction and all edges initialized on the trivial object $\mathbf{1}$ (gray). We generate a bulk string-net ground state with two boundaries labeled by \mathcal{C} (“smooth” boundary) by applying the controlled plaquette projector $C-B_p$ (17), sequentially, row by row. (b) In every step of the circuit, at least three of the edges are trivial initially, and the bottom edge serves as the control edge.

dynamics to the string net by projecting every plaquette on the trivial anyon sector

$$B_p = \sum_s \frac{d_s}{D} B_p^s, \quad (B_p)^2 = B_p, \quad (15)$$

with d_s the quantum dimensions of the simple objects s and $D = \sum_s (d_s)^2$ the total quantum dimension. The action of B_p can be schematically drawn as

$$B_p \left| \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \\ \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array} \right\rangle = \left| \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \\ \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array} \right\rangle. \quad (16)$$

After the weighted sum of s loops (15) is created in the middle of the plaquette, the loops are fused in the lattice and every

similar to the action of B_p^s , but with the control edge treated as in the trivial state [10]:

$$\begin{aligned} \left| \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \\ \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array} \right\rangle &= \sum_s \frac{d_s}{D} (C-B_p) \left| \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \\ \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array} \right\rangle = \\ \sum_s \frac{d_s}{D} B_p^s \left| \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \\ \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array} \right\rangle &= \sum_s \frac{d_s}{D} \left| \begin{array}{c} \diagup \quad \diagdown \\ \diagdown \quad \diagup \\ \diagup \quad \diagdown \\ \diagdown \quad \diagup \end{array} \right\rangle. \end{aligned} \quad (17)$$

The $C-B_p$ operator now acts as a unitary with the same action as B_p^s , provided at least one edge is in the trivial state initially (gray). This is shown in Refs. [10,25]. The proof relies on the unitarity of the F symbols of the fusion category \mathcal{C} (B2). The full circuit is shown in Fig. 6. A cylinder is

vertex is recoupled until the hexagonal basis states are recovered [Eq. (B13)]. An explicit ground state on the torus can be obtained by initializing all the edges on the trivial object, a state which trivially satisfies the vertex constraints, and then applying the plaquette operator on every plaquette. We remark that these plaquette operators are not unitary, so this does not directly lead to a circuit to prepare the ground state.

Similarly, to define our circuit, we start with an initial state with all edges fixed on the trivial object. We leave the lattice unoriented here for simplicity and choose an explicit orientation in Appendix B. Just like for the toric code, the circuit is constructed such that a control edge can be assigned for every individual plaquette operator. This edge is not yet entangled in the lattice and is still in the trivial state $|\mathbf{1}\rangle$. To merge the edge into the lattice, it is first mapped to $|\mathbf{1}\rangle \rightarrow \sum_s \frac{d_s}{D} |s\rangle$, after which it is used as the control to draw the loops around the added plaquette. The controlled- B_p ($C-B_p$) action is

generated with boundaries labeled by \mathcal{C} . This boundary is the generalization of the smooth boundary in the toric code for general string nets with Hamiltonian (14), where both the bulk and boundary edges are labeled by objects in \mathcal{C} . The effect of applying operators in a new row is to push

the boundary down and enlarge the bulk string-net ground state.

Following Kitaev and Kong, the string-net boundaries with edges labeled by objects in \mathcal{C} are not the only possible ground-state solution of Eq. (14). More generally, the edges can carry labels in a \mathcal{C} -module category \mathcal{M} (indicated by blue lines in the diagrams). Given the input category \mathcal{C} and the choice of boundary \mathcal{M} , it is always possible to define the ‘‘Morita dual’’ category $\mathcal{C}_{\mathcal{M}}^*$ and consider \mathcal{M} as a $(\mathcal{C}_{\mathcal{M}}^*, \mathcal{C})$ -bimodule category [26]. This is the structure we require to define our general quantum circuit, the details of which are explained in Appendix B.

The opposite category $\overline{\mathcal{M}}$ is a $(\mathcal{C}, \mathcal{C}_{\mathcal{M}}^*)$ -bimodule category, with objects that are in one-to-one correspondence with those of \mathcal{M} , but with their orientation reversed. One can now construct a weighted sum of simple objects in \mathcal{M} ($B_p^{\mathcal{M}}$):

$$B_p^{\mathcal{M}} \left| \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} \right\rangle = \left| \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} \right\rangle, \quad (18)$$

$$B_p^{\mathcal{M}} = \sum_s \frac{d_s}{D} B_p^{s, \mathcal{M}}, \quad (19)$$

with d_s the quantum dimensions of the objects in \mathcal{M} . When applied to all plaquettes, $B_p^{s, \mathcal{M}}$ maps the ground state of a string net with input category $\mathcal{C}_{\mathcal{M}}^*$ to that of a string net with input category \mathcal{C} (the operator $B_p^{\overline{\mathcal{M}}}$ does the reverse).

The initial trivial state, on which our general sequential circuit will act, is interpreted as a state with all edges fixed on the trivial object in the Morita dual category $\mathcal{C}_{\mathcal{M}}^*$. The objects in \mathcal{M} have an action on the objects in $\mathcal{C}_{\mathcal{M}}^*$ ($\mathcal{C}_{\mathcal{M}}^* \times \mathcal{M} \rightarrow \mathcal{M}$), such that on the boundary, the loops are fused with the trivial edges ($\in \mathcal{C}_{\mathcal{M}}^*$) to obtain boundaries with \mathcal{M} labels. In the bulk, loops are fused pairwise on every edge $\overline{\mathcal{M}} \times \mathcal{M} \rightarrow \mathcal{C}$ [by virtue of (B11)], such that the desired bulk string net \mathcal{C} is recovered. The general sequential circuit is shown in Fig. 7. It is the generalization of the one shown in Fig. 6 for arbitrary gapped boundaries labeled by \mathcal{M} . The circuit is slightly more complicated as we need different plaquette operators on different sublattices for the first two rows. The general picture, however, is very similar. The same isometry argument can be invoked here for the general plaquette operator, the proof of which we omit, but is a generalization of the one given in Refs. [10,25], and relies on the unitarity of all the F symbols involved (see Appendix B).

Note that \mathcal{C} is always a valid choice for the module category over itself ($\mathcal{M} = \mathcal{C}$), in which case the Morita dual is simply \mathcal{C} itself. This case corresponds to the original circuit and the smooth boundary is recovered. The toric code is recovered from this general picture by choosing $\mathcal{C} = \text{Vec}_{\mathbb{Z}_2}$. The ‘‘smooth’’ and ‘‘rough’’ boundaries are produced by the sequential circuit by choosing $\mathcal{M} = \mathcal{C} = \text{Vec}_{\mathbb{Z}_2}$ and $\mathcal{M} = \text{Vec}$ (the category of vector spaces, with only one trivial object), respectively [27].

We finish this section by showing how the circuit needs to be adapted for generating the string-net ground state on the torus. We can try to close the cylinder in Fig. 7 by applying

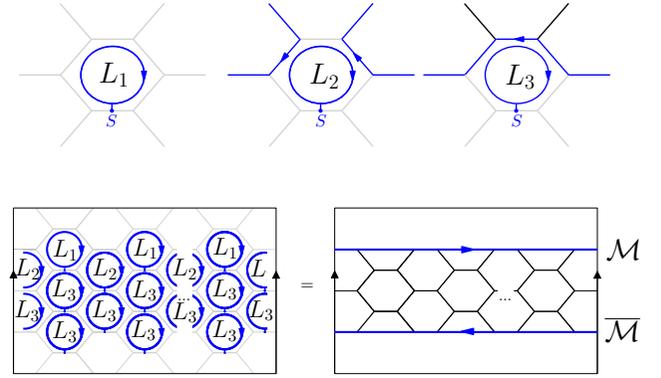


FIG. 7. The general sequential circuit to generate the string-net ground state with boundaries labeled by \mathcal{M} by acting row by row on the plaquettes with a controlled- $B_p^{\mathcal{M}}$. We start with an initial state with all edges fixed on the trivial object in the category $\mathcal{C}_{\mathcal{M}}^*$ (gray). We need three different plaquette operators: L_1 and L_2 for the first two rows and L_3 for all rows below that. The precise action of the three operators is shown in Eqs. (B13)–(B15). In the bulk, the loops are fused in the lattice to the string-net labeled by \mathcal{C} [by virtue of Eq. (B11)], but the boundary edges are labeled by objects in \mathcal{M} (top) and $\overline{\mathcal{M}}$ (bottom).

a new row with periodic boundary conditions, using the side edges as the control edges this time (because the bottom edges of the new row are already occupied after the circuit for the first row). However, we cannot completely close the cylinder, as there are no unoccupied edges left to use as the control on the very last plaquette. The solution is explained in Ref. [25], and we use a generalization of it here to construct the torus ground state in different representations labeled by \mathcal{M} . The string-net ground state is created on a minimal torus first (a torus with two vertices, one plaquette, and three edges) by initializing the edges on the trivial object in \mathcal{C} and applying the plaquette operator in the reverse direction of Eq. (19) on the one plaquette [Eq. (B20)]. The result is a minimal torus ground state of a string-net with input category $\mathcal{C}_{\mathcal{M}}^*$. After this initial step, the original sequential circuit is now used to add plaquettes to obtain the ground state for a string net with input category \mathcal{C} in the bulk. For the last row, the circuit needs to be altered, using side edges as control qudits similar to the toric code circuit. The plaquette operator can now be applied on the last plaquette since we are guaranteed to be in the torus ground state already. The procedure is shown in Fig. 8. The general circuit can be used to generate a projected entangled pair state (PEPS) representation of the string-net ground state, in which case the representation from Ref. [27] is recovered, generalizing the original representation from Refs. [28,29]. Note that in the Abelian group case, the minimal torus is trivial, and for $\mathcal{M} = \text{Vec}_{\mathbb{Z}_2}$ we recover the toric code sequential circuit (on a hexagonal lattice).

Just like for the toric code in Fig. 5(b), we can change the string-net circuit such that after each step, we are left with a string net with periodic boundary conditions in both directions (without any open boundaries). The periodic boundary condition in the vertical direction can be enforced in the circuit without applying a last row that zips the boundaries together

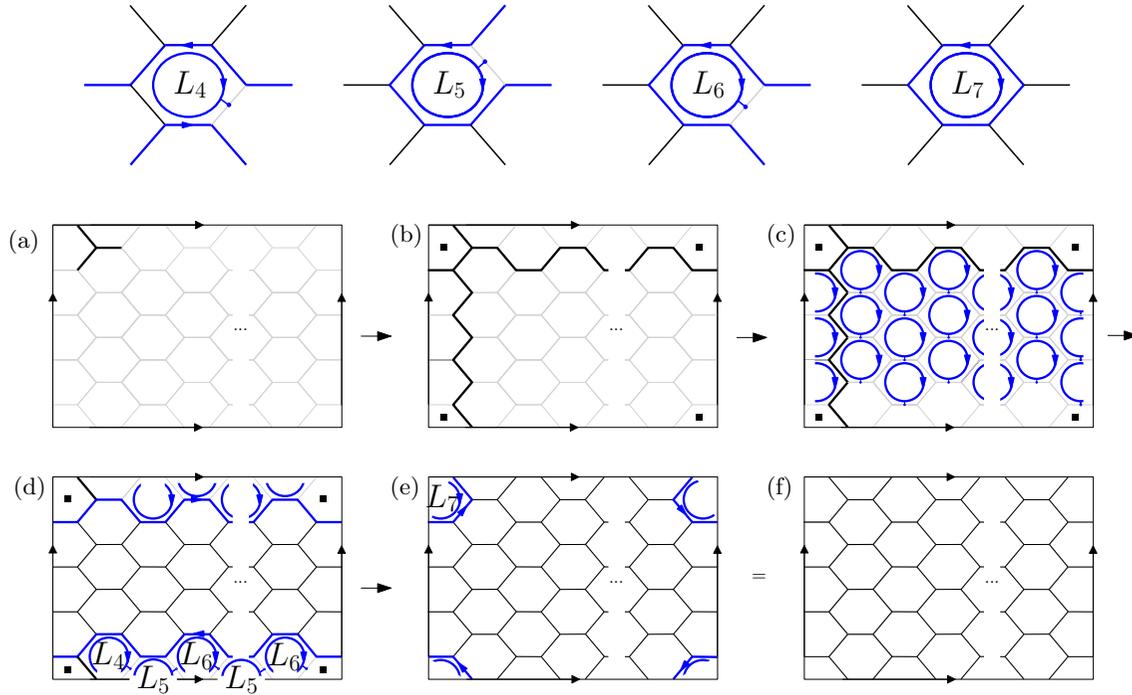


FIG. 8. The procedure for generating the string-net ground state on the torus. (a) The ground state of the string net with input category $C_{\mathcal{M}}^*$ is created on the minimal torus by fixing three edges according to Eq. (B20). (b) The minimal torus is expanded by copying the initial edges along the two noncontractible cycles of the torus (this is a sequential circuit). The black boxes identify the same plaquette at the four corners. (c) The sequential circuit is used to generate the bulk ground state (using the bottom edge as the control) using Eqs. (B13)–(B15). (d) For the last row, the circuit needs to be altered, using a side edge as the control [Eqs. (B16)–(B18)]. (e), (f) The plaquette operator (B19) can now be applied on the last plaquette to obtain the torus ground state. This operator is now unitary since we are guaranteed to be in the torus ground state already.

by first initializing a one-row torus ground state and enlarging the torus in every step.

V. MAP TO (3 + 1)D TOPOLOGICAL PHASES AND WALKER-WANG MODELS

We now describe SQCs for preparing topological phases in (3 + 1)D. For the (3 + 1)-dimensional [(3 + 1)D] toric code, there are two ways to view the bulk wave function: either as a condensate (equal-weighted superposition) of membranelike excitations or as a condensate of looplike excitations [30]. Correspondingly, there are two types of gapped boundary: a rough boundary that condenses the charges and a smooth boundary that condenses the looplike fluxes. In this section, we discuss two sequential quantum circuits for generating the (3 + 1)D toric code from product states. The first one generates a membrane condensate in the bulk with a rough boundary, while the second one generates a loop condensate in the bulk with a smooth boundary. The first type of circuit can be generalized to other (3 + 1)D Dijkgraaf-Witten gauge theories. In contrast, the second type of circuit can be generalized to other Walker-Wang models, as we discuss later in this section.

A. Point-charge condensed boundary

Consider the (3 + 1)D toric code defined on a cubic lattice. In the membrane condensate picture, the Z_2 DOFs are on each

plaquette. The Hamiltonian contains two types of terms, one associated with each cube and one associated with each edge:

$$H_{\text{3DTC}} = - \sum_{\text{cube } p \in \text{cube}} X_p - \sum_{\text{edge } p \in \text{edge}} Z_p. \quad (20)$$

Starting from a product state Hamiltonian $H = - \sum_p Z_p$, the toric code can be generated with a sequential circuit as shown in Fig. 9. Within each cube [Fig. 9(a)], the bottom

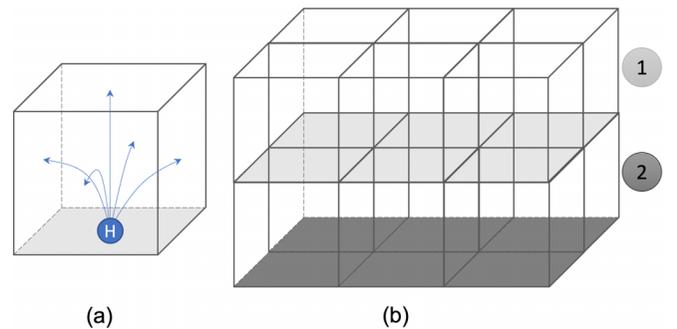


FIG. 9. Sequential circuit that generates the (3 + 1)D toric code with charge condensed boundary. (a) Within each cube, the bottom plaquette is first transformed by a Hadamard gate and then used as the control qubit for a set of controlled-NOT operations targeting all the other plaquettes in the same cube. (b) Such operations are applied to all cubes in layer 1 first, then to layer 2 beneath it, and so on.

plaquette is first transformed by a Hadamard gate that exchanges X_p and Z_p . It is then used as the control qubit for a set of controlled-NOT operations targeting all the other plaquettes in the same cube. After these operations, the Z_p term on the bottom plaquette gets mapped to the cube term $\prod_{p \in \text{cube}} X_p$ in the toric code Hamiltonian. The sequential circuit acts by applying this transformation first to all the cubes (at the same time) in layer 1, then to all the cubes in layer 2 underneath layer 1, and so on. This circuit is a direct generalization of the circuit generating 2D toric code discussed in Sec. IV.

Applying the circuit up to layer n generates a gapped boundary between the $(3+1)$ D toric code and the vacuum state. It is easy to see that this is the rough boundary of the toric code (rough in the dual lattice) where the gauge charge excitation condenses.

The sequential circuit described in Fig. 9 can be generalized to all Dijkgraaf-Witten (DW) gauge theories in $(3+1)$ D. DW gauge theories are membrane condensates in $(3+1)$ D where different membrane configurations can come with different phase factors (for toric code the phase factors are all 1). To generate the membrane condensate, we can generalize the sequential circuit in Fig. 9 from Z_2 DOF to group G labeled DOF and then supplement each elementary closed membrane generation operation in (a) with the appropriate phase factor. Similar to the string-net circuit discussed in Sec. IV, the elementary closed membrane generation operations in the same layer can be shown to commute with each other. Therefore, the DW states can be generated layer by layer with a linear depth circuit. If the circuit is terminated at a particular layer, the gapped boundary is a condensate of the bosonic gauge charges, just like in the toric code case.

B. Flux-loop condensed boundary

A different sequential circuit can give rise to the flux-loop condensed boundary for the $(3+1)$ D toric code. The circuit generates a loop condensate in the bulk. To describe this circuit, we take the dual lattice of that in Fig. 9 so that DOFs are associated with each edge. The toric code Hamiltonian is then expressed as

$$H_{3\text{DTC}} = - \sum_{\text{vertex}} \prod_{e \ni \text{vertex}} X_e - \sum_{\text{plaquette}} \prod_{e \in \text{plaquette}} Z_e. \quad (21)$$

We start with a trivial product state with Hamiltonian $H = - \sum X_e$. The circuit to obtain the toric code is shown in Fig. 10. Within each plaquette [Fig. 10(a)], a Hadamard gate is first applied to a chosen edge (the thick black one), which is then used as the target for a set of controlled-NOT gates with all the other edges in the plaquette as the control. After these operations, the X_e term on the target edge gets mapped to the plaquette term $\prod_{e \in \text{plaquette}} Z_e$ in the toric code Hamiltonian. The sequential circuit acts by applying this transformation first to all the plaquettes in the first row on the bottom surface with the black edges as the target, then to all the plaquettes in the second row with the gray edges as the target, and so on for all N rows in the bottom surface. Then to move up, apply the transformation in Fig. 10(a) to all the vertical plaquettes immediately above the bottom surface with the orange edges as the target in step $N+1$. This generates the plaquette term for all the involved vertical plaquettes. After this step, the

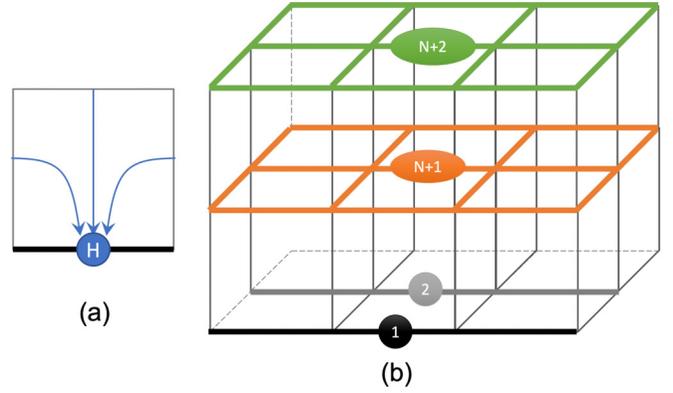


FIG. 10. Sequential circuit that generates the $(3+1)$ D toric code with flux-loop condensed boundary. (a) Within each plaquette, a Hadamard gate is first applied to a chosen edge (the thick black one), which is then used as the target for a set of controlled-NOT gates with all the other edges in the plaquette as the control. (b) Such operations are first applied row by row to all the plaquettes in the bottom surface.

plaquette terms on the orange plane are automatically satisfied because the product of the six plaquette terms around a cube is the identity. Therefore, repeating step $N+1$ allows us to expand the topological region and push the boundary upward.

It can be checked that if the sequential circuit is terminated at step $N+m$, the gapped boundary at the m th plane is the smooth boundary type where the flux loop condenses.

C. Walker-Wang with loop condensed boundary

The circuit in Fig. 10 can be generalized to construct all Walker-Wang (WW) models in $(3+1)$ D [31]. The ground state of any WW model is a loop condensate, although the loops are more complicated than that in the toric code. To construct the ground-state wave function from a product state, we can choose a target edge in each plaquette and use it to draw loops in each plaquette in a way similar to that described in Sec. IV for the string nets. The only difference is that, in the WW wave functions, when we draw loops, there can be extra phase factors due to edges lying over or under the plaquettes. Applying these operations in each plaquette in the sequence shown in Fig. 10(b) generates the WW wave function. Note that the plaquette terms in the WW model satisfy the same local constraint as that in the toric code model: the product of the six plaquette terms around a cube is identity. For toric code, this constraint is satisfied in the whole Hilbert space, while for a generic WW model, this constraint is only satisfied in the closed loop subspace. To see why this constraint holds, notice that the product of plaquette terms in the WW model in a closed surface measures the strings that go through the surface. For a topologically trivial closed surface like the surface of a cube, in the closed loop subspace, all strings going into the surface must come out. Therefore, the net flux going through a cube is zero in the closed loop subspace: the product of all the plaquette terms must be identity. Therefore, generating the plaquette terms in the bottom and side faces of a cube automatically gives rise to the plaquette term on the top face. The circuit depicted in Fig. 10 hence proceeds in the same way as for toric code. The gapped boundary created

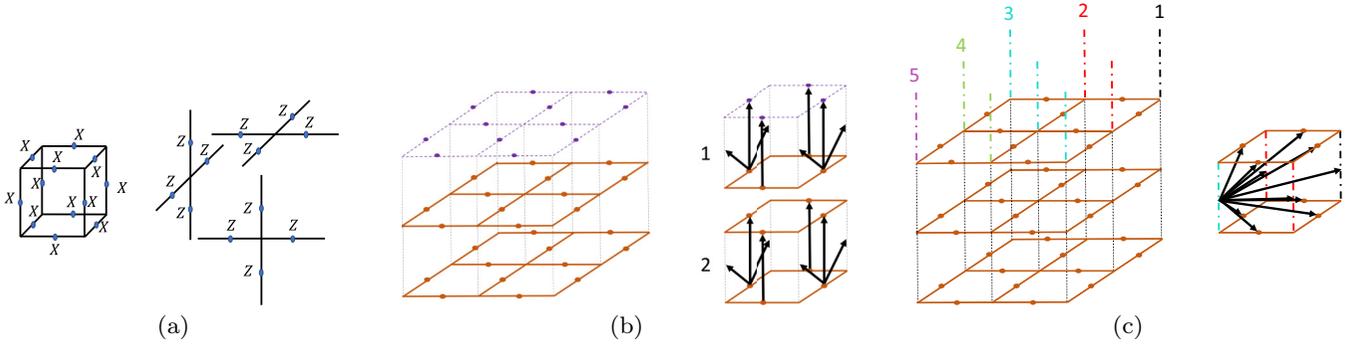


FIG. 11. (a) Pauli stabilizer terms of the X -cube model. The model is defined on a cubic lattice and qubits live on the edges. The X stabilizer term is associated with the cube and the Z stabilizer terms are associated with the vertices. (b) Sequential circuit for X -cube model (of height $L_z = 2$ cubes) with smooth boundaries at the top and bottom and periodic boundaries on the sides. Left: We start with $L_z = 2$ layers of toric code (shown in orange) with periodic boundary conditions in the layer directions and a layer of trivial qubits in state $|0\rangle$ (shown in purple). We add trivial qubits in state $|0\rangle$ on the vertical edges; the dots on the vertical edges are not shown for clarity. Right: On each vertical plaquette, we perform CNOT gates as shown using arrows that go from control to target qubits. We first do these gates as shown in parallel on all the top layer vertical plaquettes, then we do the layer below it, and so on. In the figure, we have $L_z = 2$, hence, our circuit has depth 2 as shown on the right. (c) Sequential circuit to convert X -cube model with smooth boundaries into an X -cube model on a torus. We add a layer of qubits on vertical edges in state $|+\rangle$ above the top smooth boundary (the top orange layer) such that we can join to the bottom smooth boundary (the bottom orange layer) via these vertical edges to impose periodic boundary conditions; the dots on the vertical edges are not shown for clarity. In one step of the sequential circuit, all qubits on the vertical edges in the same color are used as control qubits. The CNOT gates from each of these control qubits have target qubits on the edges of a single cube. An example is shown on the right. The CNOT gates are shown using arrows that go from control to target qubits.

with a circuit terminating at step $N + m$ has the corresponding $(2 + 1)$ D topological order of the WW model.

VI. MAP TO FRACTON PHASES

In this section, we construct sequential circuits for the preparation of the X -cube model, which is a well-known example of fracton topological order [32]. The X -cube model is a commuting projector model where the projectors are Pauli operators, as shown in Fig. 11(a). We use two different ways to construct the circuit: the first one is associated with the “foliation” structure of the X -cube model [33] while the second one is related to the “ p -string condensation” picture [34,35].

A. Foliation circuit for X -cube model

We now state the foliation circuit for the X -cube model. We write the circuit to prepare an X -cube model of height L_z cubes with smooth boundaries at the top and bottom and periodic boundary conditions on the sides; however, it can be generalized to other choices of boundary conditions. We start with L_z layers of toric code with periodic boundary conditions which were prepared using the sequential circuit shown in Sec. IV A and a layer of trivial qubits in state $|0\rangle$. We then combine the toric code layers and the layer of trivial qubits into an X -cube model with smooth boundaries using a sequential circuit as shown in Fig. 11(b). The CNOT gates are first applied in the top layer of vertical plaquettes, all in parallel, and then in the next layer, and so on. The overall circuit depth for preparing the X -cube model with smooth boundaries is linear. We apply a circuit of depth $O(L_x + L_y)$ for the toric code layers and then we need a sequential circuit of depth $O(L_z)$ to combine them into an X -cube model of height $O(L_z)$ cubes and with smooth boundaries [Fig. 11(b)].

The same circuit can be used to prepare the X -cube model with periodic boundary conditions in the vertical direction with fewer starting trivial qubits. To be specific, under periodic boundary conditions, the layer of trivial qubits shown in purple is not needed along the vertical direction and the orange layer at the bottom will connect to the top orange layer. The entangling gates, as shown in Fig. 11(b), will then prepare the X -cube model on a 3-torus.

However, if we first prepare the height L_z X -cube model with smooth boundaries and then convert it to the X -cube model on a 3-torus with height $L_z + 1$ by doing local gates at the top layer, we need a sequential circuit as shown in Fig. 11(c). We add a layer of trivial qubits on vertical edges above the top smooth boundary in state $|+\rangle$; these vertical edges now connect to the bottom smooth boundary due to periodic boundary conditions. Using these vertical qubits as control qubits, we do CNOT gates sequentially to prepare the layer of cube stabilizers. The CNOT gates that are applied in parallel are specified by the choice of the control qubits shown on the vertical edges with the same color and along the diagonal [left of Fig. 11(c)] and by the gates shown [right of Fig. 11(c)]. Thus, to convert the X -cube model with smooth boundaries into a 3-torus, we need a sequential circuit of depth $O(L_x + L_y)$ [Fig. 11(c)].

A generalization of the X -cube model is the Ising cage-net model, which is obtained from a coupled layer construction of double-Ising string-net models instead of toric code layers [35]. In Ref. [25], a generalized notion of foliation and a sequential circuit for preparation of the Ising cage-net model is presented. For the X -cube model, the resource toric code layers can be prepared in parallel before being inserted one by one into the bulk to increase the size of the X -cube model. For Ising cage net, however, this is not possible. Using the scheme presented in Ref. [25], a thin slab (of small L_z) of Ising

cage net can first be prepared with $O(L_x + L_y)$ steps. Then, the height of the slab L_z can be increased one at a time, each time requiring a circuit of depth $O(L_x + L_y)$. Therefore, overall a quadratic [$O(L^2)$] depth circuit is needed to generate the Ising cage-net model.

Whether one can do better than quadratic circuit depth for the Ising cage-net model is an open question. It will also be interesting to construct sequential circuit maps for more general fracton models such as the type-2 fracton models which have fractal-shaped logical operators [32].

B. p -string condensation circuit for X -cube model

We now construct a linear-depth sequential circuit for the preparation of the X -cube model using p -string condensation [34,35]. We first consider three decoupled stacks of toric codes in xy , yz , and zx planes such that a part of this stack $z > L$ has undergone p -string condensation. That is, $Z - Z$ stabilizers have been added to the double edges of the 3-foliated stack lattice for $z > L$, such that only the product of toric code X plaquette stabilizers around a cube which share the support of $Z - Z$ double-edge stabilizers survive as stabilizers. In the part $z \leq L$, we do not add these $Z - Z$ stabilizers and, hence, we still have toric code vertex and plaquette stabilizers. Thus, the part $z > L$ is like the X -cube model, the $z < L$ part is like the decoupled toric code stack, and there is a gapped boundary between the two at $z = L$. We assume periodic boundary conditions in the x and y directions, while the boundary conditions in the z direction do not play a role since we will only consider moving the $z = L$ interface by one lattice constant.

See Fig. 12(a) for this starting configuration of stabilizers. In the X -cube-like part, the stabilizers are the 24-body X stabilizer terms around a cube, $Z - Z$ stabilizers on every composite edge, and the original Z vertex stabilizers of the toric code. At the $z = L$ plane interface between the X -cube-like part and the decoupled toric code part, the X stabilizers supported on the cube are a 20-body X term on the cube above $z = L$ and the toric code X -plaquette term in the $z = L$ plane.

Our goal is to grow the X -cube portion of the model from $z > L$ to $z > L - 1$, i.e., to push the gapped boundary between the two models, similar to how our previously defined circuits push a gapped boundary to vacuum. In Fig. 12(b), we write the finite-depth circuit that acts on the layer of cubes right below $z = L$ to achieve this.

The $Z - Z$ stabilizers on the vertical bonds right below $z = L$ follow from the relation among the toric code Z vertex stabilizer terms around a “vertex” of the 3D stack lattice and the $Z - Z$ double-edge stabilizer terms supported on the same qubits as those vertex stabilizers. In this manner, the interface at $z = L$ has moved to $z = L - 1$ after the circuit is applied on the layer of cubes right below $z = L$.

Starting from toric code layers, the circuit depth of the p -string condensation circuit is linear, i.e., it scales as $O(L_z)$ since it takes a finite-depth circuit to grow the X -cube part by one unit length in L_z . Since the toric code layers in xy planes can be prepared in parallel in circuit depth $O(L_x + L_y)$, the overall circuit depth scales linearly in the system sizes as $O(L_x + L_y + L_z)$.

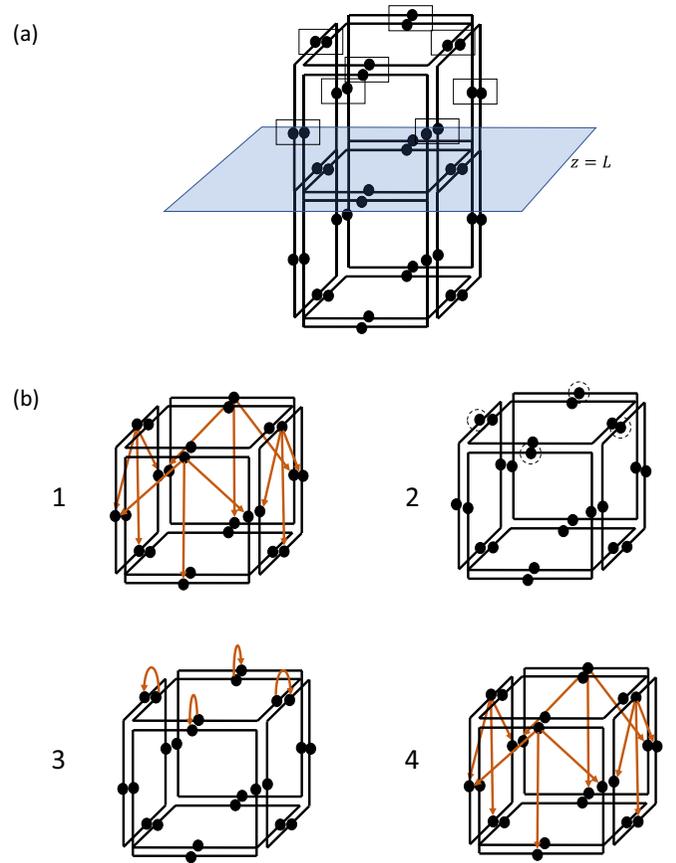


FIG. 12. (a) We start with the 3-foliated stack of 3D toric codes such that the resulting lattice has double edges with two qubits each. We add $Z - Z$ stabilizers on the double edges for $z > L$ such that only a product of toric code X -plaquette stabilizer terms that share the support of $Z - Z$ stabilizers around a cube survives as a stabilizer. In other words, we have performed p -string condensation in the region $z > L$. (b) The sequence of gates that acts on the cubes in the layer below $z = L$ to grow the X -cube-like part from $z > L$ to $z > L - 1$. The arrows denote the CNOT gates and the dashed circles denote Hadamard gates. We show the gates on a single cube here but the gates act across the entire layer of cubes.

VII. QUANTUM CELLULAR AUTOMATA

In this section, we show that arbitrary quantum cellular automata (QCA) can be realized as sequential circuits. A QCA is a unitary operator Q that preserves locality, meaning that, for any operator O_i supported on lattice site i , QO_iQ^\dagger is supported on sites that are at most a distance c away from i for some constant c [36]. We focus on implementing translationally invariant QCA on periodic boundary conditions. Any finite-depth quantum circuit (FDQC) gives an example of a QCA, but there are also QCA that cannot be realized as FDQC, such as the 1D shift operator S , which acts on operators on a 1D lattice as $SO_iS^\dagger = O_{i+1}$. However, it is easy to show that S can be realized as a sequential circuit of SWAP gates [37]. Now, we will show that any QCA in any spatial dimension can be realized as a sequential circuit. Additionally, if the QCA Q commutes with some global onsite symmetry $U_g = u_g^{\otimes N}$, the gates in the corresponding sequential circuits will also commute with $U(g)$. Therefore, symmetric QCA are a strict

subset of symmetric SQCs. On the other hand, SQC are not always QCAs as they do not always preserve locality.

The construction we use is closely related to the construction used in Ref. [38] to realize QCA as finite-depth circuits with geometrically nonlocal long-range gates. Let us first consider 1D QCA, and then later generalize to higher dimensions. The starting point is the standard form of a 1D QCA. After blocking sufficiently many sites, any QCA will have a unit range, meaning that QO_iQ^\dagger is supported only on sites $i-1, i, i+1$. Then, it was shown in Ref. [39] (see also Ref. [40]) that the QCA on a ring of N sites can be expressed in the following way:

$$Q^{(N)} = \left(\prod_{i=1}^{N/2} v_{2i-1,2i} \right) \left(\prod_{i=1}^{N/2} u_{2i,2i+1} \right), \quad (22)$$

where we take periodic boundaries. Therein, u is a unitary map between a $(d \times d)$ -dimensional Hilbert space to an $(\ell \times r)$ -dimensional Hilbert space where $\ell r = d^2$, and similarly v maps from and $(r \times \ell)$ -dimensional space to a $(d \times d)$ -dimensional space. We emphasize that these internal dimensions ℓ and r do not represent any physical degrees of freedom, nor do they appear in the sequential circuits we construct; they are simply a convenient technical tool. Indeed, since the input and output Hilbert spaces differ, u and v do not represent physical operations on their own, but they can be combined to define physical operations such as $Q^{(N)}$. As such, Eq. (22) is not an FDQC, so it can also represent QCA that are not FDQCs, such as the shift QCA.

To realize this QCA as a SQC, we closely follow the idea used for SPT states in Sec. III. The fundamental unitary gates in the SQC are the QCA on a ring of length 4,

$$Q_{i,j,k,l}^{(4)} = v_{i,j} v_{k,l} u_{j,k} u_{l,i}, \quad (23)$$

and also the gate

$$w_{i,j} = u_{j,i}^{-1} v_{i,j}^{-1}. \quad (24)$$

Note that, unlike u and v , the operators $Q^{(4)}$ and w are genuine physical unitary operators acting on four and two sites, respectively. Using these gates, we realize the QCA on a chain of even length $N > 4$ as an SQC in the following way:

$$Q^{(N)} \approx Q_{1,2,3,4}^{(4)} \prod_{i=1}^{\frac{N-4}{2}} \text{SWAP}_{2i+2,2i+4} w_{2i+1,2i+4} \times Q_{2i+1,2i+2,2i+3,2i+4}^{(4)}. \quad (25)$$

This equality follows from commuting all SWAP gates to one side of the equation, as is shown graphically in Fig. 13. Therein, the “ \approx ” sign means equivalence up to the SWAP gates which can easily be undone in a sequential manner.

Now we show how to apply the same construction to arbitrary spatial dimension D . We assume the topology of a D -dimensional hypercubic lattice with periodic boundary conditions. Once again, we can block sites such that Q spreads any operator by at most one site in all directions. Now, suppose we compactify all dimensions except one by viewing the D -dimensional system as a ring of supersites, each consisting of $(D-1)$ -dimensional tori. Then Q can be viewed as a 1D QCA acting on this compactified system. Therefore,

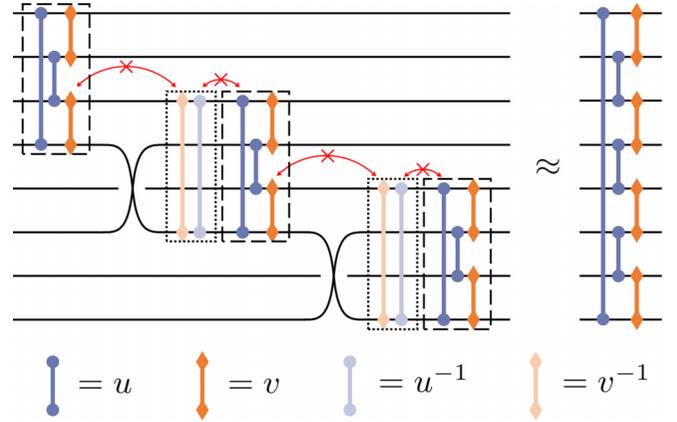


FIG. 13. Circuit diagram corresponding to Eq. (25) with $N = 8$. The sign “ \approx ” means equivalent up to a sequential circuit of SWAP gates. The gates $Q^{(4)}$ and w are shown in dashed and dotted boxes, respectively. Operators in the bulk of the circuit cancel pairwise, as indicated by the crosses. Note that this diagram does not depict the differences between the onsite physical dimension d and the internal dimensions ℓ, r of the standard form of the QCA.

we may use the above result to express it as a SQC in one direction; see Fig. 14(a). According to Eq. (25), the gates in this SQC are $Q^{(4)}$ and w and SWAP. The SWAP gate acts on the compactified systems by swapping corresponding sites between the two tori, so each SWAP of tori is local and can be done in depth 1. Since Q is locality preserving in all dimensions, $Q^{(4)}$ can be viewed as a $(D-1)$ -dimensional QCA. Likewise, w , which essentially acts as the QCA on two supersites, is also a $(D-1)$ -dimensional QCA, as was proven explicitly in Ref. [38]. Now, we use an inductive argument. We have already shown how to realize any 1D QCA as an SQC. Now, assume we can realize a $(D-1)$ -dimensional QCA as an SQC. According to the above discussion, any D -dimensional QCA Q can be written as an SQC consisting of $(D-1)$ -dimensional QCA. By the inductive assumption, each of these QCA is, in turn, an SQC, such that Q is itself an SQC. When constructed in this way, the depth of the SQC realizing Q grows like $O(N)$ where N is the number of sites. This inductive procedure is illustrated in Fig. 14 for $D = 2$.

It turns out that the SQCs constructed above are also composed of symmetric gates, so they are symmetric SQCs. That is, suppose that $Q^{(N)}$ commutes with some global onsite unitary symmetry $u^{\otimes N}$ for all N . It is clear that the SWAP gates commute with the global symmetry, as does $Q^{(4)}$ by

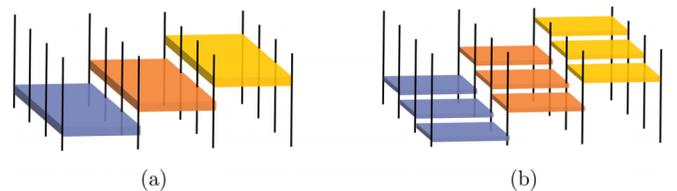


FIG. 14. To construct a D -dimensional QCA as a sequential circuit, we decompose it in terms of $(D-1)$ -dimensional QCA acting sequentially in one direction (a), then decompose each $(D-1)$ -dimensional QCA into $(D-2)$ -dimensional QCA (b), and so on.

assumption. It was shown in Ref. [38] that w also commutes with the symmetry. Therefore, each of the local gates in the SQC [Eq. (25)] is symmetric, so it is a symmetric SQC. One application of this comes from applying the construction to the FDQCs, which create SPT ground states from symmetric product states [14]. These FDQCs commute with the global symmetry protecting the SPT as a whole, but the individual gates do not commute with the symmetry. Applying our construction gives an SQC consisting of symmetric gates that realize the same unitary. This gives an alternative, but similar, construction of SQCs for SPT states compared to those given in Sec. III.

It is interesting to consider truncating the SQCs we have constructed. Consider the case of $D = 3$, where an important class of QCA is given by those which disentangle ground states of Walker-Wang models [41–43]. It has been argued that these QCA are nontrivial, meaning that they cannot be written as a product of FDQCs and translations. The argument is rooted in the fact that if the QCA acts as an FDQC followed by a translation, then it can be easily truncated to act only in a finite region of space. These truncated circuits could then be used to generate an isolated boundary of the Walker-Wang models that host surface topological order, which is conjectured to be impossible in some cases. Thus, the nontrivial nature of the QCA is tied to the inability to truncate its action to a finite region of space straightforwardly. Since we have constructed SQCs realizing QCA, and these SQCs can be truncated, one may worry that there is a contradiction. However, this is not the case, as truncating the SQC implements the QCA exactly on a smaller periodic system without introducing any boundaries. Therefore, the SQCs cannot be used to isolate boundaries of the Walker-Wang models, and the nontriviality of the QCA, as described above, is compatible with the existence of a SQC realizing the QCA. The sequential circuit shown in Sec. IV B for constructing the Walker-Wang wave function is not mapped from a QCA and does leave open boundaries if truncated. But as it always generates two surfaces (top and bottom) at the same time, there is no contradiction with the nontriviality of the QCA either.

VIII. SUMMARY AND OUTLOOK

In this paper, we discussed the generation of nontrivial gapped quantum states, starting from product states, using sequential quantum circuits. In particular, we discussed how the circuit that generates the (symmetrized) symmetry-breaking state and the SPT states preserves global symmetry but not necessarily locality; how the circuit that generates fractional topological states is associated with gapped boundaries to the vacuum with either charge or loop condensation; and how the circuit that generates fracton states is related to the foliation structure or p -string condensation in the fracton states.

We can compare four types of many-body unitaries: the finite-depth quantum circuit (FDQC), the quantum cellular automata (QCA), the sequential quantum circuit (SQC), and the linear depth quantum circuit (LDQC). Table I summarizes the similarities and differences between them. Based on what we know, we see that they have a strict containment relation

$$\text{FDQC} \subset \text{QCA} \subset \text{SQC} \subset \text{LDQC}. \quad (26)$$

TABLE I. Comparison of four types of many-body unitary transformations: finite-depth local unitary circuit (FDQC), quantum cellular automaton (QCA), sequential quantum circuit (SQC), and linear depth quantum circuit (LDQC). “Yes” means all the unitaries in this class preserve certain properties. “No” means some unitaries in this class do not preserve the property. Asterisk: we consider Walker-Wang models with a modular input category as short-range entangled even though they are generated with a nontrivial QCA.

Preserving	FDQC	QCA	SQC	LDQC
Finite correlation length	Yes	Yes	No	No
Zero correlation length	Yes	Yes	No	No
Short-range entanglement	Yes	Yes*	No	No
Locality	Yes	Yes	No	No
Ground-state degeneracy	Yes	Yes	No	No
Entanglement area law	Yes	Yes [44]	Yes	No

Table I summarizes the similarities and differences between the three sets in terms of their effect on global properties of gapped many-body states. Linear depth circuit is the most powerful which can map gapped states to generic entanglement volume-law states. The other three types of unitary all preserve entanglement area law, hence mapping gapped states to gapped states. SQC can change short-range correlation into long-range correlation, short-range entanglement into long-range entanglement. It can also change the locality of their parent Hamiltonian and its ground-state degeneracy, hence capable of mapping between different gapped phases. They can also generate a nonzero finite correlation length from a state with zero correlation length.³ FDQC and QCA, on the other hand, preserve the locality of operator, correlation function, and entanglement and cannot change ground-state degeneracy of gapped Hamiltonian.

The multiscale entanglement renormalization procedure (MERA) [45–47] is another way to map nontrivial entangled states to product states. It takes a many-body entangled wave function, applies one layer of finite-depth circuit which disentangles local degrees of freedom (DOF), and maps the wave function back to its original form but with a doubled unit cell. Such a step is then repeated at the renormalized length scale until the wave function is completely disentangled after $\log L$ steps. It differs from the sequential circuit in that in later steps of the procedure, the degrees of freedom are very far away from each other, and the unitary gates applied are not local anymore. This nonlocality in unitary gates is also why the MERA procedure does not violate the linear lower bound in circuit depth to generate GHZ or topological states [2]. MERA is more powerful than a sequential circuit as it can map product states to gapless states [48].

Throughout the paper, we focused on gapped phases with a commuting projector Hamiltonian and a zero correlation length ground-state wave function and an important class of gapped phases left out of the discussion are the chiral phases.

³For example, SQCs can generate arbitrary matrix product states on open boundary conditions, which can have finite, nonzero correlation length [50].

In an upcoming work [49], we show that chiral phases can be generated by incorporating sequential adiabatic evolution, an analog version of sequential quantum circuit, where local Hamiltonian terms in subregions are adiabatically evolved in a sequential manner while the whole system remains gapped throughout the process.

ACKNOWLEDGMENTS

We are indebted to inspiring discussions with W. Ji, L. Lootens, B. Vancraeynest-De Cuiper, X.-G. Wen, C. Xu, C. Zhang, and M. Levin. X.C. and A.D. are supported by the National Science Foundation under Award No. DMR-1654340, the Simons Investigator Award (Award ID No. 828078), and the Institute for Quantum Information and Matter at Caltech. X.C. and N.T. are supported by the Walter Burke Institute for Theoretical Physics at Caltech. R.V. is supported by the Belgian-American Educational Foundation. The work of M.H. on fracton systems was supported by the U.S. Department of Energy, Office of Science, Basic Energy Sciences (BES) under Award No. DE-SC0014415. The work of D.T.S., the work of M.H. on aspects other than fractons, and (in part) the work of X.C. and A.D. was supported by the Simons Collaboration on Ultra-Quantum Matter, which is funded by grants from the Simons Foundation (Grant No. 651440, D.T.S. and M.H.; Grant No. 651438, X.C. and A.D.).

APPENDIX A: MAP TO SYMMETRY-BREAKING PHASES OF GENERAL FINITE GROUPS

In this Appendix, we show how the construction in Sec. II can be generalized from the Z_2 group to arbitrary finite groups. We start by generalizing the Hilbert space from the Z_2 case $\{|0\rangle, |1\rangle\}$ to a finite group G $\{|z\rangle : z \in G\}$ and replacing the X and $(1 \pm Z)/2$ operators with

$$\begin{aligned} X^g |z\rangle &= |gz\rangle, & g, z \in G \\ T^h |z\rangle &= \delta_{h,z} |z\rangle, & h, z \in G \end{aligned} \quad (\text{A1})$$

where $(X^g)^\dagger = X^{g^{-1}}$ is unitary and T^h is a projection onto the state $|h\rangle$. These operators satisfy the commutation relation $X^g T^h = T^{gh} X^g$. Using these new operators, we can rewrite the Z_2 -invariant transverse field Ising model (1) as

$$H = -J \sum_i \sum_{h \in G} T_i^h T_{i+1}^h - B \frac{1}{|G|} \sum_i \sum_{g \in G} X_i^g, \quad (\text{A2})$$

which is now invariant under a global symmetry G , i.e., $(\prod_i X_i^g) H (\prod_i X_i^g)^\dagger = H$.⁴

⁴We remark that the Hamiltonian (A2) actually has a larger $S_{|G|}$ symmetry. If one wishes, the redundant symmetry can be explicitly broken by modifying the transverse field. For example,

$$H = -J \sum_i \sum_{h \in G} T_i^h T_{i+1}^h - B \frac{1}{|G|} \sum_i \sum_{g \in G} C_g \overleftarrow{X}_i^g,$$

where C_g are some random coefficients with the constraint that $C_g = C_{g^{-1}}$ and $\overleftarrow{X}^g |h\rangle = |hg^{-1}\rangle$ denotes right multiplication.

Similar to the transverse field Ising model, the Hamiltonian above has a symmetric phase ($B \gg J > 0$) and a symmetry-breaking phase ($J \gg B > 0$). The corresponding fixed-point wave functions are given by

$$\begin{aligned} |\psi_{\text{SY}}\rangle &= |++++\dots\rangle, \\ |\psi_{\text{SB}}\rangle &= \frac{1}{\sqrt{|G|}} \sum_g |gg\dots g\rangle, \end{aligned} \quad (\text{A3})$$

where $|+\rangle = \frac{1}{\sqrt{|G|}} \sum_g |g\rangle$. To construct a quantum circuit that maps $|\psi_{\text{SY}}\rangle$ to $|\psi_{\text{SB}}\rangle$, we define

$$R_{ij}(\mathcal{W}) \equiv \sum_{h \in G} T_i^h X_j^h \mathcal{W}_j X_j^{h^{-1}}, \quad (\text{A4})$$

where \mathcal{W} is a unitary operator so that $R_{ij}(\mathcal{W})$ is unitary in each subspace projected by T_i^h and, consequently, unitary in the full Hilbert space. One can also check that $R_{ij}(\mathcal{W})$ is invariant under the symmetry operation

$$\begin{aligned} \left(\prod_k X_k^g \right) R_{ij}(\mathcal{W}) \left(\prod_k X_k^g \right)^\dagger &= \sum_{h \in G} T_i^{gh} X_j^{gh} \mathcal{W}_j X_j^{(gh)^{-1}} \\ &= R_{ij}(\mathcal{W}). \end{aligned} \quad (\text{A5})$$

Any circuit made up of $R_{ij}(\mathcal{W})$ gates is hence symmetric. In analogy to Eqs. (7) and (8), consider a circuit of the form

$$U = R_{N1}(S) \prod_{i=N}^1 U_{i,i+1}, \quad (\text{A6})$$

where

$$U_{i,i+1} = R_{i,i+1}(S^\dagger \text{Had} S^\dagger) R_{i,i+1}(S). \quad (\text{A7})$$

The matrices $R_{i,i+1}(S^\dagger \text{Had} S^\dagger)$ and $R_{i,i+1}(S)$ in (A7) are chosen such that they will reduce to $R(X_{i+1})$ and $R(Z_i, Z_{i+1})$ in (8) for the $G = Z_2$ case, respectively. To map $|\psi_{\text{SY}}\rangle$ to $|\psi_{\text{SB}}\rangle$, we need to find unitaries Had and S that map

$$\text{Had}|+\rangle = |e\rangle, \quad S^\dagger |z\rangle = e^{-i\theta(z)} |z\rangle, \quad (\text{A8})$$

such that

$$\begin{aligned} U_{i,i+1}|+, +\rangle &= \frac{1}{\sqrt{|G|}} \sum_h X_{i+1}^h S_{i+1}^\dagger \text{Had}_{i+1} |h, +\rangle \\ &= \frac{e^{-i\theta(e)}}{\sqrt{|G|}} \sum_h X_{i+1}^h |h, e\rangle \\ &= \frac{e^{-i\theta(e)}}{\sqrt{|G|}} \sum_h |h, h\rangle. \end{aligned} \quad (\text{A9})$$

According to the mapping (A8), the first row of Had is restricted to be $\frac{1}{\sqrt{|G|}}(1, \dots, 1)$, and S must be diagonal. In addition to these two constraints, the Had and S matrices can be selected freely as long as Had is unitary and S is diagonal. A particular choice of Had can be taken as

$$\text{Had} = \sum_{p, \mu, \nu} \sum_z \sqrt{\frac{\rho_p}{|G|}} A_{\mu\nu}^p(z) |p, \mu, \nu\rangle \langle z|. \quad (\text{A10})$$

Here p labels inequivalent irreducible representations of the group G , μ and ν are the row and column indices of the

representing matrix A^p , and ρ_p is the dimension of the representation space. ‘‘Had’’ is a mapping from group elements to entries in the irreducible representation matrices and in particular it should map the identity element to the trivial representation of G in order to ensure the mapping (A8).

We can also check the mappings of operators under the circuit (A6) and discuss whether these mappings hold for a general group G . First, the symmetry operator remains invariant under the circuit

$$\prod_{i=1}^N X_i^g \rightarrow \prod_{i=1}^N X_i^g \quad (\text{A11})$$

as our unitary circuit converses the symmetry. And the transverse magnetic field term is always mapped to the ferromagnetic term:

$$\frac{1}{|G|} \sum_{g \in G} X_i^g \rightarrow \sum_h T_{i-1}^h T_i^h, \quad i = 2, \dots, N \quad (\text{A12})$$

which can be proved by the following steps:

$$\begin{aligned} U_{i-1,i} \left(\frac{1}{|G|} \sum_{g \in G} X_i^g \right) U_{i-1,i}^\dagger &= \sum_h T_{i-1}^h X_i^h S_i^\dagger \text{Had}_i |+\rangle \langle +| \text{Had}_i^\dagger S_i X_i^{h^{-1}} \\ &= \sum_h T_{i-1}^h T_i^h \end{aligned} \quad (\text{A13})$$

and

$$U_{i,i+1} \left(\sum_h T_{i-1}^h T_i^h \right) U_{i,i+1}^\dagger = \sum_h T_{i-1}^h T_i^h. \quad (\text{A14})$$

Unfortunately, the other mappings in the Z_2 case in (10) do not hold for a general finite group G . For instance, the ferromagnetic term $\sum_h T_i^h T_{i+1}^h$ generally does not map to the transverse field term $\frac{1}{|G|} \sum_{g \in G} X_{i+1}^g$.

Things become easier in the case where G is Abelian, as the Had matrix in (A10) reduces to conventional Fourier transformations of the Abelian group G . Without loss of generality, we consider the example of a cyclic group Z_M , where we can choose

$$\begin{aligned} \text{Had} &= \sum_{\alpha, z=0}^{M-1} \frac{1}{\sqrt{M}} \exp\left(\frac{2\pi i}{M} \alpha z\right) |\alpha\rangle \langle z|, \\ S &= \sum_{z=0}^{M-1} \exp\left(\frac{\pi i}{M} z^2\right) |z\rangle \langle z|. \end{aligned} \quad (\text{A15})$$

In the simplest case where $G = Z_2$, the Had and S matrices will reduce to the conventional Hadamard gate and phase gate, respectively. It can also be verified that $R_{ij}(S) = \exp[-i\frac{\pi}{4}(Z_i Z_j - 1)]$ and $R_{ij}(S^\dagger \text{Had} S^\dagger) = \exp(-i\frac{\pi}{4} X_j)$, which implies that the generalized circuit (A6) will reduce to (7) in the Z_2 case up to a global phase.

For the Abelian groups, all the operator mappings for the Z_2 case are preserved as long as we choose the gate as in

(A15). The mappings in (10) will become

$$\begin{aligned} \frac{1}{|G|} \sum_g X_i^g &\rightarrow \sum_h T_{i-1}^h T_i^h, \quad i = 2, \dots, N \\ \frac{1}{|G|} \sum_g X_1^g &\rightarrow \sum_g \frac{1}{|G|} \sum_{h,h'} e^{-\frac{2\pi i}{|G|}(h'-h)g} T_1^h T_N^{h'} \prod_{i=1}^N X_i^g, \\ \sum_h T_i^h T_{i+1}^h &\rightarrow \frac{1}{|G|} \sum_g X_i^g, \quad i = 2, \dots, N \\ \sum_h T_1^h T_2^h &\rightarrow \frac{1}{|G|} \sum_g X_1^g \prod_{i=1}^N X_i^g. \end{aligned} \quad (\text{A16})$$

In the bulk of the system, the transverse field term maps to the Ising term and vice versa. Near the left end point, the correspondence is preserved only under the symmetric subspace $\prod_{i=1}^N X_i^g = 1$.

Generalization to higher-dimensional versions shows no difference with the Z_2 case, as shown in Fig. 3.

APPENDIX B: MAP TO STRING-NET STATES WITH GENERAL GAPPED BOUNDARIES AND THE TORUS

This Appendix contains some of the necessary notions from category theory to define the general sequential circuit in Sec. III B.

Following Kitaev and Kong, gapped boundaries of string-net models are classified by \mathcal{C} -module categories \mathcal{M} , where \mathcal{C} is the (unitary) fusion category of the string-net model. The boundary can be interpreted as a gapped domain wall between the string net \mathcal{C} and the vacuum, given by the category of vector spaces Vec , in which case \mathcal{M} can be viewed as a $(\mathcal{C}, \text{Vec})$ -bimodule category. At the same time, given \mathcal{M} and \mathcal{C} , we can always find the (unique) Morita dual category of \mathcal{C} , $\mathcal{D} = \mathcal{C}_{\mathcal{M}}^*$, in which case \mathcal{M} is an invertible $(\mathcal{C}, \mathcal{D})$ -bimodule category. This is the structure we will require to define our quantum circuit. We refer the reader to Refs. [26,27] for more details about the underlying mathematics and will present only some key ingredients here in order to define the sequential circuit. Given the resemblance of the operators needed to define our sequential circuit and the (finite-depth) circuit from Ref. [24], we follow the same conventions here.

The Hilbert space of string-net models consists of configurations on a hexagonal lattice, with the edges labeled by simple objects a, b, \dots of a unitary fusion category \mathcal{C} . The simple objects obey fusion constraints: $a \times b = \sum_c N_{ab}^c c$, which are assumed to be multiplicity free ($N_{ab}^c \in \{0, 1\}$) from now on. There exists a unit object $\mathbf{1}$, such that $\forall a, \mathbf{1} \times a = a$. Each simple object has a quantum dimension d_a associated to it. We can diagrammatically write the resolution of the identity

$$\begin{array}{c} a \\ | \\ a \end{array} \begin{array}{c} b \\ | \\ b \end{array} = \sum_c \sqrt{\frac{d_c}{d_a d_b}} \begin{array}{c} a \quad b \\ \diagdown \quad / \\ c \\ / \quad \diagdown \\ a \quad b \end{array}. \quad (\text{B1})$$

The fusion category has an associator or ${}^{\mathcal{C}}F$ -symbol: we use the \mathcal{C} superscript to distinguish the symbol from other associators that will be defined below, expressed in terms of the simple objects as

$$\begin{array}{c} b \\ \swarrow \\ e \\ \swarrow \searrow \\ a \quad d \end{array} \quad \begin{array}{c} c \\ \swarrow \\ e \\ \swarrow \searrow \\ a \quad d \end{array} = \sum_f ({}^{\mathcal{C}}F_d^{abc})_e^f \begin{array}{c} b \\ \swarrow \\ f \\ \swarrow \searrow \\ a \quad d \end{array} \quad \begin{array}{c} c \\ \swarrow \\ f \\ \swarrow \searrow \\ a \quad d \end{array}. \quad (\text{B2})$$

This ${}^{\mathcal{C}}F$ -symbol is simply the usual F -symbol for fusion categories and obeys the well-known pentagon equation. We also require the ‘‘bubble pop’’ identity:

$$\begin{array}{c} c' \\ \uparrow \\ \text{bubble} \\ \downarrow \\ c \end{array} = \sqrt{\frac{d_a d_b}{d_c}} \delta_{cc'} \begin{array}{c} c \\ \uparrow \\ \downarrow \\ c \end{array}. \quad (\text{B3})$$

A (right) \mathcal{C} -module category \mathcal{M} has simple objects that will be denoted by capital Roman letters (A, B, C, \dots) and a right-action of \mathcal{C} on \mathcal{M} , $\triangleleft: \mathcal{M} \times \mathcal{C} \rightarrow \mathcal{M}$. The action is strict: $A \triangleleft \mathbf{1} = A$ and generally we can write $A \triangleleft a = \sum_{B \in \mathcal{M}} N_{Aa}^B B$. Note that these new N_{Aa}^B are distinct from the original fusion rules of \mathcal{C} (N_{ab}^c) and may carry multiplicities ($N_{Aa}^B > 1$) even if the N_{ab}^c do not. \mathcal{M} has an associator ${}^{\triangleleft}F$ that implements the following recoupling at the boundary of the string net:

$$\begin{array}{c} A \\ \uparrow \\ j \\ \swarrow \searrow \\ C \quad a \\ \downarrow \\ B \end{array} \quad \begin{array}{c} b \\ \swarrow \\ j \\ \swarrow \searrow \\ k \end{array} = \sum_{c,m} ({}^{\triangleleft}F_B^{Aab})_{C,jk}^{c,m} \begin{array}{c} A \\ \uparrow \\ m \\ \swarrow \searrow \\ C \quad a \\ \downarrow \\ B \end{array} \quad \begin{array}{c} b \\ \swarrow \\ m \\ \swarrow \searrow \\ k \end{array}, \quad (\text{B4})$$

where the boundary is now blue, labeled by objects in the module category \mathcal{M} . A suitable gauge choice can be made such that

$$({}^{\triangleleft}F_B^{A1b})_{b,1k}^{B,m} = ({}^{\triangleleft}F_B^{Aa1})_{a,k1}^{A,m} = \delta_k^m. \quad (\text{B5})$$

The inverse symbol is denoted by ${}^{\triangleleft}F$.

We now have a second unitary fusion category \mathcal{D} with simple objects labeled by greek letters ($\alpha, \beta, \delta, \dots$). In our case, this category will be chosen to be the unique Morita dual of \mathcal{C} , given $\mathcal{M}: \mathcal{D} = \mathcal{C}_{\mathcal{M}}^*$. The F -move for \mathcal{D} is defined in the following way:

$$\begin{array}{c} \beta \\ \swarrow \\ \mu \\ \swarrow \searrow \\ \alpha \quad \delta \end{array} \quad \begin{array}{c} \gamma \\ \swarrow \\ \mu \\ \swarrow \searrow \\ k \end{array} = \sum_{\nu, m, n} ({}^{\mathcal{D}}F_{\delta}^{\alpha\beta\gamma})_{\mu, jk}^{\nu, mn} \begin{array}{c} \beta \\ \swarrow \\ \nu \\ \swarrow \searrow \\ \alpha \quad \delta \end{array} \quad \begin{array}{c} \gamma \\ \swarrow \\ \nu \\ \swarrow \searrow \\ n \end{array}. \quad (\text{B6})$$

Just like for \mathcal{M} , it is not guaranteed that \mathcal{D} is multiplicity free, even if \mathcal{C} is.

We will now take \mathcal{M} to be a left \mathcal{D} -module category, with an associator ${}^{\triangleright}F$. The bulk string net \mathcal{D} is now on the left of

the boundary:

$$\begin{array}{c} \alpha \\ \swarrow \\ j \\ \swarrow \searrow \\ \gamma \quad k \end{array} \quad \begin{array}{c} \beta \\ \swarrow \\ j \\ \swarrow \searrow \\ k \end{array} \quad \begin{array}{c} A \\ \uparrow \\ j \\ \swarrow \searrow \\ B \end{array} = \sum_{C, mn} ({}^{\triangleright}F_B^{\alpha\beta A})_{\gamma, jk}^{C, mn} \begin{array}{c} \alpha \\ \swarrow \\ m \\ \swarrow \searrow \\ n \end{array} \quad \begin{array}{c} \beta \\ \swarrow \\ m \\ \swarrow \searrow \\ n \end{array} \quad \begin{array}{c} A \\ \uparrow \\ m \\ \swarrow \searrow \\ B \end{array}. \quad (\text{B7})$$

A similar gauge choice can be made such that

$$({}^{\triangleright}F_B^{1\beta A})_{\beta, 1k}^{B, m1} = ({}^{\triangleright}F_B^{\alpha 1 A})_{a, 1k}^{A, 1m} = \delta_k^m. \quad (\text{B8})$$

The inverse symbol is denoted by ${}^{\triangleright}F$. For \mathcal{M} to be a $(\mathcal{D}, \mathcal{C})$ -bimodule category we require one last additional associator ${}^{\triangleright\triangleleft}F$:

$$\begin{array}{c} \alpha \\ \swarrow \\ j \\ \swarrow \searrow \\ C \quad k \end{array} \quad \begin{array}{c} A \\ \uparrow \\ j \\ \swarrow \searrow \\ B \end{array} = \sum_{D, mn} ({}^{\triangleright\triangleleft}F_B^{\alpha Ab})_{C, jk}^{D, mn} \begin{array}{c} \alpha \\ \swarrow \\ m \\ \swarrow \searrow \\ n \end{array} \quad \begin{array}{c} A \\ \uparrow \\ m \\ \swarrow \searrow \\ B \end{array}. \quad (\text{B9})$$

with a similar gauge choice leading to

$$({}^{\triangleright\triangleleft}F_B^{1Ab})_{A, 1k}^{b, m1} = ({}^{\triangleright\triangleleft}F_B^{\alpha A 1})_{B, k1}^{A, 1m} = \delta_k^m. \quad (\text{B10})$$

The inverse symbol is denoted by ${}^{\triangleright\triangleleft}F$. There are in total six coupled pentagon equations that the set of F -symbols should satisfy. These are given in Ref. [27].

One final ingredient we need is a resolution of the identity that allows for fusion of two \mathcal{M} lines in terms of a \mathcal{D} and a \mathcal{C} line. Although \mathcal{M} has no intrinsic duality, we can define the ‘‘opposite’’ bimodule category $\overline{\mathcal{M}}$, a $(\mathcal{C}, \mathcal{D})$ -bimodule category, which contains the dual objects of \mathcal{M} . The rigorous mathematical picture behind this is called a Morita context [22,51]. This allows us to define

$$\begin{array}{c} A \\ \downarrow \\ A \end{array} \quad \begin{array}{c} B \\ \downarrow \\ B \end{array} = \sum_{c, n} \sqrt{\frac{d_c}{d_A d_B}} \begin{array}{c} A \\ \swarrow \\ n \\ \downarrow \\ c \\ \uparrow \\ n \\ \swarrow \searrow \\ A \quad B \end{array}, \quad (\text{B11})$$

and the opposite resolution in terms of a \mathcal{D} line:

$$\begin{array}{c} A \\ \downarrow \\ A \end{array} \quad \begin{array}{c} B \\ \downarrow \\ B \end{array} = \sum_{\gamma, n} \sqrt{\frac{d_\gamma}{d_A d_B}} \begin{array}{c} A \\ \swarrow \\ n \\ \downarrow \\ \gamma \\ \uparrow \\ n \\ \swarrow \searrow \\ A \quad B \end{array}, \quad (\text{B12})$$

where d_A, d_B are quantum dimensions for the objects in \mathcal{M} . Note that the invertibility of the bimodule \mathcal{M} guarantees (B11) and (B12).

We are now ready to write the precise action of the general sequential circuit in Figs. 7 (cylinder case) and 8 (torus case) in terms of the individual plaquette actions for the seven

sublattices L_1, \dots, L_7 . The original smooth boundary case in Fig. 6 is recovered by choosing $\mathcal{M} = \mathcal{C}$. In this case, the Morita dual is simply \mathcal{C} itself ($\mathcal{D} = \mathcal{C}$) and all F -symbols defined above reduce to ${}^{\mathcal{C}}F$ (up to some suitable permutations of the labels). For simplicity, we will only treat the general case. We will follow the orientation convention as

in Ref. [24]. The control edges in the sequential circuit are colored red in Eqs. (B13)–(B19). In the full sequential circuit, the control edge has not been acted on by previous operations and is still labeled by the trivial object in \mathcal{D} . For the purpose of generality, we assume it to be arbitrary in Eqs. (B13)–(B18):

$$[1B_p^{S, \mathcal{M}}]_{\{\epsilon_i, k_i\}}^{\{\tilde{E}_i, h_i\}} = \sum_{\{n_i\}} \sqrt{\frac{d_{\epsilon_6} d_{\epsilon_5} d_{\epsilon_3} d_{\epsilon_2} d_{\tilde{E}_4} d_{\tilde{E}_1}}{d_{\alpha_5} d_{\alpha_2} d_{\epsilon_4} d_{\epsilon_1} d_S^2}}$$

$$\begin{pmatrix} \triangleright F_{\tilde{E}_1}^{\alpha_1 \epsilon_2 S} \end{pmatrix}_{\epsilon_1, k_1 n_1}^{\tilde{E}_2, j_2 h_1} \begin{pmatrix} \triangleright F_{\tilde{E}_2}^{\epsilon_2 \epsilon_3 \tilde{E}_3} \end{pmatrix}_{\alpha_2, k_2 h_2}^{S, n_3 n_2} \begin{pmatrix} \triangleright F_S^{\epsilon_3 \alpha_3 \tilde{E}_4} \end{pmatrix}_{\epsilon_4, k_3 n_4}^{\tilde{E}_3, h_3 n_3}$$

$$\begin{pmatrix} \triangleright F_S^{\epsilon_5 \alpha_4 \tilde{E}_4} \end{pmatrix}_{\epsilon_4, k_4 n_4}^{\tilde{E}_5, h_4 n_5} \begin{pmatrix} \triangleright F_{\tilde{E}_6}^{\epsilon_6 \epsilon_5 \tilde{E}_5} \end{pmatrix}_{\alpha_5, k_5 h_5}^{S, n_6 n_5} \begin{pmatrix} \triangleright F_{\tilde{E}_1}^{\alpha_6 \epsilon_6 S} \end{pmatrix}_{\epsilon_1, k_6 n_1}^{\tilde{E}_6, n_6 h_6}.$$

The case for the original plaquette operator B_p^S [Eq. (15)] is recovered by choosing $\mathcal{M} = \mathcal{C} = \mathcal{D}$ in (B13). The matrix $[B_p^S]$ we recover in this case is different than the one in the original Levin-Wen string-net paper because of the different orientation convention and the fact that no tetrahedral symmetry of the ${}^{\mathcal{C}}F$ -symbols is assumed [52]:

$$[2B_p^{S, \mathcal{M}}]_{\{\tilde{e}_i, \tilde{E}_i, h_i\}}^{\{\epsilon_i, E_i, k_i\}} = \sum_{\{\tilde{e}_i, \tilde{E}_i, h_i\}} \sqrt{\frac{d_{\epsilon_2} d_{\epsilon_3} d_{\epsilon_4} d_{\tilde{E}_1} d_{\tilde{E}_3} d_{\tilde{e}_1} d_{\tilde{e}_2}}{d_{\epsilon_1} d_{A_2} d_{A_3} d_{\tilde{E}_2} d_{\tilde{E}_4} d_S^2}}$$

$$\begin{pmatrix} \triangleright F_{\tilde{E}_1}^{\alpha_1 \epsilon_2 S} \end{pmatrix}_{\epsilon_1, k_1 n_1}^{\tilde{E}_2, n_2 h_1} \begin{pmatrix} \triangleright F_{A_1}^{\epsilon_2 S \tilde{e}_1} \end{pmatrix}_{\tilde{E}_2, n_2 h_2}^{E_1, n_3 k_2} \begin{pmatrix} \triangleright F_{E_1}^{\epsilon_3 \tilde{E}_3 \tilde{e}_1} \end{pmatrix}_{S, n_4 n_3}^{A_2, h_3 k_3}$$

$$\begin{pmatrix} \triangleright F_{E_2}^{\epsilon_3 \tilde{E}_3 \tilde{e}_2} \end{pmatrix}_{S, n_4 n_5}^{A_3, h_4 k_4} \begin{pmatrix} \triangleright F_{A_4}^{\epsilon_4 S \tilde{e}_2} \end{pmatrix}_{\tilde{E}_4, n_6 h_5}^{E_2, n_5 k_5} \begin{pmatrix} \triangleright F_{\tilde{E}_1}^{\alpha_2 \epsilon_4 S} \end{pmatrix}_{\epsilon_1, k_6 n_1}^{\tilde{E}_4, n_6 h_6}.$$

$$A_1 = \sum_{\{\tilde{e}_i, \tilde{E}_i, h_i\}} [{}^3B_p^{S, \mathcal{M}}]_{\{\epsilon_i, E_i, k_i\}}$$

$$[{}^3B_p^{S, \mathcal{M}}]_{\{\epsilon_i, E_i, k_i\}} = \sum_{\{n_i\}} \sqrt{\frac{d_{\epsilon_2} d_{\epsilon_3} d_{A_2} d_{\tilde{E}_1} d_{\tilde{e}_1} d_{\tilde{e}_3}}{d_{\epsilon_1} d_{E_3} d_{\alpha_2} d_{\tilde{E}_2} d_{\tilde{E}_3} d_{\tilde{e}_2}}}$$

$$\begin{pmatrix} \langle \triangleright F_{\tilde{E}_1}^{\alpha_1 \epsilon_2 S} \rangle_{\epsilon_1, k_1 n_1}^{\tilde{E}_2, n_2 h_1} & (\times F_{A_1}^{\epsilon_2 S \tilde{e}_1})_{\tilde{E}_2, n_2 h_2}^{E_1, n_3 k_2} & (\langle \triangleleft F_{E_2}^{S \tilde{e}_1 a_2} \rangle)_{E_1, n_3 k_3}^{\tilde{e}_2, n_4} \\ (\langle \triangleleft F_{\tilde{E}_2}^{S \tilde{e}_3 a_3} \rangle)_{E_3, n_5 k_4}^{\tilde{e}_2, n_4} & (\times F_{A_2}^{\epsilon_3 S \tilde{e}_3})_{\tilde{E}_3, n_6 h_3}^{E_3, n_5 k_5} & (\langle \triangleright F_{E_1}^{\alpha_4 \epsilon_3 S} \rangle)_{\epsilon_1, k_6 n_1}^{\tilde{E}_3, n_6 h_4} \end{pmatrix}.$$

$$A_2 = \sum_{\{\tilde{e}_i, \tilde{E}_i, h_i\}} [{}^4B_p^{S, \mathcal{M}}]_{\{\epsilon_i, E_i, k_i\}}$$

$$[{}^4B_p^{S, \mathcal{M}}]_{\{\epsilon_i, E_i, k_i\}} = \sum_{\{n_i\}} \sqrt{\frac{d_{\epsilon_1} d_{E_3} d_{E_1} d_{\tilde{e}_1} d_{\tilde{e}_2} d_{\tilde{e}_4} d_{\tilde{E}_2}}{d_{A_1} d_{E_4} d_{A_4} d_{\tilde{E}_2} d_{\tilde{e}_3} d_S^2}}$$

$$\begin{pmatrix} (\times F_{\tilde{E}_1}^{\epsilon_1 E_1 \tilde{e}_1})_{A_1, k_1 h_1}^{S, n_1 n_2} & (\times F_{A_2}^{\epsilon_1 S \tilde{e}_2})_{\tilde{E}_1, n_2 h_2}^{E_2, n_3 k_2} & (\langle \triangleleft F_{E_3}^{S \tilde{e}_2 a_1} \rangle)_{E_2, n_3 k_3}^{\tilde{e}_3, n_4} \\ (\langle \triangleleft F_{\tilde{E}_3}^{S \tilde{e}_4 a_2} \rangle)_{E_4, n_5 k_4}^{\tilde{e}_3, n_4} & (\times F_{A_3}^{\epsilon_2 S \tilde{e}_4})_{\tilde{E}_2, n_6 h_3}^{E_4, n_5 k_5} & (\times F_{\tilde{E}_2}^{\epsilon_2 E_1 \tilde{e}_1})_{A_4, k_6 h_4}^{S, n_1 n_6} \end{pmatrix}.$$

$$A_1 = \sum_{\{\tilde{e}_i, \tilde{E}_i, h_i\}} [{}^5B_p^{S, \mathcal{M}}]_{\{\epsilon_i, E_i, k_i\}}$$

$$[{}^5B_p^{S, \mathcal{M}}]_{\{\epsilon_i, E_i, k_i\}} = \sum_{\{n_i\}} \sqrt{\frac{d_{\epsilon_1} d_{E_1} d_{E_3} d_{\tilde{e}_2} d_{\tilde{e}_4} d_{\tilde{e}_5}}{d_{A_2} d_{E_2} d_{a_3} d_{\tilde{e}_1} d_S^2}}$$

$$\begin{pmatrix} (\langle \triangleleft F_S^{E_1 a_1 \tilde{e}_2} \rangle)_{E_2, k_1 n_2}^{\tilde{e}_1, n_1} & (\times F_S^{\epsilon_1 A_1 \tilde{e}_2})_{E_2, k_2 n_2}^{E_1, h_1 n_3} & (\times F_{E_3}^{\epsilon_1 \tilde{E}_1 \tilde{e}_3})_{S, n_3 n_4}^{A_2, h_2 k_3} \\ (\langle \triangleleft F_{E_3}^{S \tilde{e}_4 a_2} \rangle)_{E_4, n_5 k_4}^{\tilde{e}_3, n_4} & (\langle \triangleleft F_{E_4}^{E_5 \tilde{e}_5 \tilde{e}_4} \rangle)_{S, n_6 n_5}^{a_3, k_5} & (\langle \triangleleft F_S^{E_1 a_4 \tilde{e}_5} \rangle)_{E_5, k_6 n_6}^{\tilde{e}_1, n_1} \end{pmatrix}.$$

$$A_2 = \sum_{\{\tilde{\epsilon}_i, \tilde{E}_i, h_i\}} [{}^6B_p^{S, \mathcal{M}}]_{\{\epsilon_i, E_i, k_i\}} \quad (B18)$$

$$[{}^6B_p^{S, \mathcal{M}}]_{\{\epsilon_i, E_i, k_i\}}^{\{\tilde{\epsilon}_i, \tilde{E}_i, h_i\}} = \sum_{\{n_i\}} \sqrt{\frac{d_{\epsilon_1} d_{E_3} d_{E_1} d_{\tilde{\epsilon}_2} d_{\tilde{\epsilon}_4} d_{\tilde{\epsilon}_5}}{d_{A_1} d_{E_2} d_{a_3} d_{\tilde{\epsilon}_3} d_S^2}}$$

$$\begin{pmatrix} \langle \boxtimes F_{\tilde{E}_1}^{\epsilon_1 E_1 \tilde{\epsilon}_1} \rangle_{A_1, k_1 h_2}^{S, n_1 n_2} & \langle \boxtimes F_{A_2}^{\epsilon_1 S \tilde{\epsilon}_2} \rangle_{\tilde{E}_1, n_2 h_2}^{E_2, n_3 k_2} & \langle \langle F_{E_3}^S \tilde{\epsilon}_2 a_1 \rangle \rangle_{E_2, n_3 k_3}^{\tilde{\epsilon}_3, n_4} \\ \langle \langle F_{E_3}^S \tilde{\epsilon}_4 a_2 \rangle \rangle_{E_4, n_5 k_4}^{\tilde{\epsilon}_3, n_4} & \langle \langle F_{E_4}^S \tilde{\epsilon}_5 \tilde{\epsilon}_4 \rangle \rangle_{S, n_6 n_5}^{a_3, k_5} & \langle \langle F_S^E a_4 \tilde{\epsilon}_5 \rangle \rangle_{E_5, k_6 n_5}^{\tilde{\epsilon}_1, n_1} \end{pmatrix}.$$

$$a_2 = \sum_{\{\tilde{\epsilon}_i\}} [{}^7B_p^{S, \mathcal{M}}]_{\{E_i, k_i\}} \quad (B19)$$

$$[{}^7B_p^{S, \mathcal{M}}]_{\{E_i, k_i\}}^{\{\tilde{\epsilon}_i\}} = \sum_{\{n_i\}} \sqrt{\frac{d_{E_1} d_{E_4} d_{\tilde{\epsilon}_2} d_{\tilde{\epsilon}_3} d_{\tilde{\epsilon}_5} d_{\tilde{\epsilon}_6}}{d_{a_2} d_{a_5} d_{\tilde{\epsilon}_1} d_{\tilde{\epsilon}_4} d_S^2}}$$

$$\begin{pmatrix} \langle \langle F_S^{E_1 a_1 \tilde{\epsilon}_2} \rangle \rangle_{E_2, k_1 n_2}^{\tilde{\epsilon}_1, n_1} & \langle \langle F_{E_3}^{E_2 \tilde{\epsilon}_2 \tilde{\epsilon}_3} \rangle \rangle_{S, n_2 n_3}^{a_2, k_2} & \langle \langle F_{E_4}^S \tilde{\epsilon}_3 a_3 \rangle \rangle_{E_3, n_3 k_3}^{\tilde{\epsilon}_4, n_4} \\ \langle \langle F_{E_4}^S \tilde{\epsilon}_5 a_4 \rangle \rangle_{E_5, n_5 k_4}^{\tilde{\epsilon}_4, n_4} & \langle \langle F_{E_5}^{E_6 \tilde{\epsilon}_6 \tilde{\epsilon}_5} \rangle \rangle_{S, n_6 n_5}^{a_5, k_5} & \langle \langle F_S^{E_1 a_6 \tilde{\epsilon}_6} \rangle \rangle_{E_6, k_6 n_6}^{\tilde{\epsilon}_1, n_1} \end{pmatrix}.$$

Finally, we show the action of growing an S -loop on the minimal torus with trivial edges:

$$= \sum_{\alpha, \beta, \gamma \{h_i\}} \sqrt{\frac{d_\alpha d_\gamma}{d_\beta d_S^2}} \left(\langle \boxtimes F_S^{\alpha \gamma S} \rangle \right)_{\beta, h_5 h_2}^{S, h_3 h_1} \left(\langle \boxtimes F_S^{\gamma \alpha S} \rangle \right)_{\beta, h_4 h_2}^{S, h_1 h_3} \quad (B20)$$

This action is the first step in the sequential circuit to construct the string-net ground state on the torus (Fig. 8). It guarantees that the state is already in the ground state before the

last plaquette operator (B19) is performed. Therefore, (B19) is unitary, even though no edge is left to serve as a valid control.

APPENDIX C: MAP TO SPTS IN TWO DIMENSIONS AND HIGHER

The sequential circuit in Sec. III for generating the CZX state (the 2D \mathbb{Z}_2 SPT) from the trivial state can be generalized to group cohomology SPT states with any symmetry and in any dimension [14]. Following the notation in Sec. III, consider fixed-point wave functions as shown in Fig. 4(c).

There are four spin DOFs in each site with basis states $|g\rangle$, $g \in G$. The symmetry acts on each site as

$$U_g |g_1, g_2, g_3, g_4\rangle \rightarrow \frac{\alpha(g_2, g_1)\alpha(g_1, g_4)}{\alpha(g_3, g_4)\alpha(g_2, g_3)} |gg_1, gg_2, gg_3, gg_4\rangle, \quad (C1)$$

where $\alpha(g_i, g_j) = v_3(g_i, g_j, g^{-1}g^*, g^*)$. Four spins connected by a black square are in a local entangled state $\frac{1}{\sqrt{|G|}} \sum_g |gggg\rangle$. When the entangled states are each contained within a lattice site, the state is a trivial SPT. When the entangled states are between lattice sites, the state is a nontrivial SPT. See Fig. 4(c).

The circuit that maps between the two proceeds with a sequence of SWAP gates, as shown in Fig. 4(d). The sequence of SWAP gates maps between the two wave functions. However, after each SWAP gate, the symmetry action changes and needs to be restored. Following the notation in Fig. 4(c), after the step of SWAP, spins 1 and 5 are exchanged and spins 4 and 8 are exchanged. Therefore, the phase difference that needs to be corrected to restore the symmetry action is

$$\frac{\alpha(g_2, g_1)\alpha(g_1, g_4)\alpha(g_6, g_5)\alpha(g_5, g_8)\alpha(g_3, g_8)\alpha(g_2, g_3)\alpha(g_7, g_4)\alpha(g_6, g_7)}{\alpha(g_3, g_4)\alpha(g_2, g_3)\alpha(g_7, g_8)\alpha(g_6, g_7)\alpha(g_2, g_5)\alpha(g_5, g_8)\alpha(g_6, g_1)\alpha(g_1, g_4)} \quad (C2)$$

which can be separated into two parts for the upper and lower halves of the squares

$$\frac{\alpha(g_2, g_1)\alpha(g_6, g_5)}{\alpha(g_2, g_5)\alpha(g_6, g_1)}, \quad \frac{\alpha(g_3, g_8)\alpha(g_7, g_4)}{\alpha(g_3, g_4)\alpha(g_7, g_8)}. \quad (C3)$$

The two parts can be corrected separately with phase factors acting on 1,2,5,6 and 4,3,8,7, respectively. We will focus on the upper part. Using cocycle calculation, we see that

$$\begin{aligned} \frac{\alpha(g_2, g_1)\alpha(g_6, g_5)}{\alpha(g_2, g_5)\alpha(g_6, g_1)} &= \frac{v_3(g_2, g_1, g^{-1}g^*, g^*)v_3(g_6, g_5, g^{-1}g^*, g^*)}{v_3(g_2, g_5, g^{-1}g^*, g^*)v_3(g_6, g_1, g^{-1}g^*, g^*)} \\ &= \frac{v_3(g_2, g_1, g^{-1}g^*, g^*)v_3(g_6, g_2, g^{-1}g^*, g^*)}{v_3(g_6, g_1, g^{-1}g^*, g^*)} \frac{v_3(g_6, g_5, g^{-1}g^*, g^*)}{v_3(g_2, g_5, g^{-1}g^*, g^*)v_3(g_6, g_2, g^{-1}g^*, g^*)} \\ &= \frac{v_3(g_6, g_2, g_1, g^*)}{v_3(g_6, g_2, g_1, g^{-1}g^*)} \frac{v_3(g_6, g_2, g_5, g^{-1}g^*)}{v_3(g_6, g_2, g_5, g^*)} \\ &= \frac{v_3(g_6, g_2, g_1, g^*)}{v_3(gg_6, gg_2, gg_1, g^*)} \frac{v_3(gg_6, gg_2, gg_5, g^*)}{v_3(g_6, g_2, g_5, g^*)} \end{aligned} \quad (C4)$$

which can be achieved by conjugating the symmetry operation by a phase factor of $\frac{v_3(g_6, g_2, g_1, g^*)}{v_3(g_6, g_2, g_5, g^*)}$.

Similar calculations can be done for the lower part of the squares and vertical swaps. Therefore, the SWAP gate can be dressed by phase factors to make sure that symmetry operators remain invariant throughout the transformation. Moreover, the phase correction coming from the upper half and the lower half cancel each other when acting on the ground-state wave function; therefore, the added phases do not change the ground state, so the ground state transforms as we wanted under the SWAP gates.

A similar construction applies to group cohomology SPT states in higher dimensions.

-
- [1] X. Chen, Z.-C. Gu, and X.-G. Wen, *Phys. Rev. B* **82**, 155138 (2010).
- [2] S. Bravyi, M. B. Hastings, and F. Verstraete, *Phys. Rev. Lett.* **97**, 050401 (2006).
- [3] C. Schön, E. Solano, F. Verstraete, J. I. Cirac, and M. M. Wolf, *Phys. Rev. Lett.* **95**, 110503 (2005).
- [4] C. Schön, K. Hammerer, M. M. Wolf, J. I. Cirac, and E. Solano, *Phys. Rev. A* **75**, 032311 (2007).
- [5] M. C. Bañuls, D. Pérez-García, M. M. Wolf, F. Verstraete, and J. I. Cirac, *Phys. Rev. A* **77**, 052306 (2008).
- [6] Z.-Y. Wei, D. Malz, and J. I. Cirac, *Phys. Rev. Lett.* **128**, 010607 (2022).
- [7] S.-H. Lin, R. Dilip, A. G. Green, A. Smith, and F. Pollmann, *PRX Quantum* **2**, 010342 (2021).
- [8] L. Lamata, J. León, D. Pérez-García, D. Salgado, and E. Solano, *Phys. Rev. Lett.* **101**, 180506 (2008).
- [9] H. Saberi, *Phys. Rev. A* **84**, 032323 (2011).
- [10] Y.-J. Liu, K. Shtengel, A. Smith, and F. Pollmann, *PRX Quantum* **3**, 040315 (2022).
- [11] Y. Huang and X. Chen, *Phys. Rev. B* **91**, 195143 (2015).
- [12] W. W. Ho and T. H. Hsieh, *SciPost Phys.* **6**, 029 (2019).
- [13] P. Chen, B. Yan, and S. X. Cui, *arXiv:2210.01682*.
- [14] X. Chen, Z.-C. Gu, Z.-X. Liu, and X.-G. Wen, *Phys. Rev. B* **87**, 155114 (2013).
- [15] X. Chen, Z.-C. Gu, and X.-G. Wen, *Phys. Rev. B* **83**, 035107 (2011).
- [16] N. Schuch, D. Pérez-García, and I. Cirac, *Phys. Rev. B* **84**, 165139 (2011).
- [17] F. Pollmann, A. M. Turner, E. Berg, and M. Oshikawa, *Phys. Rev. B* **81**, 064439 (2010).
- [18] X. Chen, Z.-X. Liu, and X.-G. Wen, *Phys. Rev. B* **84**, 235141 (2011).
- [19] L. Fidkowski, J. Haah, and M. B. Hastings, *Phys. Rev. B* **101**, 155124 (2020).
- [20] M. A. Levin and X.-G. Wen, *Phys. Rev. B* **71**, 045110 (2005).
- [21] A. Kitaev, *Ann. Phys.* **303**, 2 (2003).
- [22] A. Kitaev and L. Kong, *Commun. Math. Phys.* **313**, 351 (2012).

- [23] Y. Zhang, T. Grover, A. Turner, M. Oshikawa, and A. Vishwanath, *Phys. Rev. B* **85**, 235151 (2012).
- [24] L. Lootens, B. Vancraeynest-De Cuiper, N. Schuch, and F. Verstraete, *Phys. Rev. B* **105**, 085130 (2022).
- [25] Z. Wang, X. Ma, D. T. Stephen, M. Hermele, and X. Chen, *Phys. Rev. B* **108**, 035148 (2023).
- [26] P. Etingof, S. Gelaki, D. Nikshych, and V. Ostrik, *Tensor Categories* (American Mathematical Society, Providence, RI, 2016), Vol. 205.
- [27] L. Lootens, J. Fuchs, J. Haegeman, C. Schweigert, and F. Verstraete, *SciPost Phys.* **10**, 053 (2021).
- [28] O. Buerschaper, M. Aguado, and G. Vidal, *Phys. Rev. B* **79**, 085119 (2009).
- [29] Z.-C. Gu, M. Levin, B. Swingle, and X.-G. Wen, *Phys. Rev. B* **79**, 085118 (2009).
- [30] A. Hamma, P. Zanardi, and X.-G. Wen, *Phys. Rev. B* **72**, 035307 (2005).
- [31] K. Walker and Z. Wang, *Front. Phys.* **7**, 150 (2012).
- [32] S. Vijay, J. Haah, and L. Fu, *Phys. Rev. B* **94**, 235157 (2016).
- [33] W. Shirley, K. Slagle, Z. Wang, and X. Chen, *Phys. Rev. X* **8**, 031051 (2018).
- [34] H. Ma, E. Lake, X. Chen, and M. Hermele, *Phys. Rev. B* **95**, 245126 (2017).
- [35] A. Prem, S.-J. Huang, H. Song, and M. Hermele, *Phys. Rev. X* **9**, 021010 (2019).
- [36] T. Farrelly, *Quantum* **4**, 368 (2020).
- [37] H. C. Po, L. Fidkowski, T. Morimoto, A. C. Potter, and A. Vishwanath, *Phys. Rev. X* **6**, 041070 (2016).
- [38] D. T. Stephen, A. Dua, A. Lavasani, and R. Nandkishore, *PRX Quantum* **5**, 010304 (2024).
- [39] B. Schumacher and R. F. Werner, [arXiv:quant-ph/0405174](https://arxiv.org/abs/quant-ph/0405174).
- [40] J. I. Cirac, D. Perez-Garcia, N. Schuch, and F. Verstraete, *J. Stat. Mech.* (2017) 083105.
- [41] J. Haah, L. Fidkowski, and M. B. Hastings, *Commun. Math. Phys.* **398**, 469 (2023).
- [42] J. Haah, *J. Math. Phys.* **62**, 092202 (2021).
- [43] W. Shirley, Y.-A. Chen, A. Dua, T. D. Ellison, N. Tantivasadakarn, and D. J. Williamson, *PRX Quantum* **3**, 030326 (2022).
- [44] L. Piroli and J. I. Cirac, *Phys. Rev. Lett.* **125**, 190402 (2020).
- [45] G. Vidal, *Phys. Rev. Lett.* **101**, 110501 (2008).
- [46] Z.-C. Gu and X.-G. Wen, *Phys. Rev. B* **80**, 155131 (2009).
- [47] B. Swingle and J. McGreevy, *Phys. Rev. B* **93**, 045127 (2016).
- [48] G. Evenbly and S. R. White, *Phys. Rev. Lett.* **116**, 140403 (2016).
- [49] X. Chen, M. Hermele, and D. T. Stephen, [arXiv:2402.03433](https://arxiv.org/abs/2402.03433).
- [50] D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, *Quantum Inf. Comput.* **7**, 401 (2007).
- [51] G. Schaumann, *J. Algebra* **379**, 382 (2013).
- [52] A. Hahn and R. Wolf, *Phys. Rev. B* **102**, 115154 (2020).