

Planar Fault-Tolerant Circuits for Non-Clifford Gates on the 2D Color Code

Andreas Bauer^{1,*} and Julio C. Magdalena de la Fuente^{2,†}

¹*Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*

²*Dahlem Center for Complex Quantum Systems, Freie Universität Berlin, Berlin 14195, Germany*



(Received 30 September 2025; revised 23 February 2026; accepted 24 April 2026; published 23 June 2026)

We introduce a family of scalable planar fault-tolerant circuits that implement logical non-Clifford operations on a 2D color code, such as a logical T gate or a logical non-Pauli measurement that prepares a magic $|T\rangle$ state. The circuits are relatively simple, consisting only of physical T gates, CX gates, and few-qubit measurements. They can be implemented with an array of qubits on a 2D chip with nearest-neighbor couplings and no wire crossings. The construction is based on a spacetime path integral representation of a non-Abelian 2+1D topological phase, which is related to the 3D color code. We turn the path integral into a circuit by expressing it as a spacetime ZX tensor network and then traversing it in some chosen time direction. We describe in detail how fault tolerance is achieved using a “just-in-time” decoding strategy, for which we repurpose and extend state-of-the-art color-code matching decoders.

DOI: [10.1103/prxquantum.7.020367](https://doi.org/10.1103/prxquantum.7.020367)

I. INTRODUCTION

Topological quantum error correction is one of the most promising routes toward fault-tolerant quantum computation due to its intrinsic scalability with local, low-weight measurements. The most prominent examples of 2D topological codes are the surface code [1–3] and the color code [4], small instances of which have been realized experimentally [5–7]. One of their most attractive features is their ability to perform logical Clifford gates transversally and via lattice surgery using a 2D planar qubit connectivity [8–13]. Logical non-Clifford gates are more difficult: One of the most-studied protocols is *magic state distillation* [14–16], which comes at the cost of significant additional overhead.

Recently, Ref. [17] introduced a new approach to implement a variety of logical non-Clifford gates in a scalable, topological, fault-tolerant, purely 2D manner, using domain walls between (Abelian) stabilizer codes and non-Abelian codes. Historically, this approach emerged through 3 + 0D dimension-jump protocols [18,19] using transversal non-Clifford gates of 3D codes [20,21], which can be turned into 2+1D protocols using *just-in-time*

decoding [22–24]. From this perspective, the non-Abelian phases interpretation leads to a much greater flexibility in constructing both new global protocols for different logic gates, as well as new microscopic circuits. Despite its promising features, there are still significant steps required to bring this proposal closer to a practical implementation.

A. Contributions

In this work, we make further progress in the design of low-overhead and practical protocols for logical non-Clifford gates: We construct a family of fault-tolerant circuits, called *twisted color circuits*, as microscopic implementation for the global logical non-Clifford gates proposed in Ref. [17]. The circuits are composed of simple physical operations such as CX and T gates, and few-qubit Pauli- X or Z measurements, and the initial and final state is encoded in a 2D color code. Importantly, the required qubit connectivity is fully planar, so the protocols are implementable on a 2D chip with nearest-neighbor couplings and without wire crossings. This makes our protocols particularly suitable for specific hardware architectures such as superconducting qubits [5–7].

To derive twisted color circuits, we use the *path-integral approach* to topological quantum error correction. In this approach, a +1-postselected syndrome-extraction circuit is viewed as a spacetime ZX tensor network [25,26] or a path-integral representation of the topological phase [27–29]. Vice versa, a path integral can be turned into a circuit by traversing it in some time direction, and the +1-postselection can be removed by introducing defects. The

*Contact author: andib@mit.edu

†Contact author: jm@juliomagdalena.de

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

path-integral approach is well suited to construct fault-tolerant circuits for exotic topological phases, including non-Abelian ones, as was demonstrated in Ref. [28]. The particular ZX tensor-network path integral from which we derive twisted color circuits is called the *color path integral*, and it is inspired from the 3D color code and its transversal T gate. There is a variety of different twisted color circuits, depending on a choice of (1) a 3-colex on which the underlying 3D color code is defined, (2) a time direction, (3) a way to place *qubit worldlines*, and (4) a global spacetime topology that determines the overall logical action (see Ref. [17]).

The path-integral approach was already used in Refs. [17,28] to construct low-overhead non-Abelian circuits, which are related to the transversal CCZ gate on three copies of the 3D toric code rather than the transversal T gate in the 3D color code. Compared to these circuits, the two advantages of twisted color circuits are (1) that they use single-qubit physical T gates instead of 3-qubit CCZ gates, and (2) that their qubit connectivity is naturally planar. Note that Refs. [17,22] also give protocols based on physical T gates, but explicit circuits are not given and would have a larger qubit footprint and not be planar.

We also provide a direct physical interpretation of the color path integral as a lattice topological quantum field theory, or a discrete gauge theory, for a non-Abelian topological phase. We formulate the underlying cohomology theory, called *color cohomology* (see also Ref. [30]), establish the gauge invariance of the according action, show its equivalence to a known Dijkgraaf-Witten gauge theory, and study its flux and charge defects. The latter are essential for the decoding of twisted color circuits. The detailed interplay of defects that we derive can also be used to accurately and efficiently benchmark the logical performance of the protocol, despite the non-Clifford gates in the circuit. As also noted in Ref. [17], it is not necessary to simulate the full circuit including the non-Clifford operations.

Like the protocols in Refs. [17,22–24], Pauli- X errors in our twisted color circuits have to be decoded and corrected *just-in-time* during the execution of the circuit, instead of globally at the end of the protocol. We describe in detail how this decoding works microscopically, concretely implementing ideas of Ref. [22]. Our method of just-in-time decoding invokes an arbitrary 3D-color-code decoder twice at every time step: Once to obtain a fresh global “current best estimate,” and another time to join the previously committed corrections with the current best estimate. This is to be contrasted with more heuristic RG-based decoders which are mainly discussed in Refs. [17,23,24]. Our method is closer to the original ideas in Ref. [22] and the matching-based algorithm mentioned in Ref. [17]. We revisit the restriction decoder [30], which can be used as a matching-based global decoder for the Z errors in our circuits, and propose a global decoding algorithm for the X errors. Furthermore, we study how the knowledge of the

Pauli- X decoding can be used later to optimize the global decoding of the Pauli- Z errors.

B. Structure

We give a complete self-contained pedagogical example of a twisted color circuit implementing a specific logical gate in Sec. II, which we hope to be accessible to a broad error-correction audience. We provide deeper mathematical insights into the construction in the following Secs. III and IV, where we introduce color cohomology and study the color path integral as a spacetime representation of a non-Abelian topological phase. We give more details on decoding and fault tolerance of twisted color circuits using a just-in-time matching decoder in Sec. V. We provide further examples for twisted color circuits in Sec. VI. An error-correction inclined reader might be particularly interested in Secs. II, V, and VI, while a reader interested in the mathematical background and the relation to non-Abelian topological fixed-point path integrals might want to focus on Secs. III and IV.

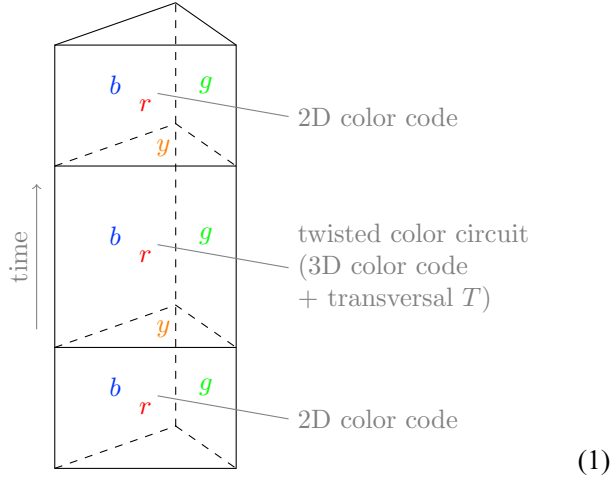
II. EXAMPLE: $|T\rangle$ -STATE MEASUREMENT ON HEXAGONAL GRID

In this section, we provide a self-contained exposition of our construction, by focusing on one particular example of a twisted color circuit. Some basic familiarity with (1) the three-dimensional color code [4,31] and (2) the use of ZX diagrams to represent and manipulate error-correcting protocols [25–27,29,32–34] is helpful but not necessary. We do not aim to provide the “best” example in terms of expected overhead and logical error rate, but instead the one that may be the most convenient for the reader to follow. In Sec. VI we provide further examples of twisted color circuits. We remark that depending on the hardware requirements further improvements of the local structure of the circuits can be made.

A. The global protocol

Twisted color circuits are geometrically local circuits of unitaries and measurements in $2 + 1D$. Moreover, they have a macroscopic fault distance which makes them *topological*. To create a protocol that does not only store but also process logical information, we define the circuits in a spacetime that is composed of different subregions. This includes 3D *bulk* regions that are interfaced along 2D *boundary* and *domain-wall* regions, that respectively are interfaced along one-dimensional *corner* regions. Note that all of these subregions and interfaces have to be defined carefully in order to give a topologically protected protocol. Within each subregion, the circuit is uniform, that is, it is fully specified by a spacetime unit cell. The *global protocol* is specified by a spacetime diagram where we give a

label to each region that describes the unit cell of the circuit. Our example uses a global protocol from Ref. [17] that performs a logical $|T\rangle$ -state measurement,



That is, the circuit that we construct will act on qubits distributed over a triangle. We start by performing the $2 + 1$ D syndrome-extraction circuit for the 2D color code, on the usual triangle with r (red), g (green), and b (blue) color boundaries. Through some constant-depth transitioning circuit, we switch to the twisted color circuit, related to 3D color code and its transversal T gate. After applying the twisted color circuit for some time that needs to be increased linearly with the desired fault distance of the protocol, we switch back to the 2D color code via another constant-depth transitioning circuit. We label the interface between the 2D and 3D color-code regions with y (yellow) since it is related to the y color boundary of the 3D color code. The spatial boundaries of the twisted color circuit are related to the r , g , and b color boundaries of the 3D color code. Overall (including error correction), the circuit performs a logical $|T\rangle$ -state measurement of the qubit encoded in the 2D color code [17]. This is a measurement with eigenstates $|T\rangle := |0\rangle + e^{2\pi i(1/8)} |1\rangle$ and $|0\rangle + e^{2\pi i(5/8)} |1\rangle$, which can be used to prepare the $|T\rangle$ magic state.

1. Dimensional jumps

We believe that it is most natural to describe our protocols directly in terms of a $2 + 1$ D non-Abelian topological phase as we do in Sec. IV. However, for this section, we motivate them from a measurement-based perspective, as in Refs. [22,23]. To this end, we reinterpret the space-time diagram above, viewing the middle part as a $3 + 0$ D measurement-based protocol. When we reach the temporal y domain wall after performing some 2D-color-code syndrome-extraction circuit, we perform a dimension jump [18]: We prepare the 3D color code state in the middle triangle prism with the according r , g , b , and y color boundaries, by preparing qubits in the $|+\rangle$ state and measuring all Z -type stabilizers. In doing so, we inject the 2D

color-code logic state at the y boundary at the bottom. Then we perform the transversal T gate in the 3D volume, and collapse the logic state of the 3D color code onto a 2D color code at its top y boundary. Finally, we continue with the 2D color-code syndrome-extraction circuit.

2. $3 + 0 = 2 + 1$

Note that there is nothing $3 + 1$ D about the above protocol since the 3D color code in the middle only exists for constant time. We can turn a $3 + 0$ D measurement-based protocol [35,36] into a $2 + 1$ D circuit protocol, by keeping a few 2D slices of qubits “active” at a time, and sweeping through the 3D code by initializing a new slice on one side and measuring out the oldest slice on the other side. As was first noticed in Ref. [22], this can also be done with the $3 + 0$ D color-code circuit including the transversal T gate. As we need to fix the Pauli frame before applying each physical T gate, this requires continuous decoding during the execution of the circuit before the full syndrome history is available, a task called *just-in-time decoding* in Ref. [22]. In this work, we provide a method to obtain such $2 + 1$ D circuits without keeping multiple 2D slices active. As a consequence, the twisted color circuits we obtain from our method need a smaller number of qubits, and most importantly, allow for a fully planar connectivity. In order to derive these circuits, we first turn the $3 + 0$ D protocol into a 3D ZX tensor-network path integral in spacetime, which we refer to as the *color path integral*. Then we turn this ZX tensor network into a circuit, using methods similar to those in Refs. [25–29].

B. The color path integral

In this subsection, we define the $3 + 0$ D color-code measurement-based protocol mentioned in the previous section more precisely, and then turn it into what we call the *color path integral*, expressed as a ZX tensor network. We first focus on the bulk of the protocol and later define appropriate boundary conditions.

1. The 3D color code

The 3D color code [31] can be defined on any *3-colex*, which is a 3D cellulation that is Poincaré dual to a triangulation [37], and whose volumes are colored red (r), green (g), blue (b), and yellow (y) such that adjacent volumes always have different colors.

In every 3-colex, we denote

- (a) the set of i -cells by Δ_i ,
- (b) the pair of colors (such as rg) of the volumes adjacent to a face $f \in \Delta_2$ by $\mathbf{c}_f$,
- (c) the triple of colors of the volumes adjacent to an edge $e \in \Delta_1$ by $\mathbf{c}_e$, where we denote the triple that does not contain a color c by $\bar{c}$ (such as $rgb \simeq \bar{y}$),

- (d) the *sign* of a vertex $p \in \Delta_0$ by $\text{sgn}_p \in \{\pm 1\}$. We have $\text{sgn}_p = +1$ if for the r , g , and b volumes around the $rgb = \bar{y}$ -colored edge adjacent to p , the clockwise ordering when looking away from p coincides with the ordering $r - g - b$, and $\text{sgn}_p = -1$ otherwise. In particular, the vertices at the endpoints of an edge always have opposite sign.

For our example twisted color circuit, we take the 3-colex shown in Fig. 1 consisting of hexagon-prism volumes, which is combinatorially equivalent to a *bitruncated cubic honeycomb* [38]. The 3D color code is defined on one qubit for every vertex of the 3-colex. There is one stabilizer for every face f , acting as a product of Z at all the vertices p of f ,

$$S_f^Z = \prod_{p \in f} Z_p, \quad (2)$$

and one stabilizer for every volume v , acting as a product of X at all vertices of v ,

$$S_v^X = \prod_{p \in v} X_p. \quad (3)$$

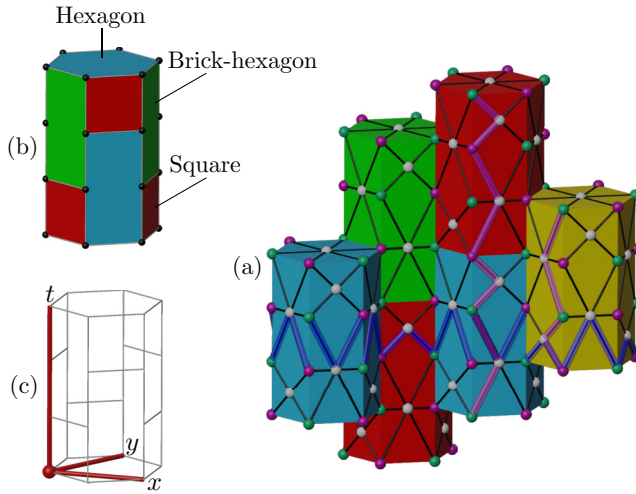


FIG. 1. (a) Snippet of the 3-colex and associated color path integral used to define the example twisted color circuit. The volumes of the 3-colex are hexagon prisms colored accordingly. The X -tensors of the ZX diagram are white balls. The $+\pi/4$ Z -tensors are purple balls, and the $-\pi/4$ Z -tensors are dark green balls. Bonds are represented as black wires. The two purple shaded paths of bonds are examples of qubit worldlines—there are two such worldlines for each t -perpendicular edge. The bonds shaded in dark blue are an example for a qubit timeslice. (b) Yellow volume of the 3-colex, whose boundary faces have been colored like the adjacent volume. The boundary itself forms a 2-colex. There are three types of faces in the 3-colex, which we will call “hexagons,” “squares,” and “brick hexagons.” (c) The coordinate system on the left shows the time (t) direction and defines the spatial x and y coordinates.

The transversal $\bar{T}$ gate [20] acts as T or T^{-1} on every vertex,

$$\bar{T} = \prod_{p \in \Delta_0} T_p^{\text{sgn}_p}, \quad T = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i \frac{1}{8}} \end{pmatrix}. \quad (4)$$

2. Color path integral from 3 + 0D protocol

Based on the 3D color code, consider the following measurement-based protocol:

- (1) Prepare the qubit at every vertex in the $|+\rangle$ state.
- (2) Measure S_f^Z at every face f , and apply X corrections such that all measurement outcomes are flipped to $+1$. This prepares a codestate of the 3D color code.
- (3) Apply the transversal T gate.
- (4) Measure every qubit in the X basis.

Note that in order to make this protocol useful, we would have to add inputs and outputs at spatial boundaries as in the previous section, which we do not consider here. The color path integral can be obtained from considering the $+1$ -postselected version of the above protocol. That is, instead of applying corrections, we replace the measurements in the second and fourth step by the projector for the $+1$ outcome. This postselected circuit can be written as a ZX tensor-network diagram. A ZX tensor network [33,39,40] is one consisting of X -tensors and Z -tensors. Both X and Z -tensors can be defined for an arbitrary number of indices and an arbitrary phase $0 \leq \alpha < 2\pi$. The Z -tensors force all their indices (valued in $\mathbb{Z}_2$) to take the same configuration,

$$c \text{---} \overset{b}{\underset{\alpha}{\bullet}} \text{---} a = e^{i\alpha a} \cdot \delta_{a=b=c=\dots} \quad (5)$$

Here, we are using a Kronecker- δ -symbol which is 1 if the condition in its subscript is fulfilled and 0 otherwise. The second family are X -tensors, whose entries only depend on the total parity of its indices,

$$c \text{---} \overset{b}{\underset{\alpha}{\circ}} \text{---} a = \sum_{x \in \mathbb{Z}_2} \frac{1}{2} (1 + e^{i(\alpha + \pi x)}) \cdot \delta_{a+b+c+\dots=x} \quad (6)$$

Note that the sum in the subscript is in $\mathbb{Z}_2$. To represent the protocol that we just described we only need to some specific phases $\alpha \in \{0, \pm\pi/4, \pi\}$, and use colors instead of

labels to indicate these phases,

$$\begin{array}{cc}
 \text{---} \circ \text{---} := \text{---} \circ^0 \text{---}, & \text{---} \bullet \text{---} := \text{---} \bullet^0 \text{---}, \\
 \text{---} \circ^{\pi/4} \text{---} := \text{---} \bullet^{\pi/4} \text{---}, & \text{---} \bullet^{\pi/4} \text{---} := \text{---} \bullet^{\pi/4} \text{---}, \\
 \text{---} \circ^{\pi} \text{---} := \text{---} \bullet^{\pi} \text{---}, & \text{---} \bullet^{\pi} \text{---} := \text{---} \bullet^{\pi} \text{---}
 \end{array} \quad (7)$$

Note that the phase factor assigned to the purple and green Z -tensors is that of the T and T^{-1} gate.

The postselected 3 + 0D protocol is a finite-depth circuit that creates qubits out of nothing, and in the end discards them by projecting them onto the $|+\rangle$ state. The operators in this circuit can be represented as simple ZX diagrams, which we interpret as linear operators from the bottom to the top in the following diagrams.

- (a) The $|+\rangle$ state preparation is an operator mapping from the trivial vector space $\mathbb{C}$ to the vector space of a qubit $\mathbb{C}^2$ and can be diagrammatically represented using a 1-index Z -tensor,

$$\begin{array}{c} a \\ | \\ \bullet \end{array} = 1 = \sqrt{2} \langle a|+\rangle \quad \forall a. \quad (8)$$

- (b) The +1-postselected $Z^{\otimes n}$ -measurement ($M_{Z^{\otimes n}}$) acting on the n vertices of a face has the familiar form [33] (here for $n = 6$),

$$\begin{array}{c}
 \begin{array}{cccccc}
 & c' & & f' & & \\
 b' & \bullet & a' & & d' & \bullet & e' \\
 & | & | & & | & & \\
 \bullet & c & & d & & f & \bullet \\
 & | & & | & & & \\
 b & & a & & d & & e
 \end{array} \\
 = \delta_{a=a'} \delta_{b=b'} \delta_{c=c'} \delta_{d=d'} \delta_{e=e'} \delta_{f=f'} \delta_{a+b+c+d+e+f=0} \\
 = \langle a', b', c', d', e', f' | (1 + Z^{\otimes 6}) | a, b, c, d, e, f \rangle.
 \end{array} \quad (9)$$

- (c) The T gate at every vertex p with $\text{sgn}_p = +1$ is a 2-index Z -tensor with $\pi/4$ phase,

$$\begin{array}{c} b \\ | \\ \bullet \\ | \\ a \end{array} = \delta_{a=b} e^{\frac{2\pi i}{8}} = \langle b|T|a\rangle, \quad (10)$$

- (d) The T^{-1} gate at every $\text{sgn}_p = -1$ vertex is defined analogously,

$$\begin{array}{c} b \\ | \\ \bullet \\ | \\ a \end{array} = \delta_{a=b} e^{-\frac{2\pi i}{8}} = \langle b|T^{-1}|a\rangle. \quad (11)$$

- (e) The postselected destructive single-qubit X measurement is a projection onto the $|+\rangle$ state, and thus the diagram is the time-reversed version of Eq. (8),

$$\begin{array}{c} \bullet \\ | \\ a \end{array} = 1 = \sqrt{2} \langle +|a\rangle \quad \forall a. \quad (12)$$

Composing the ZX building blocks above according to the circuit yields a ZX tensor-network diagram on the 3-colex. At every vertex, we first have a $|+\rangle$ preparation, then the qubit is involved in six $Z^{\otimes n}$ -measurements at the six adjacent faces, then we perform a T or T^{-1} gate, and, finally, a projection onto the $|+\rangle$ state. Each of these operations involves one Z -tensor, and all these Z -tensors can be fused into a single one (using the “spider-fusion” rule of the ZX calculus),

$$\begin{array}{c} \bullet \\ | \\ \bullet \\ | \\ \bullet \\ | \\ \bullet \\ | \\ \bullet \\ | \\ \bullet \\ | \\ \bullet \end{array} = \begin{array}{c} \diagup \quad \diagdown \\ \bullet \\ \diagdown \quad \diagup \end{array}. \quad (13)$$

After this fusion, the ZX tensor-network diagram takes a very simple form, as shown in Fig. 1: There is one Z -tensor at every vertex of the 3-colex, one X -tensor at every face, and one bond connecting the tensors at every vertex and adjacent face. All Z -tensors have six indices, and the X -tensors have some even number of indices depending on the chosen 3-colex (either four or six in the 3-colex we consider for our example). The Z -tensors have a phase of $\text{sgn}_p \pi/4$. We will to refer this ZX tensor network as the *color path integral*.

Note that we can also introduce an *untwisted color path integral*, where all Z tensors have trivial phases, corresponding to the same 3 + 0D protocol without T gates. This untwisted color path integral can also be turned into an *untwisted color circuit*, which is interesting in its

own right, but only yields Clifford gates. We will use the untwisted path integral and circuit for pedagogical reasons at some places in Secs. IV and V.

C. The twisted color circuit

In this section we turn the ZX color path integral back into a circuit, but one in $2 + 1D$ instead of $3 + 0D$. After choosing which of the directions in the 3-colex correspond to space and which to time, we apply some ZX *rewrite rules* to obtain a +1-postselected circuit. The real circuit is then obtained by simply replacing postselected measurements by real measurements. To retain fault tolerance in the non-post-selected case, we need to add decoding and corrections, as we discuss in Sec. II G and in more detail in Sec. V.

Our example twisted color circuit is obtained from the 3-colex in Fig. 1. We start by choosing space and time directions in our 3-colex, indicated in the coordinate system in Fig. 1. According to these chosen space and time coordinates we identify sets of bonds in the ZX -diagram that correspond to different qubits at a fixed time coordinate (called *qubit timeslices*), as well as sets of bonds that are represented by a fixed qubit at different times (called *qubit worldlines*). Both are shown in Fig. 1.

Projecting the 3-colex along the time direction yields a hexagonal lattice, as shown in Fig. 2. For each edge of the hexagonal lattice, there are two qubit worldlines, and accordingly two qubits. The next step is to turn the ZX diagram into a +1-postselected circuit of operators with respect to the chosen qubit worldlines and timeslices, using the ZX rewrite rules. In particular, we use the spider-fusion rules to split up some of the tensors into a group of “smaller” tensors, and then regroup the smaller tensors, such that each group can be viewed as an operator acting on the qubit worldlines. The resulting postselected circuit uses the following operators, expressed as ZX diagrams:

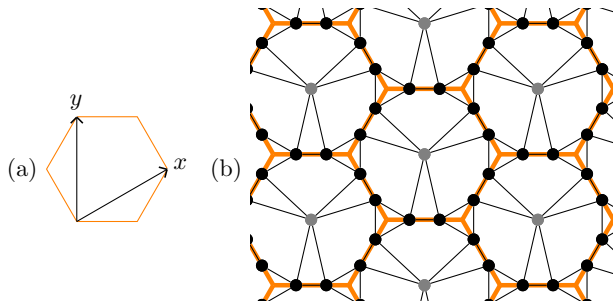


FIG. 2. (a) Spatial coordinate system. (b) Projection of the 3-colex of Fig. 1 along time direction yields a hexagonal lattice shown in orange. The qubits of the circuit are black dots. The black lines show the planar connectivity required by the 2-qubit gates in the circuit. In practice, the $Z^{\otimes 6}$ measurements require an ancilla qubit in the center of each face, which we indicate with gray dots.

- (a) A unitary controlled- X (CX) gate,

$$\begin{array}{c} c \quad d \\ | \quad | \\ \bullet \quad \circ \\ | \quad | \\ a \quad b \end{array} = \delta_{a=c=b+d} = \langle c, d | CX | a, b \rangle . \quad (14)$$

- (b) a +1-postselected 2-qubit XX -measurement M_{XX} ,

$$\begin{array}{c} c \quad d \\ \diagup \quad \diagdown \\ \circ \\ \diagdown \quad \diagup \\ a \quad b \end{array} = \delta_{a+b+c+d=0} = \langle c, d | (1 + XX) | a, b \rangle . \quad (15)$$

- (c) a +1-postselected 2-qubit ZZ -measurement M_{ZZ} ,

$$\begin{array}{c} c \quad d \\ \diagup \quad \diagdown \\ \bullet \\ \diagdown \quad \diagup \\ a \quad b \end{array} = \delta_{a=b=c=d} = \langle c, d | \frac{1}{2}(1 + ZZ) | a, b \rangle . \quad (16)$$

- (d) the +1-postselected measurement $M_{Z^{\otimes 6}}$ shown in Eq. (9), the T gate in Eq. (10), and the T^{-1} gate in Eq. (11).

Concretely, we will have

- (a) one M_{ZZ} measurement for every vertex of the 3-colex,
- (b) one CX gate for every vertex,
- (c) one M_{XX} measurement for every time-parallel brick hexagon or square,
- (d) and one $M_{Z^{\otimes 6}}$ measurement for every time-perpendicular hexagon.

Let us now show how we regroup the tensors of the ZX path integral to obtain these operators: Every Z -tensor at a vertex $p \in \Delta_0$ has six indices connecting it to X -tensors at (1) a past time-parallel brick hexagon, (2) a past time-parallel square, (3) a future brick hexagon, (4) a future square, (5) a time-perpendicular hexagon, and (6) a brick hexagon whose center has the same time coordinate as p . We split it up into one 4-index Z -tensor, two 3-index Z -tensors, and one 2-index Z tensor carrying the T^{sgn_p} phase,

$$\begin{array}{c} 4 \quad 3 \\ \diagup \quad \diagdown \\ \bullet \\ \diagdown \quad \diagup \\ 1 \quad 2 \end{array} = M_{Z^{\otimes 6}} \begin{array}{c} T \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagdown \quad \diagup \\ \bullet \quad \bullet \end{array} \begin{array}{c} CX \\ \diagup \quad \diagdown \\ \bullet \quad \bullet \\ \diagdown \quad \diagup \\ \bullet \quad \bullet \end{array} M_{ZZ} . \quad (17)$$

The labels at the indices correspond to the numbers of the list above. As shown, it gives rise to the M_{ZZ} measurement

at the vertex, contributes to half of the CX gate at the vertex, and contributes to one of the six Z -tensors of the $M_{Z^{\otimes 6}}$ measurement at the adjacent time-perpendicular hexagon (5). Next, the 6-index X -tensor at every time-parallel brick hexagon is split into one 4-index X -tensor and two 3-index X -tensors,

$$\text{Diagram (18): A central vertex with six lines extending from it. The lines are grouped into three pairs. The top pair is labeled CX , the bottom pair is labeled CX , and the middle pair is labeled M_{XX} . The entire diagram is enclosed in a dashed blue circle.$$
(18)

As shown, it gives rise to the M_{XX} measurement of the hexagon and contributes half of the CX gates at both of its middle-time-coordinate vertices. The 4-index X -tensor at every time-parallel square is not split up and simply gives rise to the M_{XX} measurement at the square. Finally, the 6-index X -tensor at every time-perpendicular hexagon forms the central X -tensor of the $Z^{\otimes 6}$ measurement.

The last step in deriving the twisted color circuit is to observe on which qubits the operators above act in which order, which is best done by looking at Fig. 1. The resulting circuit is shown in Fig. 3. In practice, one might perform the $M_{Z^{\otimes 6}}$ measurement using an ancilla qubit in the middle of the hexagon, via six CX gates and a single-qubit measurement, or more involved methods that use more ancillary qubits [41–43]. The resulting circuit then consists of 1- and 2-qubit operations and has fully planar connectivity as shown in Fig. 2.

1. Spacetime unit cell

If we ignore the difference between T and T^{-1} in the circuit, and the 4-coloring and signs on the vertices of the 3-colex, then both have the same minimal unit cell. Such a unit cell is spanned by translations under which the 3-colex in Fig. 1 is invariant. There are multiple ways of defining a minimal unit cell of the 3-colex. We choose one spanned by $\frac{1}{3}t + x$, $2x - y$, and $2y - x$ since this leads to the most compact representation of the circuit in Fig. 3 [44]. If we do distinguish between T and T^{-1} , and accordingly include the vertex signs, the minimal unit cell is enlarged by a factor of 2: The translation by $\frac{1}{3}t + x$ inverts the vertex signs, so we have to replace it by $2(\frac{1}{3}t + x)$. In Fig. 3 we incorporate this by swapping T and T^{-1} in step (4).

If we include the 4-coloring of the 3-colex, then the unit cell becomes larger. For example, the 3-colex is no longer invariant under a displacement by t but only by $4t$. As a consequence, the 12 time steps of the circuit corresponding to a t displacement implement some nontrivial logical Clifford unitary when put on a finite lattice with periodic boundary conditions, and only 48 time steps (or 16 time steps together with a shift by x) implement the logical identity. Further, for certain open boundary conditions, such as these discussed in Sec. IID, the minimal unit cell is that

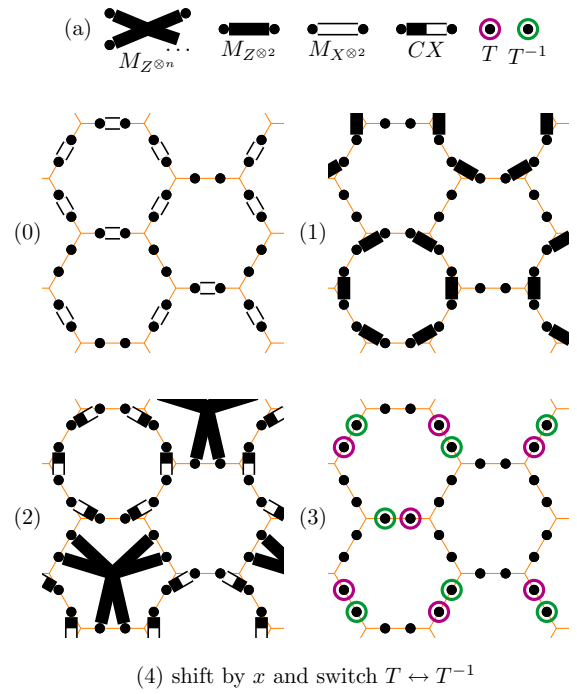


FIG. 3. (a) Symbols used for Z -type measurements, X -type measurements, CX , T , and T^{-1} gates. (0)–(3): 4 time steps of the circuit corresponding to a time translation by $\frac{1}{3}t$ in the 3-colex in Fig. 1. In step (0), we perform all the M_{XX} measurements resulting from the X -tensors on squares and brick-hexagons. In step (1), we perform all the M_{ZZ} measurements resulting from the Z -tensors on every vertex. In step (2), we perform all the $Z^{\otimes 6}$ measurements at the hexagons, as well as the CX gates related to the time-perpendicular bonds between the Z -tensors and brick-hexagon X -tensors. In step (3), we perform all the T or T^{-1} gates implementing the $e^{\pm 2\pi i/8}$ phases of every Z -tensor. After step (3), we continue with step (0), but with all operators shifted by x , and with T and T^{-1} in step (3) exchanged. One full time period thus consists of 12 steps.

of the 3-colex including the 4-coloring. Accordingly, a full time period of the boundary circuit consists of 48 instead of 12 steps. A third place where the 4-coloring matters is the decoding as we discuss in Sec. IIG.

D. Boundaries of the twisted color circuit

Let us next describe how to define open boundary conditions for our circuit. We cannot simply truncate the circuit in an arbitrary way, but we have to do it in a specific topologically protected way. Further, there are multiple distinct boundary conditions implementing different boundary topological phases, and we have to choose the ones that result in the desired logical action. For the $|T\rangle$ -state measurement protocol shown in Eq. (1), we need three different boundaries corresponding to three different boundary phases. We name these boundaries the r , g , and b color boundaries, after the related boundaries of the 3D

color code. Note that the three boundaries are related by permuting the four 3-colex colors, so it suffices to focus on the r boundary [45].

Constructing the circuit on the boundary is straightforward once we have defined suitable boundary conditions for the color path integral. Such boundary conditions correspond to ways to terminate the ZX diagram in a way that does not result in open indices at the boundary and that yields a macroscopic fault distance. Another way to view this is to cut the tensor network along some plane resulting in a *state boundary* (see Sec. IV) with open indices, and then close these open indices by contracting them with some auxiliary 2D boundary tensor network [34].

Geometrically, we define a color boundary along any surface consisting of faces of the 3-colex. The specific time-parallel boundary surface that we choose in our hexagon-prism 3-colex extends in t direction in a straight manner and in y direction in a zig-zag manner, corresponding to the surface in front in Fig. 4. The ZX diagram for the r color boundary on this surface is explained and depicted in Fig. 4. We will argue why these boundaries are topologically protected and give rise to the correct logical action later in Sec. IV E and Appendix A [46].

Next, we show how to turn the path-integral boundaries into a circuit, using the same method as in Sec. II C. Two representative qubit worldlines along the boundary are shown in Fig. 4. The projected spatial lattice is a hexagonal lattice with “zig-zag” boundary, as shown in Fig. 5. Just like in the bulk, there are two qubits on every boundary edge.

After identifying qubit worldlines, we slightly reshape the ZX diagram to turn it into a +1-postselected circuit. In addition to the operators introduced in Secs. II B and II C, the circuit contains postselected nondestructive single-qubit X measurements which are expressed as a ZX diagram as follows:

$$\begin{array}{c} b \\ \bullet \\ \bullet \\ a \end{array} = 1 = \langle a|+ \rangle \langle +|b \rangle \quad \forall a, b. \quad (19)$$

The square and brick-hexagon boundary faces that still have X -tensors at their center give rise to operators in the same way as in the bulk. The square boundary faces without X -tensor but two adjacent time-parallel red bonds correspond to doing nothing on the two qubits whose worldlines run through the bonds,

$$\begin{array}{cc} \bullet & \bullet \\ \vdots & \vdots \end{array} \begin{array}{cc} \bullet & \bullet \\ \vdots & \vdots \end{array} = \begin{array}{cc} \bullet & \bullet \\ \vdots & \vdots \end{array} \begin{array}{cc} \bullet & \bullet \\ \vdots & \vdots \end{array} \quad (20)$$

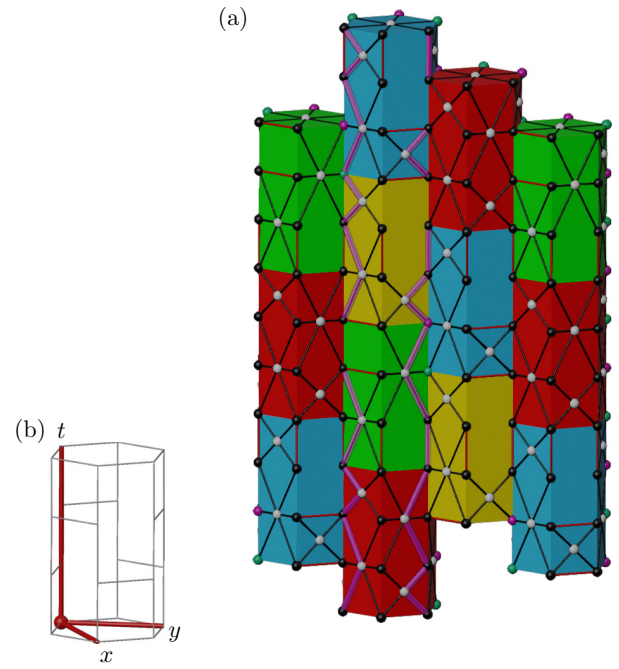


FIG. 4. (a) Color path integral with r color boundary on a snippet of 3-colex. The boundary surface is in front, extending along the (y, t) plane (in a zig-zag manner along y). The ZX diagram is a simple truncation of the bulk diagram (see Fig. 1) up to the following three considerations: (1) For every $\bar{r}$ -colored boundary edge, we add a new bond (drawn in red) connecting the Z -tensors at its vertices. (2) For every gb , gy , or by -colored boundary face, we may or may not include the X -tensor, which is the same up to rewrite rules. We chose to keep the X -tensor at “every second” of these boundary faces, since this is convenient for turning the path integral into a circuit. (3) We only include the $\text{sgn}_p \pi/4$ phases at the boundary vertices that are adjacent to a g , b , and y bulk volume, and the remaining boundary Z -tensors are phaseless. Two representative qubit worldlines inside the boundary are shaded in purple. (b) Coordinate system.

The square boundary faces without X -tensor but two adjacent time-perpendicular red bonds correspond to a projection of the two qubits onto the Bell state, which can be written as a +1-postselected ZZ followed by a XX measurement,

$$\begin{array}{cc} \bullet & \bullet \\ \vdots & \vdots \end{array} \begin{array}{cc} \bullet & \bullet \\ \vdots & \vdots \end{array} = \begin{array}{c} \bullet \\ \vdots \end{array} \begin{array}{c} \bullet \\ \vdots \end{array} \begin{array}{c} \bullet \\ \vdots \end{array} \begin{array}{c} \bullet \\ \vdots \end{array} \quad (21)$$

The ZX diagram around the boundary brick hexagons without X -tensor but a time-perpendicular red bond at its largest-time edge (in addition to two time-parallel red bonds) is rewritten such that it corresponds to two single-qubit X -measurements and one 2-qubit M_{ZZ}

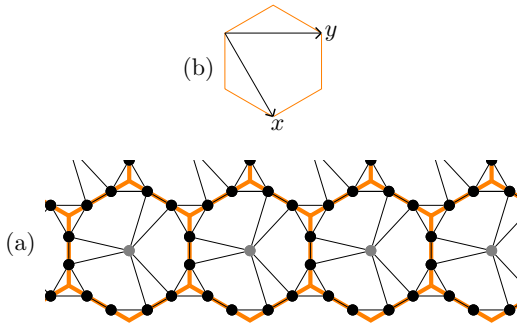


FIG. 5. (a) Spatial projection of the bulk 3-colex with boundary from Fig. 4, yielding a hexagonal lattice with “zig-zag” boundary at the bottom. Left, right, and top are not boundaries but truncations of the drawing. The qubits are represented by black dots, and there are two at every bulk and boundary edge. The black lines represent the connectivity of 2-qubit operations required by the circuit. (b) Coordinate system, with x and y coordinates consistent with Fig. 4(b), such that the bottom edge of (a) coincides with the front surface of Fig. 4(a). In the coordinate system of Fig. 2, the boundary is located on the right and extends in the vertical direction.

measurement,

$$\begin{array}{c} \cdots \\ \bullet \\ \cdots \end{array} \begin{array}{c} \cdots \\ \bullet \\ \cdots \end{array} = \begin{array}{c} \cdots \\ \bullet \\ \cdots \end{array} \begin{array}{c} \cdots \\ \bullet \\ \cdots \end{array} + \begin{array}{c} \cdots \\ \bullet \\ \cdots \end{array} \begin{array}{c} \cdots \\ \bullet \\ \cdots \end{array} + \begin{array}{c} \cdots \\ \bullet \\ \cdots \end{array} \begin{array}{c} \cdots \\ \bullet \\ \cdots \end{array} \quad (22)$$

The ZX diagram around the brick hexagons without X -tensor but with a red bond at the smallest-time edge is rewritten similarly,

$$\begin{array}{c} \cdots & \bullet & \cdots & \bullet & \cdots \\ & | & & | & \\ & \textcolor{red}{-} & & \textcolor{red}{-} & \\ & | & & | & \\ \cdots & \bullet & \cdots & \bullet & \cdots \end{array} = \textcolor{teal}{M}_X \begin{array}{c} \cdots & \bullet & \cdots \\ & | & \\ \cdots & \bullet & \cdots \\ & | & \\ \cdots & \bullet & \cdots \\ & | & \\ \cdots & \bullet & \cdots \end{array} \textcolor{teal}{M}_X \cdot \textcolor{teal}{M}_{ZZ} \begin{array}{c} \cdots & \bullet & \cdots \\ & | & \\ \cdots & \bullet & \cdots \\ & | & \\ \cdots & \bullet & \cdots \\ & | & \\ \cdots & \bullet & \cdots \end{array} \quad (23)$$

After these modifications, we are ready to read off the boundary circuit from Fig. 4. Since the ZX boundary path integral depends on the 4-coloring of the underlying 3-colex, the resulting circuit has an enlarged unit cell compared to the bulk. A convenient choice of unit cell is given by translations by $\frac{4}{3}t - y$, and translations by $3y$. So in order to specify a full unit cell of the circuit, we need to show how it acts on the boundary qubits of three spatial hexagons, and for a time period of $\frac{4}{3}t$, corresponding to four of the bulk cycles in Fig. 3. The resulting circuit is shown in Fig. 6.

E. Input and output domain walls

In this section we show how to transfer logical information from a 2D color code into the twisted color circuit and back, which will be the key to implementing a fault-tolerant non-Clifford operation. In other words, we implement a $2 + 0\text{D}$ interface—or *domain wall*—between a $2 + 1\text{D}$ syndrome-extraction circuit of the 2D color code and the twisted color circuit. To construct the interfacing circuit, we will start with a $3 + 0\text{D}$ protocol performing a dimension jump from the 2D to the 3D color code. Then we will turn the interface to the 3D color code into an interface to the color path integral from Sec. II B, and then finally turn this path integral back into the desired $2 + 0\text{D}$ interfacing section of the circuit.

1. 2D color code

Let us recall the definition of a 2D color code on $2\text{-}colex$, which is a 2-cellulation dual to a triangulation, whose faces are colored r , g , or b such that no adjacent faces have the same color [4]. There are two stabilizer generators for each face f , one acting as Z and X on all the vertices p of f , respectively:

$$S_f^Z = \prod_{p \in f} Z_p, \quad S_f^X = \prod_{p \in f} X_p. \quad (24)$$

For our example, we use a 2D color code on a hexagonal lattice as depicted in Fig. 7(a). In the global protocol of Eq. (1), we define the 2D color code on a global triangle with *color boundaries* labeled by r , g , and b . We implement the color boundaries in a slightly unusual way, such that they can be interfaced easily with the 3-colex underlying the twisted color circuit. In the following, we describe the r color boundary. The g and b boundaries are defined analogously. Geometrically, the boundary can be placed along an arbitrary path of edges in the primal 2-colex [47]. Specifically, we put the boundaries along a zig-zag direction of the hexagonal lattice as shown in Figs. 7(b) and 7(c). As shown in Fig. 7, the stabilizers at the boundary are determined by (1) removing Z stabilizers beyond the boundary, (2) truncating X stabilizers, (3) adding a new ZZ stabilizer at every $\bar{r}$ -colored boundary edge, and (4) removing the X stabilizers that do not commute with the new ZZ stabilizers.

2. Dimension-jump protocol for non-Clifford gate

Let us now discuss a $3 + 0D$ protocol that performs a non-Clifford gate by performing a dimension jump from the 2D to the 3D color code, then the transversal T gate of the 3D color code, and then jumps back to 2D. The logical information will be transferred into and from the 3D color code at a y color boundary. For simplicity, we will assume for the rest of this section that there are no

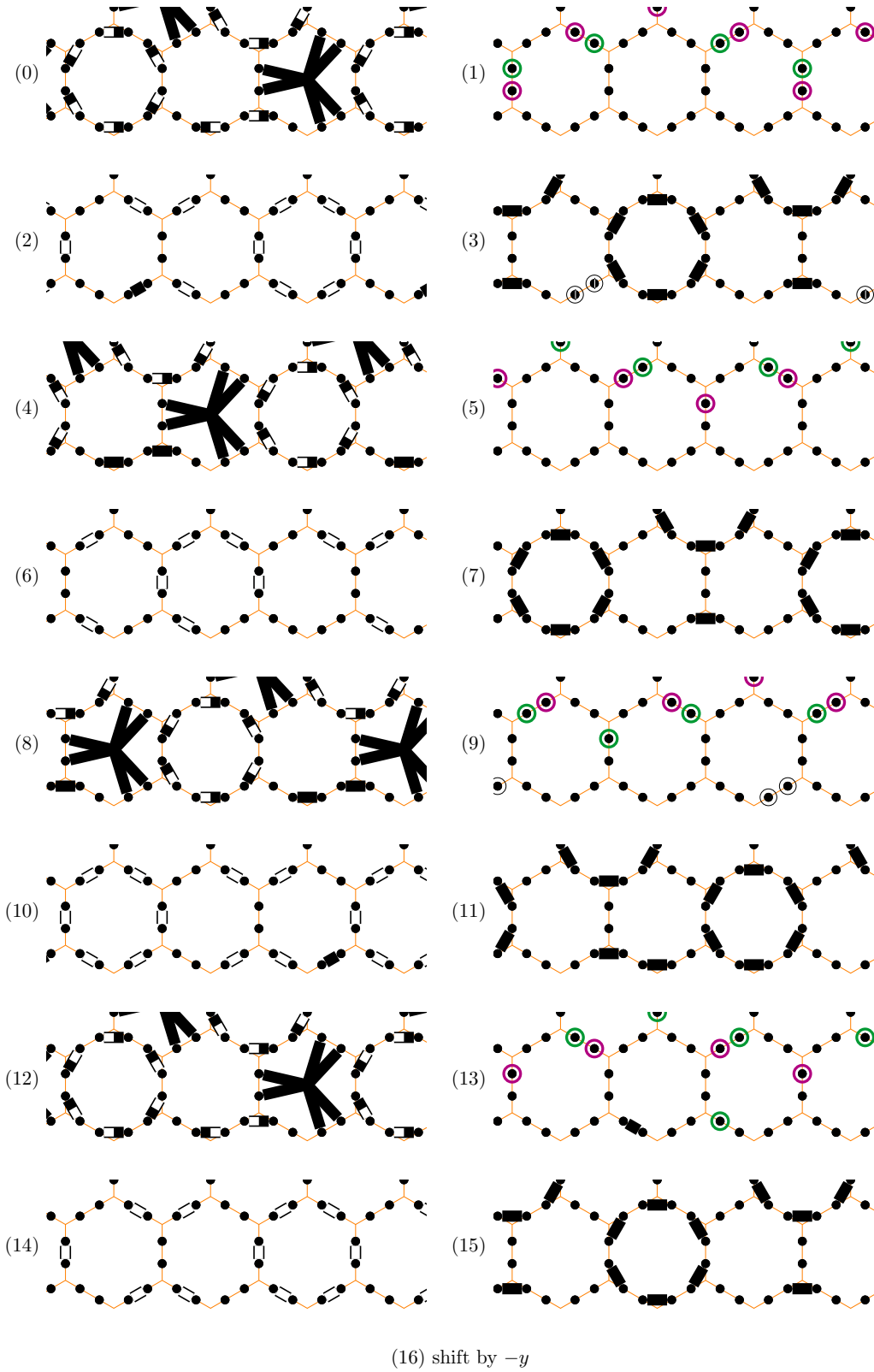


FIG. 6. (0)–(15): Circuit implementing an r color boundary on the qubits shown in Fig. 5. The symbols for gates are the same as in Fig. 3, with $\odot$ representing a single-qubit X measurement M_x .

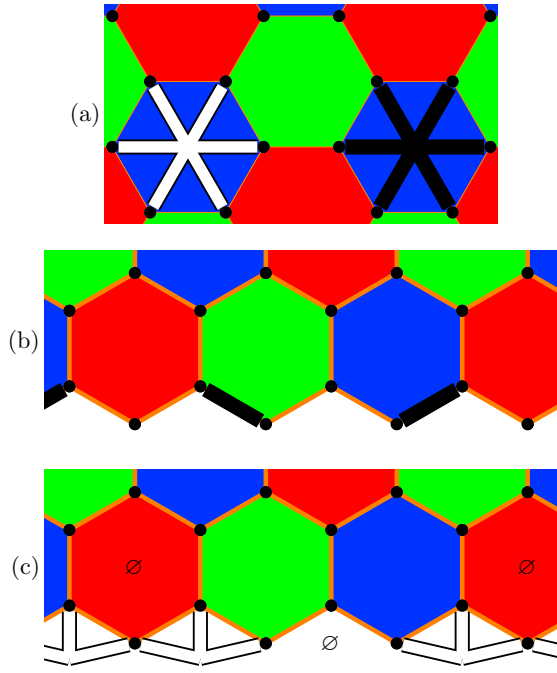


FIG. 7. (a) 2D color code on hexagonal 2-colex, with one qubit at every vertex, and one X and one Z -stabilizer at every face, using the notation from Fig. 3. (b) Additional ZZ -stabilizers at the $\bar{r}$ -colored edges of the bottom boundary. (c) Truncated X -type stabilizers. As indicated by the $\emptyset$ symbol, the $X^{\otimes n}$ operators are not stabilizers as they anticommute with the new ZZ stabilizers.

spatial boundaries of the 2D color code and in the resulting $2 + 1D$ protocol. To be concrete, let us assume that the 2D color code is supported on a torus $S_1 \times S_1$, and the 3D color code is supported on $S_1 \times S_1 \times [0, 1]$, with y -color boundaries at $S_1 \times S_1 \times 0$ and $S_1 \times S_1 \times 1$ where the logical information is transferred in and out. Apart from the different overall topology, the protocol is the same as introduced in Ref. [20,48].

To start, let us recall the y color boundaries of the 3D color code, which are analogous to the path-integral boundaries introduced in the previous section: We add one ZZ stabilizer for every $\bar{y}$ -colored boundary face, and remove all (truncated) X stabilizers that do not commute with these new ZZ stabilizers. Geometrically, we define the boundary along a time-perpendicular plane through the smallest-time hexagons of one layer of b -colored volumes, as shown in Fig. 8. Thereby, we introduce new hexagons along the boundary that truncate the r and g volumes. All boundary faces have color ry , by , or gy and form themselves an rgb -colored 2-colex. There are no $\bar{y}$ -colored boundary edges, so the boundary is simply defined with Z -type stabilizers at all faces and X -type stabilizers at all volumes.

With this, the protocol goes as follows: We start by identifying the qubits of the 2D color code with the qubits of

the 3D color code at the 2-colex of the $S_1 \times S_1 \times 0$ boundary. Then we initialize all remaining qubits of the 3D code in the $|+\rangle$ state and perform all Z -type stabilizer measurements of the 3D code. After performing corrections, we have switched from the 2D to the 3D color code. We then apply the transversal T^{sgn_p} gate of the 3D color code, which acts on the boundary qubits in the same way as it does on the bulk qubits. Finally, we perform a destructive single-qubit X measurement on all the qubits except for those on the $S_1 \times S_1 \times 1$ boundary 2-colex. After corrections, the resulting state on these 2-colex qubits will be a 2D color-code state.

3. Path integral from dimension-jump protocol

We now turn the above $3 + 0D$ protocol into a 2D domain wall between ZX tensor-network path integrals. Note that we do not work with an explicit 2D-color-code syndrome-extraction circuit or path integral. Instead, we construct the domain wall as a ZX tensor network with open indices, where we can attach the path integral for any 2D-color-code circuit. If the tensor network is contracted, these open indices support a 2D color-code state. The path integral is obtained by replacing all measurements with the according $+1$ -postselected projectors. The resulting ZX diagram is shown in Fig. 8. Every qubit becomes a Z -tensor, and every Z -type measurement becomes an X -tensor. In the $3 + 0D$ bulk, the protocol and path integral look like those in Sec. II B. Note that the path integrals for the input and output domain walls of the protocol at $S_1 \times S_1 \times 0$ and $S_1 \times S_1 \times 1$ are the same (apart from reflection).

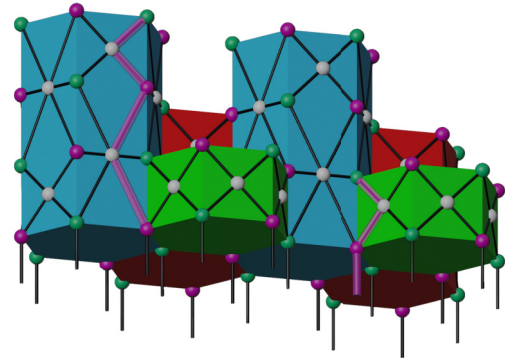


FIG. 8. Snippet of the 3-colex with path-integral domain wall to the 2D color code at the bottom. The open indices at the bottom support a 2D-color-code state on the bottom 2-colex. The domain wall cuts the rg brick hexagons down to squares, such that the according 6-index X -tensors become 4-index X -tensors. One can add X -tensors to all hexagons inside the domain wall—this is equivalent to the shown ZX diagram up to rewrite rules. Two representative worldlines are shaded in purple: Only the worldlines contained in the rg -faces near the domain wall continue as 2D-color-code worldlines, the remaining ones begin or terminate on the domain wall.

4. Circuit from path integral

Next, we turn the above path integral into a circuit. There are two circuits, one transferring logical information from the 2D color code into the twisted color circuit, and vice versa. We start with the former. We first choose how the qubit worldlines propagate through the domain wall. There is one qubit per 2-colex vertex in the 2D color code, but there are three qubit worldlines per vertex in our twisted color circuit. Thus, we need to choose one of the three worldlines to continue the 2D-color-code qubit worldline. The other two twisted-color-circuit worldlines begin at the temporal domain wall, meaning that these qubits were unused in the 2D color code, and can be seen as auxiliary qubits needed to perform the non-Clifford logic operation. Figure 8 shows one worldline that continues through, and one worldline that begins at the domain wall.

After choosing the qubit worldlines, we slightly rewrite the ZX diagram above into a circuit, whose input is the color code state at the rgb 2-colex corresponding to the bottom boundary. Every Z -tensor at every domain-wall vertex has four indices: (0) The open index, and the indices connecting it with the X -tensors at the adjacent time-parallel (1) rb brick hexagon, (2) rg square, and (3) gb square. We rewrite each such tensor as a little circuit consisting of two $|+\rangle$ initializations and two $+1$ -postselected ZZ measurements,

$$\begin{array}{c} 3 \\ | \\ 2 \\ | \\ \bullet \\ | \\ 0 \end{array} \begin{array}{c} 1 \\ | \\ 0 \end{array} = \begin{array}{c} 3 \\ | \\ 2 \\ | \\ \bullet \\ | \\ 0 \end{array} \begin{array}{c} 1 \\ | \\ 0 \end{array} \begin{array}{c} M_{ZZ} \\ \bullet \\ |+\rangle \\ \bullet \\ M_{ZZ} \\ \bullet \\ |+\rangle \end{array} \quad (25)$$

The resulting circuit implementing the domain wall is shown in Fig. 9.

5. Switching back

We also need to construct the circuit to transition from the twisted color circuit to the 2D color code. The ZX diagram is the same as in Fig. 8, just reflected vertically if time goes upward. The qubit worldlines are also the same as in Fig. 8. Similar to Eq. (25), the 4-index Z -tensors at the domain-wall vertices are split up into two $+1$ -postselected ZZ measurements, and two $+1$ -postselected destructive X measurements:

$$\begin{array}{c} 0 \\ | \\ \bullet \\ | \\ 3 \end{array} \begin{array}{c} 2 \\ | \\ 1 \end{array} = \begin{array}{c} \langle + | \\ \bullet \\ M_{ZZ} \\ \bullet \\ \langle + | \end{array} \begin{array}{c} 2 \\ | \\ \bullet \\ | \\ 0 \end{array} \begin{array}{c} 1 \\ | \\ 3 \end{array} \begin{array}{c} M_{ZZ} \\ \bullet \\ |+\rangle \\ \bullet \\ M_{ZZ} \\ \bullet \\ |+\rangle \end{array} \quad (26)$$

The resulting circuit is shown in Fig. 10.

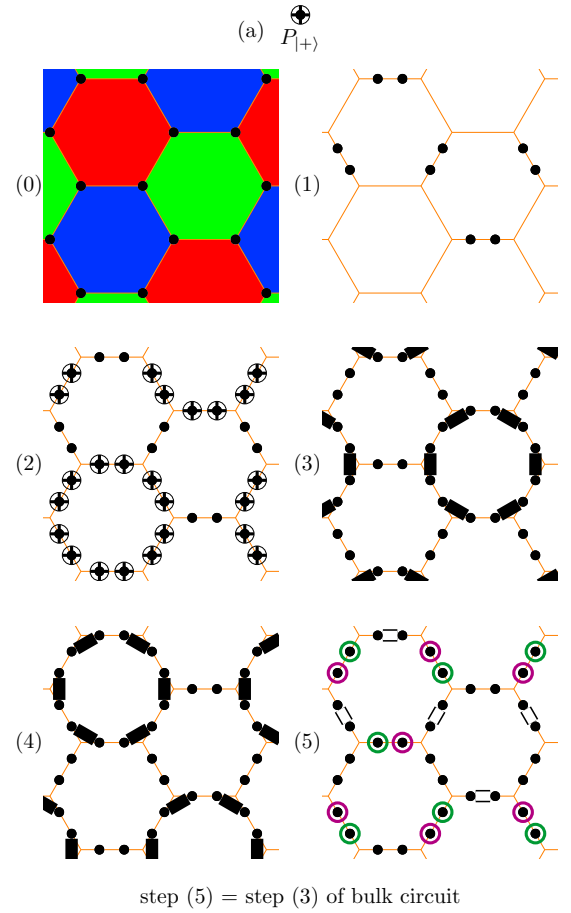


FIG. 9. (a) Symbol for the $|+\rangle$ state preparation gate. The rest of the gate symbols are the same as in Fig. 3. (0)–(5): Circuit transferring the logical information of the 2D color code to the twisted color circuit. We start with the qubits of the 2D color code (0) and shift the qubits slightly onto the adjacent $\bar{b}$ -colored edges (1). Then, we initialize the remaining qubits of the twisted color circuit in the $|+\rangle$ state (2) and perform the ZZ measurements between the moved and the initialized qubits (3 and 4), which implements all the tensors on the right-hand side of Eq. (25). In step (5), we perform XX measurements on all $\bar{b}$ -colored edges of the 2-colex, which implement the X -tensors at the rg -squares of the 3-colex in Fig. 8. After this, we continue with step (3) of the bulk protocol in Fig. 3, whose T and T^{-1} gates we have already included in step (5) above.

F. Corners

So far, we have constructed the microscopic circuits for the 3D bulk volumes of the global protocol in Eq. (1) as well as for the 2D boundaries and domain walls. The next step is to say what happens at the 1D lines where different boundary conditions or domain walls meet. Such interfaces between boundary conditions are lines in spacetime but points in space and are therefore known as *corners* in quantum error correction. There are three kinds of corners in our protocol: (a) the corners of the 2D color code, (b) time-parallel corners separating two color boundaries of the

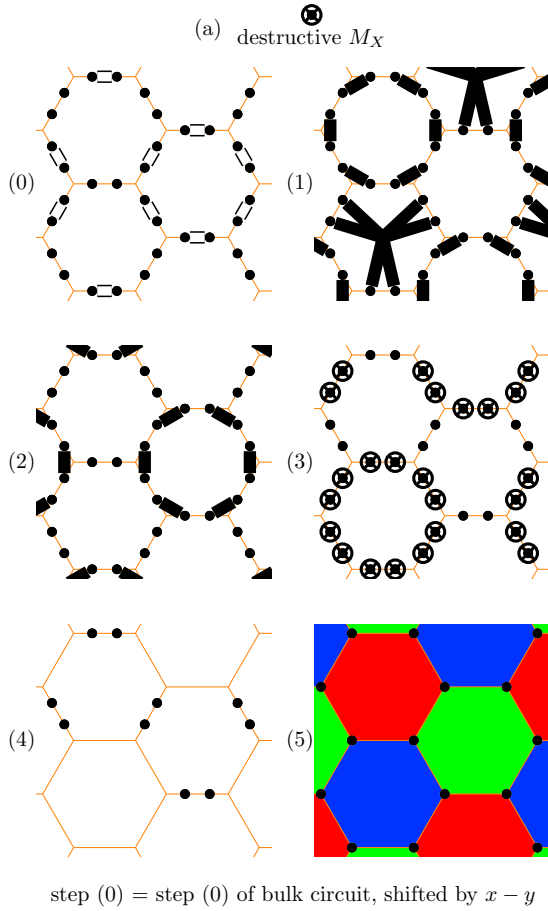


FIG. 10. (a) Symbol for destructive single-qubit M_X measurement, the remaining symbols are these in Fig. 3. (0)–(5): Circuit transferring the logical information of the twisted color circuit back to the 2D color code. Step (0) corresponds to step (0) of the twisted color circuit in Fig. 3.

twisted color circuit, and (c) time-perpendicular corners between the 2D color-code boundary and the according twisted-color-circuit boundary, where also the domain wall terminates. We will not describe these corners in detail here, but the according circuits can be constructed straightforwardly from ZX tensor networks as done for the bulk, boundaries and domain wall in the preceding subsections. Figure 11 illustrates the stabilizers and ZX tensor networks for the different kinds of corners, from which we can read off the circuits.

G. Decoding

In this section, we will sketch how to decode twisted color circuits. The aim here is to address most important aspects of the decoding procedure, without going into the combinatorial details. These details will be discussed later in Sec. V, after introducing the appropriate vocabulary in Secs. III and IV.

1. 3 + 0D protocol

We first recall how decoding works for the 3 + 0D measurement-based dimension-jump protocol [20] that we have used to derive the 2 + 1D twisted color circuit. Suppose that every Z -type face measurement and every single-qubit X measurement is faulty with a small probability. We denote the subset of faces that experience a Z -measurement error by b_E , and the subset of faces yielding a -1 measurement outcome by b . Both b_E and b can be pictured as a configuration of strings on the 3D spacetime lattice, and we refer to them as *flux configurations* [49]. Taken together, the symmetric difference $b + b_E$ must fulfill certain local parity or “Gauss law” constraints, which ensure that the strings form closed loops. That is, by measuring b , we measure the endpoints of the error configuration b_E . The endpoints (denoted db) of b (and b_E) are often called *detection events* and form the *flux syndrome*. The task of the decoder is to find an estimate $\tilde{b}$ for b_E , which must have the same endpoints db , and which we thus refer to as the *closing flux configuration*. Since larger errors have a small probability, a good strategy is to choose $\tilde{b}$ to be a low-weight configuration. Note that this task is equal to the syndrome-fixing step in the single-shot decoding of X errors in the 3D color code [50].

After performing all the Z face measurements and decoding, we need to apply corrections that map the state onto a 3D color-code state with all Z -type measurement outcomes equal to $+1$. The corrections consist of Pauli- X operators acting on a subset G of vertices, which we call the *gauge-fixing configuration*, and which we can picture as a membrane configuration. Since the effective configuration of -1 outcomes is given by $b + b_E$, choosing a membrane configuration G whose boundary is the closed-loop configuration is $b + b_E$ yields an exact code state. As we do not have access to b_E , we replace it with its estimate $\tilde{b}$, and choose an arbitrary G that bounds $b + \tilde{b}$ instead. The state after corrections is then one with a *residual flux configuration* $b_E + \tilde{b}$.

Let us next describe the decoding for measurement errors in the single-qubit X measurements at the end of the protocol. We denote the set of vertices that experience a measurement error by c_E , and the set of vertices with a -1 outcome by c . Again, both c and c_E can be pictured as string patterns, and we refer to them as *charge configurations*. If we omit the transversal T gate in the protocol, then the overall charge configuration $c + c_E$ must form closed loops. So a decoder can estimate c_E by finding a low-weight string pattern $\tilde{c}$, whose endpoints are given by the endpoints $d^T c = d^T c_E$ of c , forming the charge syndrome. This decoding problem is equal to the decoding of on-site Z errors in the 3D color code without measurement errors.

If we do apply the transversal T gate before the single-qubit X measurements, the overall charge configuration $c + c_E$ no longer needs to be closed [51]. Instead, $c + c_E$

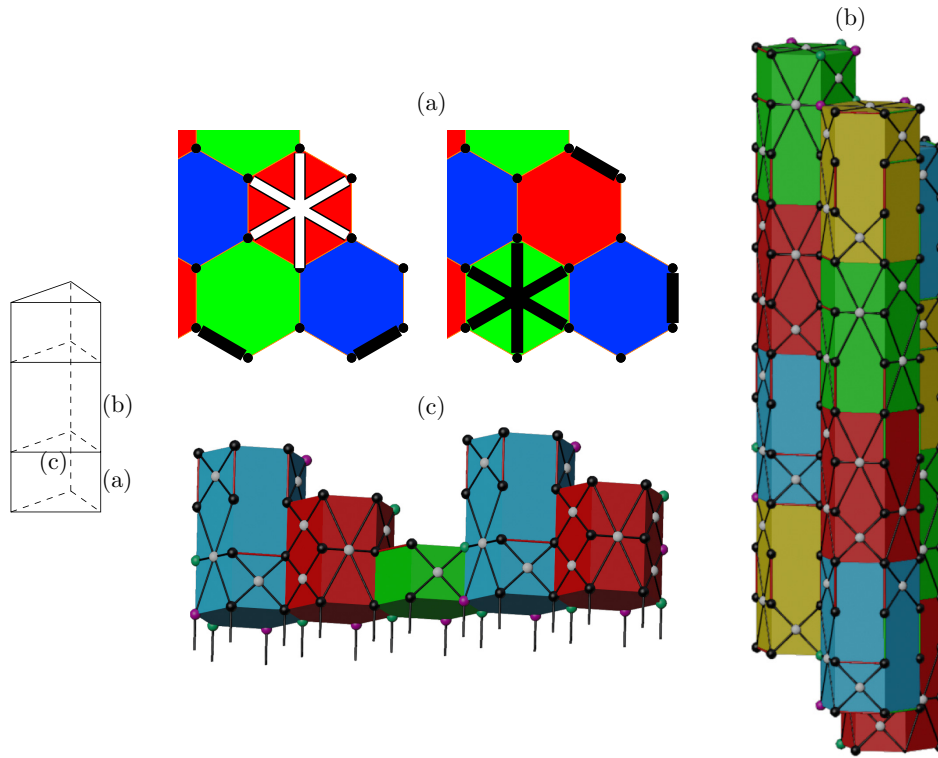


FIG. 11. (Left) Three different types of corners in the protocol, labeled (a), (b), and (c). (a) Corner of the 2D color code between a r (bottom) and a g (top right) zig-zag color boundary. Each of the two pictures shows some of the stabilizer generators near the corner, which is located at the bottom-right vertex of the bottom-right blue face. The vertex on the corner is involved in two ZZ stabilizers, since it is adjacent to a $\bar{r}$ edge of the r boundary, and a $\bar{g}$ edge of the g boundary. (b) Corner between a r (front left) and a g (front right) color boundary of the color path integral. (c) Corner joining the r boundary of the 2D color code and the r boundary of the color path integral (front surface), and the domain wall between 2D color code and color path integral (bottom surface).

forms a string pattern that may terminate anywhere on the residual flux configuration $b_E + \tilde{b}$ [52]. The reason for this is that after applying the transversal T gate to a state with flux configuration $b_E + \tilde{b}$, it is no longer an eigenstate of the X -stabilizers of the 3D color code that detect the charge syndrome. Thus, the residual flux configuration $b_E + \tilde{b}$ behaves similar to a heralded erasure, where a random charge error happens with high probability. In other words, the residual flux configuration reduces the fault distance for Z errors (or X measurement errors) in the protocol. If the error rate for Z measurement errors is low, then the X measurement errors are still decodable. Also note that it is possible to use the knowledge of $\tilde{b}$ to improve the performance of the X -measurement decoding. Further, we would like to point out that when decoding the protocol without T gates, only the cohomology class of $\tilde{b}$ matters. In contrast, with the T gate, $\tilde{b}$ itself matters and should have low weight.

Finally, we note that even though it is convenient to explain decoding for Pauli- X and Z errors, the path-integral perspective allows us to consider arbitrary errors [28]. Let E denote the subset of locations in the 3-colex where some arbitrary errors happen. It is still true that the endpoints of b must be located inside E , and the

endpoints of c must be located either inside E or on $\tilde{b}$, as shown in Fig. 12(a). This is sufficient to show fault tolerance, see Ref. [17], and we elaborate on further on this in Appendix B.

2. 2 + 1D protocol

Let us now describe how decoding changes if we turn the 3 + 0D protocol into a 2 + 1D protocol using the method of Refs. [22,23], by keeping only a few 2D slices of qubits active at a time. We need to apply the Pauli- X corrections along the gauge-fixing configuration G while we execute the 2 + 1D circuit. Since G is chosen to bound $b + \tilde{b}$, we must also choose $\tilde{b}$ while executing the circuit. That is, our choice of $\tilde{b}_t$ at a time t can only depend on b (or better, its syndrome db) at times earlier than t . In particular, we cannot use a “global” decoder operating on the 3D spacetime at the end. Instead, we have to use *just-in-time* decoding as proposed in Ref. [22]: Let $\tilde{b}_t$ denote the portion of $\tilde{b}$ located in between time steps $t - 1$ and t , and let $\tilde{b}_{\leq t}$ denote the portion of $\tilde{b}$ before time t . Then at time step t , the decoder must finalize its choice of $\tilde{b}_t$, depending only on $(db)_{\leq t}$. This choice is made in the following two subroutines: (1) The first step is to use a global decoder on

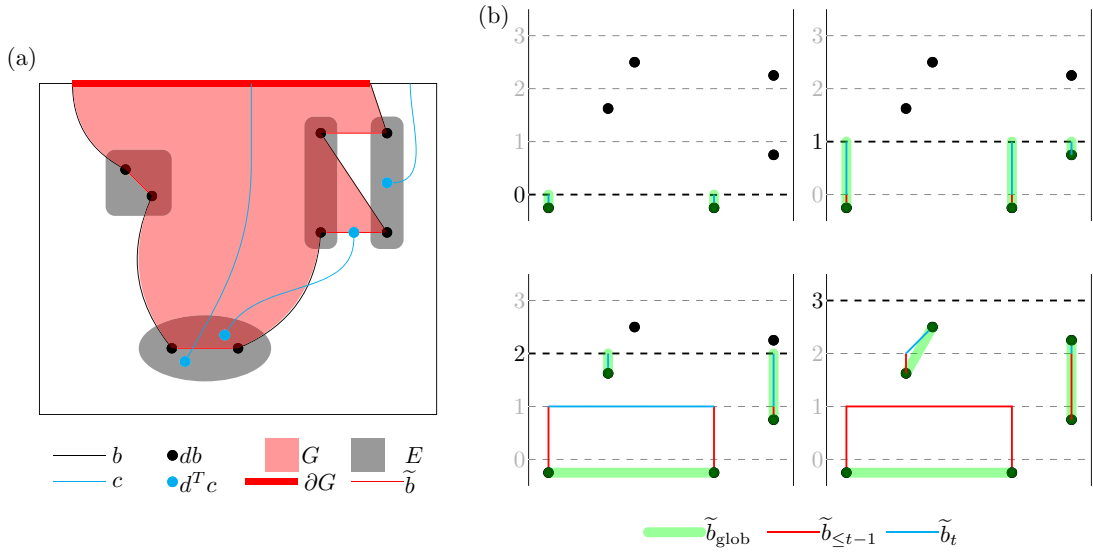


FIG. 12. (a) 2D toy picture illustrating the decoding and error correction in the 3 + 0D protocol, with output state at the top boundary. b is the flux configuration corresponding to the face Z measurements. Its endpoints db form the flux syndrome which can only be supported in the vicinity of the errors E . $\tilde{b}$ is the closing flux configuration, which is a low-weight string configuration chosen by the decoding algorithm, with the same endpoints db as b . G is the gauge-fixing configuration, an arbitrary membrane configuration whose boundary is given by $b + \tilde{b}$. c is the charge configuration corresponding to the vertex X measurements. Its endpoints $d^T c$ form the charge syndrome which can be supported in the vicinity of both the errors E and $\tilde{b}$. (b) Four time steps of just-in-time decoding algorithm. Time is going upward, and the spatial directions extend horizontally. The time slices are the horizontal gray dashed lines, with the current time slice drawn in thick and black. $\tilde{b}_{\leq t-1}$ is the portion of $\tilde{b}$ chosen before time step $t - 1$. $\tilde{b}_t$ is the portion of $\tilde{b}$ between $t - 1$ and t , which is chosen at the current time step. As shown, if an error creates a pair of flux syndrome points that are far apart spatially, then $\tilde{b}$ as chosen by the just-in-time decoder will not connect them immediately, but with some temporal delay proportional to their spatial separation.

$(db)_{\leq t}$, which yields a closing flux configuration b_{glob} , such that $b_{\leq t} + b_{\text{glob}}$ forms strings that may terminate at time t but are otherwise closed. In general, this global choice of b_{glob} is not directly compatible with the choices $\tilde{b}_{\leq t-1}$ made by the just-in-time decoder so far. (2) The next step is to reconcile b_{glob} with $\tilde{b}_{\leq t-1}$. To this end, we note that the endpoints of b_{glob} and $\tilde{b}_{\leq t-1}$ differ only between $t - 1$ and t . We can use our global decoding algorithm again to match up these endpoints. The result is $\tilde{b}_t$. The procedure is illustrated in Fig. 12(b). After decoding the flux syndrome in a just-in-time way, the charge syndrome is decoded as in the 3 + 0D protocol with a global decoder.

3. Twisted color circuits

For our twisted color circuits, there are some small additional differences to the circuits in Refs. [22,23]. First, the configuration of Z -type measurement results is no longer the same as the flux configuration b (which we define as a subset of 3-colex faces). Instead, some of the Z -measurements contribute multiple 3-colex faces to b . The same is true for the X -measurements and the charge configuration c . Second, the “gauge-fixing” operation is not

given by applying a Pauli- X operator along the gauge-fixing configuration G . Instead, in our circuit, it is given by conjugating the T^{sgn_p} gate by X , or equivalently, exchanging T and T^{-1} along G . We elaborate more on the just-in-time decoding procedure for twisted color circuits in Sec. V.

III. COLOR COHOMOLOGY

In this section, we introduce *color cohomology*, which is a cohomology theory defined on 3-colexes. As we show in Sec. IIIC, color cohomology is equivalent to $\mathbb{Z}_2^3$ -valued *cellular cohomology*, the type of cohomology theory that describes toric-code protocols. Color cohomology is essential for studying the topological nature of the path integral and its defects in Sec. IV, as well as for formulating the decoding algorithms in Sec. V. A length-2 subcomplex of this cohomology theory can also be found in Ref. [30], and the 2D case was first introduced in Ref. [53].

A. $\mathbb{Z}_2$ -valued cohomology

Let us start by introducing general $\mathbb{Z}_2$ -valued cohomology theories. A cohomology theory is defined by a *chain complex*, which is determined by

- (a) a length $n \in \mathbb{Z}_+$,
 (b) a sequence of finite *generator sets*,

$$\{X_i\}_{0 \leq i \leq n}, \quad (27)$$

- (c) a sequence of *boundary matrices*,

$$\{d_i \in \mathbb{Z}_2^{X_{i+1} \times X_i}\}_{0 \leq i < n}, \quad (28)$$

- (d) which satisfy

$$\sum_{y \in X_{i+1}} (d_{i+1})_{zy} (d_i)_{yx} = 0 \\ \forall i \leq n-2, x \in X_i, z \in X_{i+2}. \quad (29)$$

Usually, the boundary matrices are interpreted as group homomorphisms (or, similarly, as $\mathbb{Z}_2$ -linear maps between finite-field vector spaces),

$$d_i : \mathbb{Z}_2^{X_i} \rightarrow \mathbb{Z}_2^{X_{i+1}}, \quad (d_i v)_x = \sum_y (d_i)_{xy} v_y. \quad (30)$$

Then, Eq. (29) simply becomes

$$d_{i-1} d_i = 0. \quad (31)$$

Sometimes, when no confusion will arise, we will drop the subscript i of d_i and we just write d . Also, note that we will sometimes let the sequence X_i start from $i = -1$ rather than $i = 0$. It is custom to denote a chain complex as a sequence of groups with the homomorphisms as arrows,

$$\mathbb{Z}_2^{X_0} \xrightarrow{d_0} \mathbb{Z}_2^{X_1} \xrightarrow{d_1} \mathbb{Z}_2^{X_2} \xrightarrow{d_2} \dots, \quad (32)$$

will just write X_i instead of $\mathbb{Z}_2^{X_i}$ in such sequences for brevity.

Elements of the group $\mathbb{Z}_2^{X_i}$ are called *i-(co-)chains* [54]. Note that, for some $x \in X_i$, we will sometimes use x to also denote the i -chain c that is nonzero only on x , $c_y = \delta_{y=x}$. In this sense, the elements $x \in X_i$ can be understood as generating i -chains. An i -chain A is called an *i-cocycle* if $d_i A = 0$, and an *i-cycle* if $d_{i-1}^T A = 0$. Further, A is called *i-coboundary* if there exists a $i-1$ -chain B such that $d_{i-1} B = A$, and *i-boundary* if there is an $i+1$ -chain B such that $d_i^T B = A$. Due to Eq. (29), every i -(co)boundary is a i -(co)cycle. Just like the (co-)chains, the (co-)cycles and (co-)boundaries form groups under element-wise $\mathbb{Z}_2$ addition. The quotient of the group of cocycles by the group of

coboundaries,

$$H^i \simeq \text{Kernel}(d_i) / \text{Image}(d_{i-1}) \quad (33)$$

is called the *ith cohomology group*, and its elements are called *ith cohomology classes*. Dually, the quotient

$$H_i \simeq \text{Kernel}(d_{i-1}^T) / \text{Image}(d_i^T) \quad (34)$$

is the *ith homology group* and its elements are the *ith homology classes*. Homology and cohomology groups can be identified [55].

1. Chain maps

Finally, there also is a way to map between different cohomology theories, whose generator sets and boundary maps we distinguish by superscripts (a) and (b) : A *chain map* M from (a) to (b) consists of a collection of matrices

$$\{M_i \in \mathbb{Z}_2^{X_i^{(b)} \times X_i^{(a)}}\}_{0 \leq i \leq n}, \quad (35)$$

which we can interpret as homomorphisms from the group of (a) i -(co-)chains to the group of (b) i -(co-)chains, such that

$$d_i^{(b)} M_i = M_{i+1} d_i^{(a)} \quad \forall 0 \leq i < n. \quad (36)$$

A chain map can also be denoted as a commutative diagram involving the two chain complexes,

$$\begin{array}{ccccccc} X_0^{(b)} & \xrightarrow{d_0^{(b)}} & X_1^{(b)} & \xrightarrow{d_1^{(b)}} & X_2^{(b)} & \xrightarrow{d_2^{(b)}} & \dots \\ \uparrow M_0 & & \uparrow M_1 & & \uparrow M_2 & & \\ X_0^{(a)} & \xrightarrow{d_0^{(a)}} & X_1^{(a)} & \xrightarrow{d_1^{(a)}} & X_2^{(a)} & \xrightarrow{d_2^{(a)}} & \dots \end{array} \quad (37)$$

M_i maps (a) i -cocycles to (b) i -cocycles and (a) i -coboundaries to (b) i -coboundaries, so it yields a homomorphism from $(H^i)^{(a)}$ to $(H^i)^{(b)}$. Similarly, M_i^T maps (b) i -cycles to (a) i -cycles and (b) i -boundaries to (a) i -boundaries, and yields a homomorphism from $(H_i)^{(b)}$ to $(H_i)^{(a)}$.

2. Example: Cellular cohomology

After introducing $\mathbb{Z}_2$ -valued cohomology theories in general, let us briefly consider *cellular cohomology*, which is a length- n cohomology theory defined for any n -cellulation. We denote by Δ_i the set of i -cells, and define (for $0 \leq i, j \leq n$) the (i, j) -adjacency matrix ∂_{ij} as [56]

$$\partial_{ij} \in \mathbb{Z}_2^{\Delta_i \times \Delta_j}, \\ (\partial_{ij})_{xy} = \begin{cases} 1 & \text{if } x \subseteq y \text{ or } y \subseteq x \\ 0 & \text{otherwise} \end{cases}. \quad (38)$$

Here, $y \subseteq x$ means that the i -cell y is contained in the boundary of the j -cell x , or that $x = y$. Note that we will

usually just write ∂_{xy} instead of $(\partial_{ij})_{xy}$ if it is understood that $x \in \Delta_i$ and $y \in \Delta_j$. To help with this, we will usually denote vertices by p (“point”), edges by e , faces by f , and volumes by v . With this, the generator sets and boundary matrices for cellular cohomology are given by

$$X_i := \Delta_i \quad \forall 0 \leq i \leq n, \quad d_i := \partial_{i+1,i} \quad \forall 0 \leq i < n. \quad (39)$$

Equation (29) becomes

$$\sum_{y \in \Delta_{i+1}} \partial_{zy} \partial_{yx} = 0 \quad \forall x \in \Delta_i, z \in \Delta_{i+2}. \quad (40)$$

The left-hand side is the mod 2 number of $i+1$ -cells y adjacent to both the i -cell x and the $i+2$ -cell z . If $x \subset z$, then there are two such $i+1$ -cells y , otherwise there are none, so the equation holds.

All considered cohomology theories have a property that we call *local exactness* [57]: First, the generators are associated to places on some sort of lattice or graph with a notion of distance, and the coboundary maps are local, that is, zero beyond some constant distance. Second, any (co)cycle that is locally supported on the lattice or graph is a (co)boundary. This holds in particular for cellular cohomology.

Let us briefly give some intuition for cellular cohomology. We note that for an i -cell $c \in \Delta_i$, $d^T c$ consists of the boundary $i-1$ -cells of c . Thus, i -cocycles are subsets of i -cells with no boundary. In particular, 1-cycles are superpositions of closed loops consisting of lattice edges, 2-cycles are superpositions of closed membranes consisting of faces, and so on. Homology classes are equivalence classes of closed loops or membranes under local deformations. For example, the homology group on a torus is $\mathbb{Z}_2^2$, and each element of $\mathbb{Z}_2^2$ corresponds to an equivalence class of closed-loop patterns on the torus, which can be characterized by their vertical and horizontal mod 2 winding number. Homological codes encode quantum information nonlocally into the homology class [1]. For example, in the toric code, there is a basis of codestates labeled by cohomology classes: Each computational basis state is the equal-weight superposition of all 1-cycles in the same cohomology class.

We can extend cellular cohomology to a length- $n+2$ chain complex by adding two coboundary maps at the beginning and end, corresponding to

$$X_{-1} = X_{n+1} := \{0\}, \quad (d_{-1})_{p0} = 1 \quad \forall p \in \Delta_0, \quad (d_n)_{0v} = 1 \quad \forall v \in \Delta_n. \quad (41)$$

Roughly, d_{-1} and d_n can be viewed as adding “global conservation laws” for 0-cycles and n -cocycles.

B. Color cohomology: Definition

After introducing general $\mathbb{Z}_2$ -valued cohomology, and cellular cohomology as a paradigmatic example, let us now define color cohomology. We start by revisiting and extending our notation related to the 4-coloring of a 3-colex given at the beginning of Sec. II B. Let $\mathcal{C}_i$ for $0 \leq i \leq 4$ denote the set of subsets of $\mathcal{C} := \{r, g, b, y\}$ with i elements. We define the *color* $\mathbf{c}_x \in \mathcal{C}_{4-i}$ of a 3-colex i -cell $x \in \Delta_i$ as the subset of colors of its $4-i$ adjacent volumes, and we denote the set of i -cells $x \in \Delta_i$ with $\mathbf{c}_x = c$ by Δ_i^c . We will denote the union of two disjoint subsets $c_i \in \mathcal{C}_i$ and $c_j \in \mathcal{C}_j$ by $c_i c_j \in \mathcal{C}_{i+j}$ or by $c_j c_i$, and the complement of $c_i \in \mathcal{C}_i$ by $\bar{c} \in \mathcal{C}_{4-i}$, and we identify $\mathcal{C}_1$ with $\mathcal{C}$, e.g., we write $\{r\}$ simply as r . Taken together, we denote $c \in \mathcal{C}_i$ by listing its $\mathcal{C}$ -elements in arbitrary order, for example $rgy \in \mathcal{C}_3$. We use analogous notation for 2-colexes.

1. 2D

To prepare for the 3D case, let us start by discussing the analogous *2D color cohomology*, which is a locally exact length-2 chain complex defined for any 2-colex. It is instructive to define a larger structure of generator sets and homomorphisms between them,

$$\begin{array}{ccccccc} & & & & & & X'_3 \\ & & & & & & \uparrow m_3 \\ X_{-1} & \xrightarrow{d_{-1}} & X_0 & \xrightarrow{d_0} & X_1 & \xrightarrow{d_1} & X_2 & \xrightarrow{d_2} & X_3 \\ & & \uparrow m'_{-1} & & & & & & \\ & & X''_{-1} & & & & & & \end{array} \quad (42)$$

The generator sets are given by

$$\begin{aligned} X_{-1} &:= \mathcal{C}_2, & X_0 &:= \Delta_2, \\ X_1 &:= \Delta_0, & X_2 &:= \Delta_2, \\ X_3 &:= \mathcal{C}_2, \\ X''_{-1} &:= \{0\}, & X'_3 &:= \{0\}. \end{aligned} \quad (43)$$

The maps are defined by

$$\begin{aligned} (d_{-1})_{f\kappa} &:= \delta_{\mathbf{c}_f \in \kappa}, & d_0 &:= \partial_{02}, \\ d_1 &:= \partial_{20}, & (d_2)_{\kappa f} &:= \delta_{\mathbf{c}_f \in \kappa}, \\ (m'_{-1})_{\kappa 0} &:= 1 \forall \kappa, & (m_3)_{0\kappa} &:= 1 \forall \kappa. \end{aligned} \quad (44)$$

The full sequence of maps above define a length-6 chain complex. However, the middle length-2 chain complex consisting of d_0 and d_1 is most important, and only this part is locally exact. The boundary maps d_0 and d_1 are illustrated in Fig. 13.

The maps m'_{-1} and m_3 are somewhat trivial, and we could incorporate them by reducing the 3-generator sets

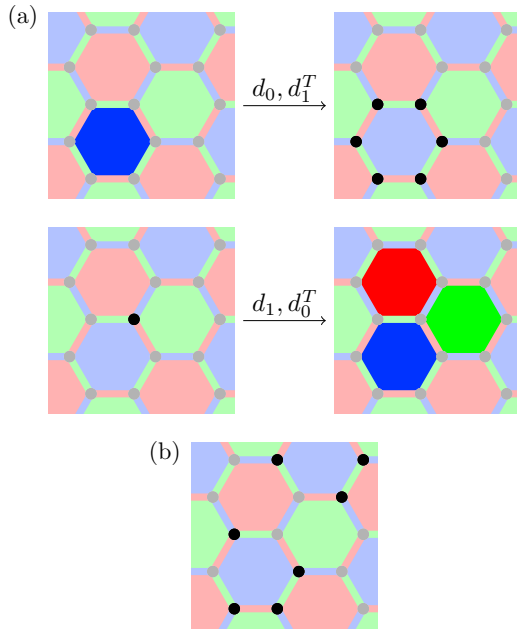


FIG. 13. Illustration of the (co)boundary maps of 2D color cohomology. (a) Boundary maps (as well as their transpose) acting on individual generators, that is, faces and vertices. In other words, each picture on the right represents a row or a column of the boundary matrix. (b) Example for a 1-cocycle. Note that 1-cycles are the same as 1-cocycles since $d_0 = d_1^T$.

X_{-1} and X_3 to 2-generator sets. Then, schematically, the chain complex would look like

$$2 \rightarrow \Delta_2 \rightarrow \Delta_0 \rightarrow \Delta_2 \rightarrow 2, \quad (45)$$

where “2” denotes a 2-element generator set. The only reason why we do not do this is that it would require us to break the permutation symmetry between the colors. d_{-1} and d_2 can be understood as global conservation laws, similar to the extension of cellular cohomology in Eq. (41).

Equation (29) for $i = 0$ becomes

$$\sum_{p \in \Delta_0} \partial_{f_1 p} \partial_{p f_0} = 0 \quad \forall f_0, f_1 \in \Delta_2. \quad (46)$$

The left-hand side is the mod 2 number of vertices p that are adjacent to both faces f_0 and f_1 . Since the 2-cellulation is trivalent, any two distinct faces f_0 and f_1 either share 0 or $2 = 0 \bmod 2$ common vertices, so the equation holds. If $f_0 = f_1$, the equation still holds since 3-colorability implies that all faces of a 2-colex have an even number of vertices. So we note that even though the 3-coloring does not show up in the definition of d_0 and d_1 in Eqs. (43) and (44), it is important to ensure that this definition forms a valid cohomology theory.

Note that 2D color cohomology is equal to the length-2 chain complex underlying the 2D color code: $X_0 = \Delta_2$,

$X_1 = \Delta_0$, and $X_2 = \Delta_2$ correspond to X -stabilizer generators, qubits, and Z -stabilizer generators, respectively. The X check matrix is given by d_0 , and the Z check matrix by d_1^T . The ground states are superpositions of color 1-cocycles with equal amplitude within each cohomology class. X_{-1} and X_3 are global redundancies between the X and Z stabilizer generators, respectively, and X_{-1}' and X_3' are redundancies among these redundancies.

2. 3D

After introducing 2D color cohomology, we are now ready to define 3D color cohomology, which is a locally exact length-3 chain complex defined on any 3-colex. Again, it is instructive to define a larger structure of generator sets and homomorphisms between them as follows:

$$\begin{array}{ccccccc} & & & & X_3' & \xrightarrow{d_3'} & X_4' \\ & & & & \uparrow m_3 & & \uparrow m_4 \\ X_{-1} & \xrightarrow{d_{-1}} & X_0 & \xrightarrow{d_0} & X_1 & \xrightarrow{d_1} & X_2 & \xrightarrow{d_2} & X_3 & \xrightarrow{d_3} & X_4 \\ & \uparrow m_{-1}' & & & & & & & & & \\ & X_{-1}'' & & & & & & & & & \\ & \uparrow m_{-1}'' & & & & & & & & & \\ & X_{-1}''' & & & & & & & & & \end{array} \quad (47)$$

The generator sets are given by

$$\begin{aligned} X_{-1} &:= \mathcal{C}_2, & X_0 &:= \Delta_3, \\ X_1 &:= \Delta_0, & X_2 &:= \Delta_2, \\ X_3 &:= \bigsqcup_{c \in \mathcal{C}_1} \Delta_3^c \times \bar{c}, & X_4 &:= \mathcal{C}_1, \\ X_{-1}''' &:= \{0\}, & X_{-1}'' &:= \mathcal{C}_1, \\ X_3' &:= \Delta_3, & X_4' &:= \{0\}. \end{aligned} \quad (48)$$

Here, $\sqcup$ denotes disjoint union, and we recall that $\bar{c}$ denotes the 3-element set of colors that are not c . That is, a 3-chain consists of three $\mathbb{Z}_2$ -variables at every c -colored volume, each labeled by one of the three other colors. The homomorphisms are given as follows:

$$\begin{aligned} (d_{-1})_{v\kappa} &:= \delta_{\mathbf{c}_v \in \kappa}, & d_0 &:= \partial_{03}, \\ d_1 &:= \partial_{20}, & (d_2)_{(v,c)f} &:= \partial_{vf} \delta_{c \notin \mathbf{c}_f}, \\ (d_3)_{c', (v,c)} &:= \partial_{c=c'}, & (d_3')_{0v} &:= 1 \forall v, \\ (m_{-1}'')_{c0} &:= 1 \forall c, & (m_{-1}')_{\kappa c} &:= \delta_{c \notin \kappa}, \\ (m_3)_{v', (v,c)} &:= \delta_{v'=v}, & (m_4)_{0c} &:= 1 \forall c. \end{aligned} \quad (49)$$

That is, d_{-1} maps a color pair κ to all volumes whose color is in κ . d_0 maps a volume to all of its vertices, and d_1 maps a vertex to all adjacent faces. d_2^T maps a volume v and a color $c \neq \mathbf{c}_v$ to all faces of v whose color does not contain c . For example, if v has color $\mathbf{c}_v = r$, then $d_2^T(v, g)$ consists

of all rb and ry faces of v . d_3^T maps a color c' to all pairs (v, c') (one for each volume v with $\mathbf{c}_v \neq c'$). $(d'_3)^T$ maps the unique generator 0 to all volumes. m_3^T maps a volume v' to all three pairs (v', c) (one for each color c with $c \neq \mathbf{c}_{v'}$). m_4^T maps the unique generator 0 to all four colors $c \in \mathcal{C}_1$. m'_{-1} maps a color c to the three color pairs κ that do not contain c . m''_{-1} maps the unique generator 0 to all four colors.

Each horizontal and each vertical line in Eq. (47) defines a chain complex, and each square commutes. This holds also for squares obtained after filling gaps with trivial 1-elements sets $\{0\}$ and zero homomorphisms. Explicitly, we have the following relations:

$$\begin{aligned} m'_{-1}m''_{-1} &= 0, & d_{-1}m'_{-1} &= 0, \\ m_3d_2 &= 0, & m_4d_3 - d'_3m_3 &= 0. \end{aligned} \quad (50)$$

The middle length-3 chain complex consisting of d_0 , d_1 , and d_2 is the most important. Only this part fulfills the local exactness property that every locally supported (co-)cycle is a (co-)boundary. We illustrate the coboundary maps d_0 , d_1 , and d_2 in Fig. 14.

Note that m_3 acts within each volume separately and thus does not couple different unit cells of the 3-colex. The

same holds trivially for m'_{-1} , m''_{-1} , and m_4 , since they not related to the 3-colex at all. So, we can use these maps to cancel some of the generators in X_{-1} , X_3 , and X_4 , while still maintaining a cell complex that is local on the lattice. In particular, using m'_{-1} and m''_{-1} , we can reduce X_{-1} from six color pairs to three generators, using m_3 , we can reduce X_3 from three to two generators per volume, and using m_4 , we can reduce X_4 from four colors to three generators. After this, we obtain a single length-5 chain complex, which schematically looks like

$$3 \rightarrow \Delta_3 \rightarrow \Delta_0 \rightarrow \Delta_2 \rightarrow 2 \times \Delta_3 \rightarrow 3, \quad (51)$$

where “2” or “3” denotes a 2-element or 3-element set. The only reason why we have not performed this reduction is that it would require us to break the symmetry under color permutations. Note that while the middle three boundary maps define circuit and path integral of our protocol as well as the decoding graphs for X and Z errors, the outer two boundary maps define global parity constraints of the syndrome and are thus important for constructing decoding algorithms based on matching decoders.

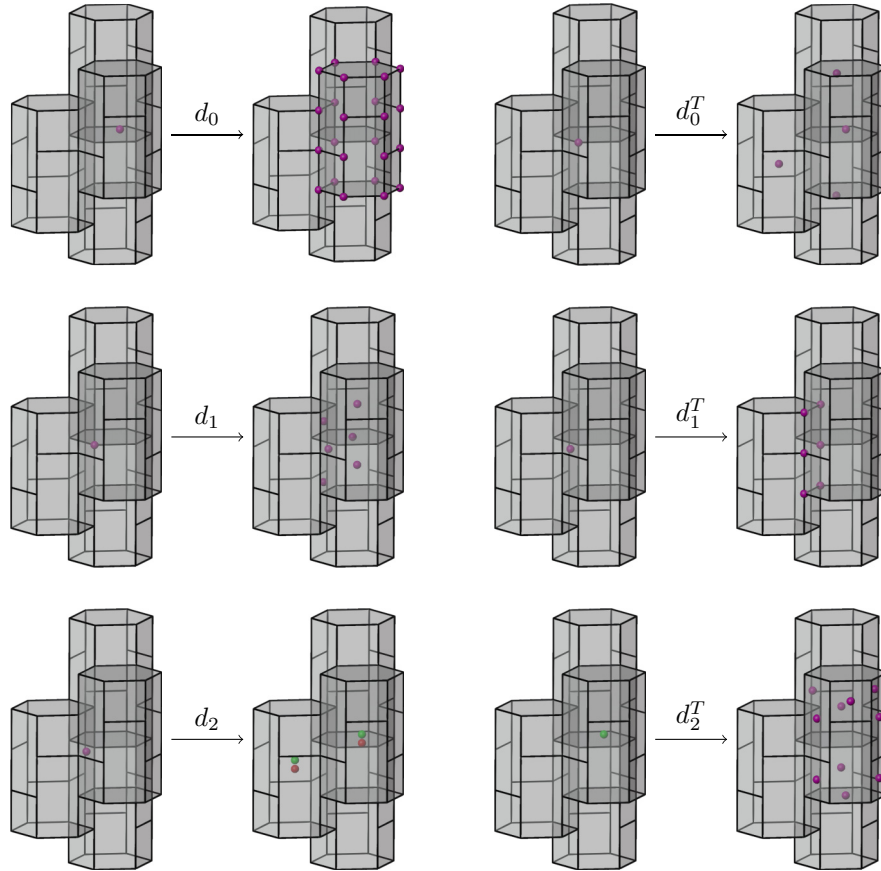


FIG. 14. Boundary maps d_0 , d_1 , and d_2 , as well as their transposes, acting on a single generator. Each generator is indicated by placing a little purple ball at the center of the vertex, volume, or face. There are three X_3 generators inside each volume v corresponding the three colors $c \neq \mathbf{c}_v$, which we distinguish by coloring the ball with color c .

Let us briefly verify the chain complex axiom in Eq. (29) for some cases. For $i = 0$, we get

$$\sum_{p \in \Delta_0} \partial_{fp} \partial_{pv} = 0 \quad \forall f \in \Delta_2, v \in \Delta_3. \quad (52)$$

The left-hand side is the mod 2 number of vertices p adjacent to both the face f and the volume v . Due to the 4-coloring, v and f are either disjoint, or they share an edge, or $f \subset v$, and in all of these cases the number of vertices p they share is divisible by 2. For $i = 1$, we have

$$\sum_{f \in \Delta_2} \delta_{c \notin \mathbf{c}_f} \partial_{vf} \partial_{fp} = 0 \quad \forall v \in \Delta_3, c \in \overline{\mathbf{c}_v}, p \in \Delta_0. \quad (53)$$

The left-hand side is the mod 2 number of faces f that are adjacent to both the vertex p and the volume v , and whose color $\mathbf{c}_f$ does not contain c . If $p \not\subset v$, then there are no such faces, and the equation holds. If $p \subset v$, they share three adjacent faces but only $2 = 0 \bmod 2$ have the right color (namely $\mathbf{c}_v$ together with either of the two colors that are not c), so the equation holds. For $i = 2$, we have

$$\sum_{v \in \Delta_3, c \in \overline{\mathbf{c}_v}} \partial_{v'v} \delta_{c \notin \mathbf{c}_f} \partial_{vf} = \sum_{c \in \mathbf{c}_v: c \notin \mathbf{c}_f} \partial_{v'f} = 0 \quad (54)$$

$$\forall f \in \Delta_2, v' \in \Delta_3.$$

The expression in the middle is 0 immediately if $f \not\subset v'$. If $f \subset v'$, it is the number of colors which are not $\mathbf{c}_{v'}$ and not in $\mathbf{c}_f$, and there are always $2 = 0 \bmod 2$ such colors.

Note that 3D color cohomology is the cohomology theory underlying the 3D color code. That is, X_0 , X_1 , and X_2 correspond to the X stabilizer generators, qubits, and Z stabilizer generators, respectively. d_0 and d_1^T are the X and Z check matrices respectively. X_3 corresponds to the *meta-checks* of the Z -stabilizer (or X -error) syndrome, which makes the 3D color code single-shot decodable, and d_2^T is the associated meta-check matrix.

Similarly, color cohomology is a convenient language to describe twisted color circuits, such as the one in Sec. II. For example, the configuration of Z measurement results can be mapped to a color 2-cochain, and the configuration of X results to a color 1-cycle. Therefore, color cohomology turns out to be the natural language to formulate the decoding of twisted color circuits, as we do in Sec. V.

C. Color cohomology: Equivalence to cellular cohomology

In this section, we show that color cohomology is equivalent to three copies of cellular cohomology (or in other words, to cellular cohomology with coefficient group $\mathbb{Z}_2^3$). Technically, we show that there is a chain map M from color to cellular cohomology, as defined around Eq. (36). Importantly, all M_i are *local* meaning that the matrix

entries are only nonzero for neighboring pairs of cells, and the chain map corresponds to an isomorphism (not just a homomorphism) of cohomology groups. A chain map very similar to M was used in Ref. [30] to show the equivalence between the 3 + 1D color code and three copies of the 3 + 1D toric code.

1. 2D

As usual, we will start with the analogous 2D case, and give a chain map M_i from 2D color cohomology to $\mathbb{Z}_2^3$ -valued (not $\mathbb{Z}_2$ -valued) cellular cohomology. The 2-cellulation on which the cellular i -chains are defined is just the 2-colex with the coloring removed. More precisely, we define one map M_i^c for every $c \in \{r, g, b\}$ from color i -chains to $\mathbb{Z}_2$ -valued cellular i -chains, such that M_i can be built from taking any two of the three M_i^c , while the third M_i^c is redundant. The M_i^c are given by

$$\begin{aligned} (M_{-1}^c)_{0\kappa} &:= \delta_{c \in \kappa}, & (M_0^c)_{pf} &:= \delta_{\mathbf{c}_f = c} \partial_{pf}, \\ (M_1^c)_{ep} &:= \delta_{\mathbf{c}_e = \bar{c}} \partial_{ep}, & (M_2^c)_{f'f} &:= \delta_{\mathbf{c}_f \neq c} \partial_{f'f}, \\ (M_3^c)_{0\kappa} &:= \delta_{c \in \kappa}, \end{aligned} \quad (55)$$

and illustrated in Fig. 15.

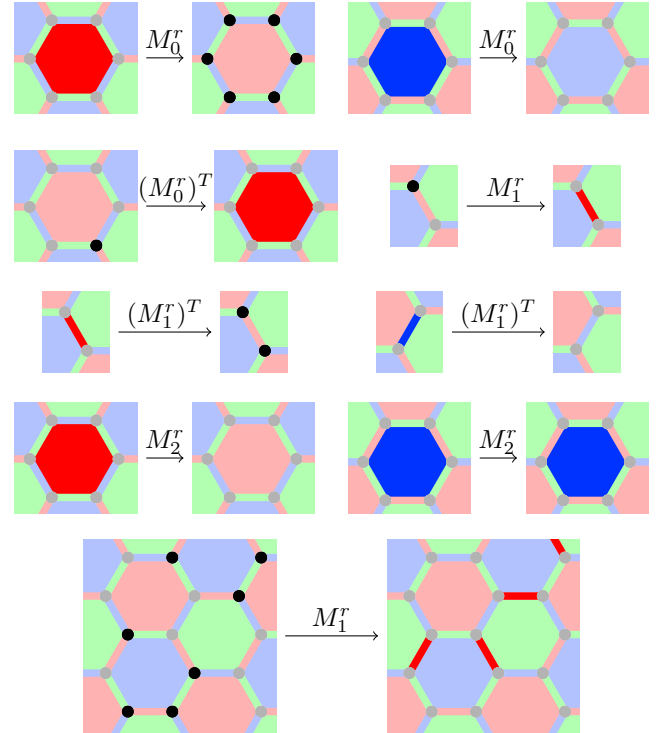


FIG. 15. Action of M_i^r or $(M_i^r)^T$ on color chains formed by a single generator. Note that $(M_2^r)^T = (M_2^r)$. Permuting colors yields analogous illustrations for M_i^g and M_i^b .

Let us now check that Eq. (36) indeed holds for some cases. For $i = 0$, we have

$$\sum_{p \in \Delta_0} \partial_{ep} \delta_{\mathbf{c}_f=c} \partial_{pf} = \sum_{p \in \Delta_0} \delta_{\mathbf{c}_e=\bar{c}} \partial_{ep} \partial_{pf}. \quad (56)$$

We confirm this by going through the different cases. If $\mathbf{c}_f \neq c$ and $\mathbf{c}_e \neq \bar{c}$, both sides are immediately zero. If $\mathbf{c}_f \neq c$ and $\mathbf{c}_e = \bar{c}$, the left-hand side is zero immediately. The right-hand side is zero as well, since any $\bar{c}$ -colored edge e either shares no vertex with a non- c -colored face f , or $e \subset f$ and they share two vertices. If $\mathbf{c}_f = c$ and $\mathbf{c}_e \neq \bar{c}$, the right-hand side is zero immediately. The left-hand side is zero as well, since any non- $\bar{c}$ -colored edge e either shares no vertex with a c -colored face, or $e \subset f$ and they share two vertices. Finally, if $\mathbf{c}_f = c$ and $\mathbf{c}_e = \bar{c}$, both sides are immediately identical. For $i = 1$, we have

$$\begin{aligned} \sum_{e \in \Delta_1} \partial_{fe} \delta_{\mathbf{c}_e=\bar{c}} \partial_{ep} &= \sum_{f' \in \Delta_2} \delta_{\mathbf{c}_f \neq c} \partial_{ff'} \partial_{f'p} \\ &= \delta_{\mathbf{c}_f \neq c} \partial_{fp}. \end{aligned} \quad (57)$$

We confirm this equation again by going through the different cases. If $\mathbf{c}_f = c$, then the right-hand side is immediately zero. The left-hand side is zero as well since there are no edges of color $\bar{c}$ in the boundary of the face f . If $\mathbf{c}_f \neq c$, then both sides are 1 if $p \in f$, and both sides are zero if $p \notin f$.

Finally, we note that $\bar{M} := M^T$ defines a chain map from transposed cellular cohomology to color cohomology. This is because color cohomology is symmetric under transposition, $d_i = d_{1-i}^T$.

2. 3D

Having discussed the equivalence between color and cellular cohomology in 2D, the generalization to 3D is straightforward: We start by defining one chain map M^c for every $c \in \mathcal{C}$ from color to $\mathbb{Z}_2$ -valued cellular cohomology, such that M can be constructed from any triple, while the fourth M^c is redundant. The M_i^c are given by

$$\begin{aligned} (M_{-1}^c)_{0\kappa} &:= \delta_{c \in \kappa}, & (M_0^c)_{pv} &:= \delta_{\mathbf{c}_v=c} \partial_{pv}, \\ (M_1^c)_{ep} &:= \delta_{\mathbf{c}_e=\bar{c}} \partial_{ep}, & (M_2^c)_{f'f} &:= \delta_{c \notin \mathbf{c}_f} \partial_{f'f}, \\ (M_3^c)_{v', (v, c')} &:= \delta_{c=c'} \partial_{v'v}, & (M_4^c)_{0c'} &:= \delta_{c=c'}. \end{aligned} \quad (58)$$

Let us verify Eq. (36). For $i = -1$, we have

$$\delta_{c \in \kappa} = \sum_{v \in \Delta_3} \delta_{\mathbf{c}_v=c} \partial_{pv} \delta_{\mathbf{c}_v \in \kappa} = \delta_{c \in \kappa} \sum_{v \in \Delta_3} \delta_{\mathbf{c}_v=c} \partial_{pv}. \quad (59)$$

The equation holds because the sum on the right is the number of c -colored volumes adjacent to the vertex p ,

which is always 1. For $i = 0$, we have

$$\sum_{p \in \Delta_0} \partial_{ep} \delta_{\mathbf{c}_v=c} \partial_{pv} = \sum_{p \in \Delta_0} \delta_{\mathbf{c}_e=\bar{c}} \partial_{ep} \partial_{pv}. \quad (60)$$

If $\mathbf{c}_v \neq c$ and $\mathbf{c}_e \neq \bar{c}$, both sides are zero immediately. If $\mathbf{c}_v = c$ and $\mathbf{c}_e \neq \bar{c}$, the left-hand side is zero immediately. The right-hand side is zero as well because e shares either no vertex with the c -colored volume v , or $e \subset v$ and they share two vertices. The case $\mathbf{c}_v \neq c$ and $\mathbf{c}_e = \bar{c}$ is the same up to color permutation. If $\mathbf{c}_v = c$ and $\mathbf{c}_e = \bar{c}$, both sides are identical immediately. For $i = 1$, we have

$$\sum_{e \in \Delta_1} \partial_{fe} \delta_{\mathbf{c}_e=\bar{c}} \partial_{ep} = \sum_{f' \in \Delta_2} \delta_{c \notin \mathbf{c}_f} \partial_{ff'} \partial_{f'p} = \delta_{c \notin \mathbf{c}_f} \partial_{fp}. \quad (61)$$

If $c \in \mathbf{c}_f$, the right-hand side is zero immediately. The left-hand side is zero as well since the $\mathbf{c}_f$ -colored face f has no $\bar{c}$ -colored edges. If $c \notin \mathbf{c}_f$, then both sides are 1 if $p \in f$, and both sides are zero if $p \notin f$. For $i = 2$, we have

$$\begin{aligned} \partial_{vf} \delta_{c \notin \mathbf{c}_f} &= \sum_{f' \in \Delta_2} \partial_{vf'} \delta_{c \notin \mathbf{c}_f} \partial_{f'f} \\ &= \sum_{v' \in \Delta_3, c' \in \mathcal{C}} \delta_{c=c'} \partial_{vv'} \delta_{c' \notin \mathbf{c}_f} \partial_{v'f} = \delta_{c \notin \mathbf{c}_f} \partial_{vf}, \end{aligned} \quad (62)$$

and both sides are identical immediately. For $i = 3$, we have

$$\sum_{v' \in \Delta_3} \delta_{c=c'} \partial_{vv'} = \sum_{c' \in \mathcal{C}} \delta_{c=c'} \delta_{c'=c'}, \quad (63)$$

which holds immediately.

We can also define a reverse chain map $\bar{M}$ from ($\mathbb{Z}_2^3$ -valued) cellular cohomology to color cohomology. More precisely, we can define one chain map $\bar{M}^\kappa$ for each color pair $\kappa \in \mathcal{C}_2$. Any triple of color pairs κ that cover all four colors can be combined into $\bar{M}$, while the other three chain maps are redundant. The 3-cellulation is obtained from the 3-colex as follows: The cellulation vertices are at the centers of the 3-colex volumes v whose color is in κ , $\mathbf{c}_v \in \kappa$. The cellulation edges are dual to the 3-colex faces f of color $\mathbf{c}_f = \kappa$, connecting the vertices at the centers of the volumes adjacent to f . The cellulation faces are the 3-colex faces of opposite color $\bar{\kappa}$, which are enlarged such that their vertices are at the centers of the adjacent $\mathbf{c}_v \in \kappa$ volumes. The cellulation volumes are the 3-colex volumes v of color $\mathbf{c}_v \notin \kappa$, again enlarged. So for each cellulation cell c , there is an associated 3-colex cell, which we denote by $x(c)$.

Now, $\overline{M}^\kappa$ is defined as follows:

$$\begin{aligned} (\overline{M}_{-1}^\kappa)_{\kappa'0} &:= \delta_{\kappa=\kappa'}, & (\overline{M}_0^\kappa)_{vp} &:= \delta_{v=x(p)}, \\ (\overline{M}_1^\kappa)_{pe} &:= \partial_{p,x(e)}, & (\overline{M}_2^\kappa)_{f'f} &:= \delta_{f'=x(f)}, \\ (\overline{M}_3^\kappa)_{(v',c)v} &:= \delta_{v'=x(v)}\delta_{c\in\kappa}, & (\overline{M}_4^\kappa)_{c0} &:= \delta_{c\in\kappa}. \end{aligned} \quad (64)$$

Physically, the equivalence between the two cohomology theories implies that the codes or circuits built from both can be understood as different microscopic realizations of the same topological phase. For example, mappings M and $\overline{M}$ define local isometries between the 3D color code (which is based on 3D color cohomology) and three copies of the toric code (each based on 3D cellular cohomology), which show that they are in the same topological phase. We will later use the cohomology equivalence in Sec. IV C to show that the color path integral represents a particular non-Abelian topological phase.

IV. THE COLOR PATH INTEGRAL AS A NON-ABELIAN PHASE

In Sec. II, we motivated the color path integral in terms of a $3 + 0D$ protocol based on the 3D color code. In this section, we will give a more natural and direct physical interpretation of the path integral, namely as a spacetime fixed-point model for a $2 + 1D$ non-Abelian topological phase. To this end, we introduce a state-sum language for general path integrals in Sec. IV A, after which the color path integral can be conveniently described in terms of color cohomology in Sec. IV B. Using the cohomology equivalence from Sec. III C, we then show in Sec. IV C that the color path integral is equivalent to a known model for a non-Abelian phase, namely the Dijkgraaf-Witten state sum for the type-III group 3-cocycle on $\mathbb{Z}_2^3$. In Sec. IV D, we equip the path integral with flux and charge defect configurations, which is necessary to turn its tensor network representation into a fault-tolerant circuit. Finally, in Sec. IV E, we discuss how to equip the path integral with boundaries.

A. Discrete path integrals

Before we get to the color path integral, we introduce path integrals more generally, and explain how they describe topological phases. We introduce their *fixed-point* property, which implies that isolated errors do not affect the logical action of the circuit. One way to define a path integral is as a tensor network with tensors assigned to certain places of a spacetime lattice, and index contractions between nearby tensors. We have defined the color path integral in this way in Sec. II B. For this section it will be more convenient to use an alternative, but equivalent, language to describe path integrals, which in the context of topological phases is known as *state sums* [58–64].

1. State-sum path integrals

A *state-sum path integral* associates variables to certain places of the spacetime lattice, such as all the vertices, all the edges, or all the corners of volumes. The variables take values in some finite set, such as $\{0, 1\}$. Evaluating the path “integral” means summing over all spacetime configurations $\vec{c}$ of the variables. The summand is a product of weights ω_x , which are also associated to certain places x of the spacetime lattice. These weights are complex numbers ω that depend on the configuration of the variables within some constant-size neighborhood. So, schematically, a state-sum path integral can be written as

$$Z = \sum_{\text{confs. } \vec{c}} \prod_{\text{places } x} \omega_x(\vec{c}|_{\text{near } x}). \quad (65)$$

The path integrals we consider are uniform, that is, the distribution of variables and weights only depends on what the lattice looks like locally. In particular, on a translation-invariant spacetime lattice the path integral is translation invariant as well, and specified by a single unit cell. In physics, state sums are commonly used in statistical mechanics, where they are referred to as “partition functions.” For example, the classical 2D Ising model at inverse temperature $\beta \in \mathbb{R}^+$ is a state sum on a 2D square lattice, with one $\{\pm 1\}$ -valued variable at each vertex, and one “Boltzmann” weight $\omega_e(c) = e^{-\beta J c(e_0)c(e_1)}$ at each edge e with endpoints e_0 and e_1 .

Let us briefly point out how state sums are equivalent to tensor networks. A tensor network is evaluated by summing over all configurations of contracted index pairs. The summand is the product over all tensor entries for the corresponding index configuration. So the contracted index pairs in the tensor network are equivalent to the variables in the state sum, and the tensors themselves are equivalent to the weights. State sums may seem more general, as an index contraction can only involve two indices, whereas a variable can enter into an arbitrary number of weights. However, we can define the contraction over a larger number of indices by connecting them all to one common *Kronecker δ -tensor*, which is the generalization of the Z -tensor in Eq. (5) to arbitrary bond dimensions.

Evaluating the path integral on a closed spacetime lattice with periodic boundary conditions yields a number Z as in Eq. (65). This number is not particularly relevant for our purposes. What is relevant instead is the evaluation of the path integral on a spacetime lattice with *state boundaries*. In this case, we freeze a configuration $\vec{c}_{\text{boundary}}$ of state-sum variables within a constant distance from the boundary, and sum over all other variables in the interior. This way, we get a vector of numbers $Z(\vec{c}_{\text{boundary}})$, instead of a single number, which can be interpreted as the state vector of a state defined on degrees of freedom living on the boundary [65]. For a tensor-network path integral, this corresponds

to cutting the indices near the boundary and leaving them open.

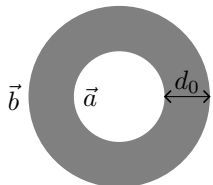
2. Physical interpretation of path integrals

The path integrals that we consider are defined in a 3D spacetime and describe the zero-temperature physics of a 2D quantum model, in contrast to thermal physics for the Ising-model path integral illustrated above. So our path integrals can be viewed as an alternative language to 2D lattice Hamiltonians [66]. The state $Z(\vec{c}_{\text{boundary}})$ then plays a role analogous to the ground state of a Hamiltonian model.

In particular, we consider path integrals that represent nontrivial 2D topological phases [67]. While lattice Hamiltonians are more commonly used to describe topological phases in condensed-matter physics, the path-integral language makes it easier to construct syndrome-extraction circuits for these topological phases. More precisely, our path integrals are directly equal to a syndrome-extraction circuit that is postselected onto a particular (+1) outcome. In this context, $Z(\vec{c}_{\text{boundary}})$ corresponds to a code state of an underlying error-correcting code. This code will, in general, be microscopically different depending on the chosen state boundary but is guaranteed to inherit necessary properties for error correction from the path integral.

3. Fixed-point path integrals

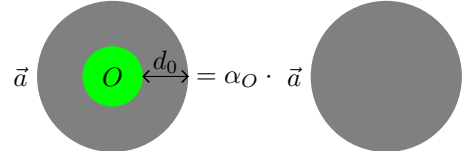
We use a strongly restricted subclass of path integrals, which are particularly useful for constructing error-correcting circuits. Namely, we use ones that represent the desired topological phase in a particularly pure manner, at zero-correlation length. We call them *fixed-point* path integrals [63,64] and define them in the following. Consider evaluating the path integral on a 3-ball with a smaller 3-ball removed from its center. We consider both interior and exterior boundary 2-spheres as state boundaries and assume that they are separated by at least some constant lattice distance d_0 [68]. Evaluating the path integral on this lattice yields a state $|T\rangle$ on the boundary, which can be considered a bipartite state where one part corresponds to the interior boundary and the other to the exterior boundary. The defining property of a fixed-point path integral is that $|T\rangle$ is a product state, $|T\rangle = |T_{\text{in}}\rangle \otimes |T_{\text{out}}\rangle$, as illustrated by following 2D toy picture,



$$\vec{b} \quad \vec{a} \quad \xleftrightarrow{d_0} \quad \rightarrow \quad \langle \vec{a}, \vec{b} | T \rangle = \langle \vec{a} | T_{\text{in}} \rangle \langle \vec{b} | T_{\text{out}} \rangle . \quad (66)$$

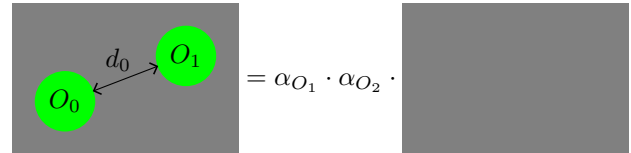
As a consequence, these path integrals have *zero-correlation length* in spacetime: If we insert a observable

O (or alter the path integral in any other way) inside some simply connected region, this only changes the path integral by a global prefactor α_O ,



$$\vec{a} \quad \xleftrightarrow{d_0} \quad = \alpha_O \cdot \vec{a} . \quad (67)$$

As a consequence, all correlators [69] between any two observables O_0 and O_1 disappear if they are separated by at least d_0 ,



$$= \alpha_{O_1} \cdot \alpha_{O_2} . \quad (68)$$

The fixed-point property in Eq. (66) is the underlying reason why the circuits that we construct from the path integral have a fault-tolerant threshold: If we view the inserted operator O as an error, then this means that isolated errors are irrelevant.

Note that typical examples of fixed-point path integrals have nice algebraic properties, such as an exact notion of topological invariance. That is, we can freely deform the underlying spacetime lattice by locally applying *rewrite rules*, which are equations between small tensor networks, such as the ZX rules [27,63,64]. Similarly, if two fixed-point path integrals are in the same topological phase, then they are typically related via local rewrite rules as well.

4. Boundaries and other defects

The state boundaries discussed above are to be contrasted with *physical boundaries*, which we have discussed in Sec. IID and will revisit in Sec. IVE. The main difference is that for physical boundaries we do sum over all boundary variables instead of keeping them fixed. To define a physical boundary, we do not only remove the variables outside the boundary and the weights depending on them, but we may also introduce new variables and weights on the boundary [70]. In the tensor-network language, a physical boundary is a way to close off the open indices obtained from merely cutting the tensor network along some surface (corresponding to a state boundary). Similarly, we can consider *domain walls*, which are 2D surfaces separating two different 3D regions where the path integral is defined differently, one on each side.

Boundaries and domain walls for the path integral are needed to construct a protocol with a globally planar layout, which implements a non-Clifford logic gate. Equation (1) shows an example of a configuration of

boundaries and domain walls that corresponds to a particular logic operation. Like the topological phases for the bulk, the different physical boundaries (or domain walls) can be classified into different *boundary phases*. The logical action of the resulting circuit only depends on the topological phases of the bulk, boundaries, and domain walls in the spacetime diagram. The boundaries and domain walls we consider are such that they obey the fixed-point property in Eq. (66) also on the boundary or domain wall.

In a conceptually similar way, we can equip a path integral with *cohomological defects* as in Sec. IV D. That is, in addition to the variables, we let the weights depend on some cohomological chain (see Sec. III A), which we do not sum over. Such cohomological chains represent the configurations of measurement outcomes of the circuit.

B. The color path integral via color cohomology

In this section, we reintroduce the color path integral from Sec. II B, formulated as a state sum using color cohomology. We first introduce the 2D analog of the 3D color path integral, both for pedagogical reasons and because we will use it later to show an important property (“gauge invariance”) of the 3D color path integral.

1. 2D

The *2D untwisted color path integral* is defined on any 2-colex as

$$Z_{\text{untw}} = \sum_{A \in \mathbb{Z}_2^{\Delta_0}} \prod_{f \in \Delta_2} \delta_{(dA)_f=0} = \sum_{A \in \mathbb{Z}_2^{\Delta_0}: dA=0} 1. \quad (69)$$

This is a state sum with one $\mathbb{Z}_2$ -valued variable at every vertex, and one weight at every face f enforcing that $(dA)_f = 0$, where $d = d_1$ is the coboundary of 2D color cohomology, see Sec. III B. As shown, this weight can be suppressed by summing only over 2D color 1-cocycles A . The *2D (twisted) color path integral* includes an additional weight at every vertex $p \in \Delta_0$,

$$Z = \sum_{A \in \mathbb{Z}_2^{\Delta_0}: dA=0} \prod_{p \in \Delta_0} e^{\text{sgn}_p \frac{2\pi i}{4} \overline{A_p}}. \quad (70)$$

Here, $\text{sgn}_p = +1$ if cycling through the adjacent faces in the ordering $r - g - b$ is clockwise, and $\text{sgn}_p = -1$ otherwise. Also, $\overline{A_i}$ means that we consider A_i as an element of $\mathbb{Z}$ instead of $\mathbb{Z}_2$. The overall weight is the exponential of the following *action*:

$$S[A] = \frac{1}{4} \sum_{p \in \Delta_0} \text{sgn}_p \overline{A_p} \mod 1 \in \mathbb{R}/\mathbb{Z}. \quad (71)$$

The central property of the action is its *gauge invariance*,

$$A' := A + d\alpha \Rightarrow S[A'] = S[A], \quad (72)$$

for any 2D color 1-cocycle A and color 0-cochain α . Gauge invariance directly implies that S only depends on the cohomology class of the color 1-cocycle A . It suffices to show gauge invariance for the generating 0-chains $\alpha = f$ formed by a single face f . We indeed find

$$\begin{aligned} S[A'] - S[A] &= \frac{1}{4} \sum_{p \in \Delta_0} \text{sgn}_p (\overline{A_p + (df)_p} - \overline{A_p}) \\ &\stackrel{(74)}{=} \frac{1}{4} \sum_{p \in \Delta_0} \text{sgn}_p \left(\overline{(df)_p} + 2\overline{A_p (df)_p} \right) \\ &\stackrel{*}{=} \frac{1}{4} \sum_{p \in f} \text{sgn}_p + \frac{1}{2} \sum_{p \in f} \overline{A_p} \stackrel{**}{=} \frac{1}{2} \overline{(dA)_f} \\ &= 0. \end{aligned} \quad (73)$$

For the middle equality, we have used that

$$\overline{a + b} - \overline{a} = \overline{b} - 2\overline{a} \overline{b} \quad (74)$$

for $a, b \in \mathbb{Z}_2$. For (*) we have used that the summand is only supported on the vertices $p \in f$. For (**) we have used that the first summand is zero, since the 3-colorability implies that f has an equal number of vertices $p \in f$ with $\text{sgn}_p = 1$ and $\text{sgn}_p = -1$. We have also used the definition of d_1 and that A is a color 1-cocycle. Finally, we note that the action in Eq. (71) [or better, the according weight in Eq. (70)] is analogous to the transversal S gate in the 2D color code, and the gauge invariance is analogous to the fact that the transversal gate preserves the code space.

2. 3D

After describing the 2D color path integral, the 3D generalization is rather straightforward. The *(3D) untwisted color path integral* is defined on any 3-colex as

$$Z_{\text{untw}} = \sum_{A \in \mathbb{Z}_2^{\Delta_0}} \prod_{f \in \Delta_2} \delta_{(dA)_f=0} = \sum_{A \in \mathbb{Z}_2^{\Delta_0}: dA=0} 1. \quad (75)$$

The *(3D twisted) color path integral* includes an additional weight,

$$Z = \sum_{A \in \mathbb{Z}_2^{\Delta_0}: dA=0} \prod_{p \in \Delta_0} e^{\text{sgn}_p \frac{2\pi i}{8} \overline{A_p}}, \quad (76)$$

recalling the sign sgn_p of a vertex p from the beginning of Sec. II B.

Let us briefly confirm that this is indeed the same as the color path integral introduced in Sec. II B as a ZX tensor

network. Indeed, the weight $\delta_{(dA)_f=0}$ at a face f is 1 if the variables at all the vertices of $p \in f$ sum to 0 mod 2, and 0 otherwise. So these weights correspond to the X tensors at every face in Fig. 1. The summation over the variable at every vertex corresponds to a “multi-index contraction,” which is implemented by a Z -tensor with indices connected to all adjacent face weights. The state-sum weight $e^{\text{sgn}_p (2\pi i/8) \overline{A_p}}$ at every vertex p is implemented by adding a phase of $\text{sgn}_p 2\pi/8$ to the according Z -tensor in the tensor network.

The overall weight in the path integral is the exponential of the following action:

$$S[A] = \frac{1}{8} \sum_{p \in \Delta_0} \text{sgn}_p \overline{A_p}. \quad (77)$$

The action is again gauge invariant as defined in Eq. (72) and thus only depends on the cohomology class of A . In the 3D case, gauge invariance ensures that the resulting path integral is a fixed-point path integral and obeys the zero-correlation length condition in Eq. (66). It suffices to show gauge invariance for the generating 0-cochains $\alpha = v$ consisting of a single volume $v \in \Delta_3$. We indeed find

$$\begin{aligned} S[A'] - S[A] &= \frac{1}{8} \sum_{p \in \Delta_0} \text{sgn}_p (\overline{A_p + (dv)_p} - \overline{A_p}) \\ &\stackrel{(74)}{=} \frac{1}{8} \sum_{p \in \Delta_0} \text{sgn}_p (\overline{(dv)_p} + 2\overline{A_p(dv)_p}) \\ &= \frac{1}{8} \sum_{p \in v} \text{sgn}_p + \frac{1}{4} \sum_{p \in v} \text{sgn}_p \overline{A_p} \\ &\stackrel{*}{=} S_{2D,v}[A|_v] = 0. \end{aligned} \quad (78)$$

For (*) we have used that due to the 4-colorability the number of $\text{sgn}_p = +1$ and $\text{sgn}_p = -1$ vertices of v is equal, so the first summand vanishes. We have also used that the second term is the action $S_{2D,v}$ of the 2D color path integral restricted to the boundary of v , which is a 2-cocycle when colored according to the adjacent volumes. Since (1) the color 1-cohomology group on any 2-sphere is trivial and (2) the 2D action is gauge invariant and only depends on the 1-cohomology class, the 2D action is the same as for the $A = 0$ color 1-cocycle, namely 0. Finally, we note that the action in Eq. (77) [or better, the corresponding phase factor in Eq. (76)] is the same as occurs in the transversal T gate of the 3D color code.

C. Equivalence to type-III twisted path integral

In this section, we establish that the color path integral represents a non-Abelian topological phase. We will do this by showing its equivalence to a known path integral, namely the *Dijkgraaf-Witten* state sum [61] for the group

$\mathbb{Z}_2^3$ and the type-III group 3-cocycle. The latter is known to host a non-Abelian anyon theory [71]. This equivalence is an extension of the equivalence between color and cellular cohomology discussed in Sec. III C.

1. 2D cellular path integral

As usual, we start with a simpler 2D analogy of the 3D equivalence. We show the equivalence of the 2D twisted color path integral with the following *2D twisted cellular path integral*, defined on any 2-cellulation:

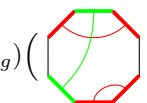
$$\begin{aligned} Z_{\text{cell}} &= \sum_{A_r, A_g \in \mathbb{Z}_2^{\Delta_1}} \prod_{f \in \Delta_2} \delta_{(d^{\text{cell}} A_r)_f=0} \delta_{(d^{\text{cell}} A_g)_f=0} \\ &\times \prod_{f \in \Delta_2} e^{2\pi i \frac{1}{2} (A_r \cup A_g)(f)}. \end{aligned} \quad (79)$$

So the path integral is the sum over all pairs of (cellular) 1-cocycles A_r and A_g , with an action

$$S[A_r, A_g] = \frac{1}{2} \sum_{f \in \Delta_2} (A_r \cup A_g)(f). \quad (80)$$

Here $\cup$ denotes the *cup product*, whose value on a face f only depends on the values of A_r and A_g on the edges of f . There are explicit formulas for the cup product on triangulations [72], cubulations [73], or general cellulations [28]. Intuitively, the cup product counts the number of intersections of the closed-loop patterns A_r and A_g . To turn this intuition into a concrete formula, we divide the edges e of the face f with $(A_r)_e = 1$ into pairs in an arbitrary way. Then, we connect the two edge centers of each edge pair with a red line. Analogously, we connect the $(A_g)_e = 1$ edge centers with green lines. $(A_r \cup A_g)(f)$ is then given by the mod 2 number of red-green line intersections inside the face. Note that this number is independent on how we pair up the $(A_r)_e = 1$ ($(A_g)_e = 1$) edges.

For example, in the 8-gon face shown below, we have 4 $(A_r)_e = 1$ edges (marked in thick red) and 2 $(A_g)_e = 1$ edges (marked in thick green). The connecting lines are shown as thin red and green lines, and there is one intersection,

$$(A_r \cup A_g) \left(\text{Diagram} \right) = 1 \bmod 2. \quad (81)$$


Thus, the cup product evaluates to 1 mod 2 and the according path-integral weight is $e^{2\pi i(1/2)1} = -1$ [74].

2. 2D equivalence

Let us now consider the 2D color path integral, which is a sum over color 1-cocycles A . Using the chain map

defined in Sec. III C, we can turn A into a pair of cellular 1-cocycles $A_r := M_1^r A$ and $A_g := M_1^g A$ on the same 2-colex without the coloring. We will now show that the actions are equivalent, i.e.,

$$\frac{1}{4} \sum_{p \in \Delta_0} \text{sgn}_p \overline{A_p} = \frac{1}{2} \sum_{f \in \Delta_2} (A_r \cup A_g)(f). \quad (82)$$

In fact, we will show the stronger statement that this equation holds locally: It holds if we restrict the right-hand side to a single b -colored face f , and the left-hand side to the vertices $p \in f$,

$$\frac{1}{4} \sum_{p \in f} \text{sgn}_p \overline{A_p} = \frac{1}{2} (M^r A \cup M^g A)(f). \quad (83)$$

Note that (1) each vertex p is part of exactly one b -colored face, and (2) $M^r A$ is only supported on the $\bar{r}$ -colored edges, and $M^g A$ only on the $\bar{g}$ -colored edges, so the right-hand side is only nonzero on the b -colored faces where rg intersections can happen. Thus summing the local equalities Eq. (83) over all b -colored faces yields Eq. (82). To show Eq. (83), we notice that both sides are 0 if the number of vertices $p \in f$ with $A_p = 1$ and $\text{sgn}_p = +1$ is equal to that with $A_p = 1$ and $\text{sgn}_p = -1$, and $\frac{1}{2}$ otherwise. For the left-hand side, this is rather obvious. For the right-hand side, we notice that every $A_p = 1$ vertex gives rise to one $(A_r)_e = 1$ edge and one $(A_g)_e = 1$ edge. If $\text{sgn}_p = +1$, then the $(A_r)_e = 1$ edge is next to the $(A_g)_e = 1$ edge in clockwise direction, otherwise their ordering is counterclockwise. Note that we start a red line from every $(A_r)_e = 1$ edge center and a green line from every $(A_g)_e = 1$ edge center. If there is one clockwise-ordered pair of $(A_r)_e = 1, (A_g)_e = 1$ edges and another counterclockwise pair, then the green and red lines connecting them do not intersect. More generally, the number of rg intersections is 0 mod 2 if the number of clockwise pairs is equal to the number of counterclockwise pairs, and 1 mod 2 otherwise.

3. 3D cellular path integral

Let us now go back to the 3D color path integral and show that it is equivalent to the following *twisted cellular path integral*:

$$\begin{aligned} Z^{\text{cell}} = & \sum_{A_r, A_g, A_b \in \mathbb{Z}_2^{\Delta_1}} \prod_{f \in \Delta_2} \delta_{(d^{\text{cell}} A_r)_f = 0} \delta_{(d^{\text{cell}} A_g)_f = 0} \\ & \times \delta_{(d^{\text{cell}} A_b)_f = 0} \prod_{v \in \Delta_3} e^{2\pi i \frac{1}{2} (A_r \cup A_g \cup A_b)(v)}. \end{aligned} \quad (84)$$

So the path integral is the sum over all triples of cellular 1-cocycles A_r, A_g , and A_b , with an action

$$S[A_r, A_g, A_b] = \frac{1}{2} \sum_{v \in \Delta_3} (A_r \cup A_g \cup A_b)(v). \quad (85)$$

Again, $\cup$ denotes is the cup product, and the value of the action on a volume v only depends on the values of A_r, A_g , and A_b on the edges of v . On a tetrahedron with vertices labeled 0, 1, 2, and 3, and edges labeled accordingly by 01, 02, 03, 12, 13, and 23, the path-integral weight is given by the type-III group 3-cocycle [71] $\omega \in Z^3(B\mathbb{Z}_2^3, U(1))$,

$$\begin{aligned} \omega & \left(((A_r)_{01}, (A_g)_{01}, (A_b)_{01}), ((A_r)_{12}, (A_g)_{12}, (A_b)_{12}), \right. \\ & \quad \left. ((A_r)_{23}, (A_g)_{23}, (A_b)_{23}) \right) \\ & = (-1)^{(A_r)_{01}(A_g)_{12}(A_b)_{23}}. \end{aligned} \quad (86)$$

Intuitively, A_r, A_g , and A_b can be pictured as closed-membrane patterns, $A_r \cup A_g$ is a closed-loop pattern formed by the intersection of A_r and A_g , and $A_r \cup A_g \cup A_b$ is the set of points where this closed-loop pattern intersects with the closed-membrane pattern A_b . In other words, the action counts the number of triple-intersections between the three closed-membrane patterns A_r, A_g , and A_b . To make this intuition into a concrete formula, we need to make some local choices. (1) We need to choose three distinct points on every edge, labeled r, g , and b . (2) For every configuration of A_r on the edges of each face, choose a way to connect the r -points on the $A_r = 1$ -edges with “ r ” lines. We make the analogous choices for A_g and A_b such that the $r - g, r - b$, and $g - b$ intersection points are always distinct. With these choices, we can compute $(A_r \cup A_g \cup A_b)(v)$ as follows: The r lines inside every face $f \subset v$ (for the corresponding A_r -configuration) together form a closed-loop pattern inside the boundary of v , and analogously for g and b . Choose a “ r ” closed-membrane pattern inside the volume v whose boundary is the r closed-loop pattern, and analogously for g and b . Now, $(A_r \cup A_g \cup A_b)(v)$ is the mod 2 number of triple-intersections of the r, g , and b membranes inside the volume v . Note that the number of triple-intersections is independent of the chosen membranes, since v is a 3-ball with trivial homology.

4. 3D equivalence

Having defined the twisted cellular path integral, let us now show that it is equivalent to the color path integral. We translate the color 1-cocycle A summed in the color path integral into three $\mathbb{Z}_2$ -valued cellular 1-cocycles $A^r := M^r A, A^g := M^g A$, and $A^b := M^b A$. Then, we show that the actions are equivalent, i.e.,

$$\frac{1}{8} \sum_{p \in \Delta_0} \text{sgn}_p \overline{A_p} = \frac{1}{2} \sum_{v \in \Delta_3} (A^r \cup A^g \cup A^b)(v). \quad (87)$$

In fact, we show the stronger condition that this equation holds locally: The equivalence also holds if we restrict to a single y -colored volume v on the right-hand side, and to all vertices of v on the left-hand side,

$$\frac{1}{8} \sum_{p \in v} \text{sgn}_p \overline{A_p} = \frac{1}{2} (M^r A \cup M^g A \cup M^b A)(v). \quad (88)$$

Note that (1) every vertex is adjacent to exactly one y -colored volume, and (2) the right-hand side is zero on every non- y -colored volume, since one of $M^r A$, $M^g A$, or $M^b A$ is not supported there, and so there are no triple- rgb intersections inside these volumes. Thus, summing Eq. (88) over all y -colored volumes yields Eq. (87). To show Eq. (88), we start by noting that it holds if $A_p = 0$ for all $p \in v$, since both sides are zero. Next, we note that the boundary of v is a 2-colex, and A defines a color 1-cocycle on this 2-colex. Since the 2D color cohomology of a 2-sphere is trivial, any color 1-cocycle A can be obtained from $A = 0$ by applying gauge transformations $A' = A + d_{2D, \text{color}} f$ for different faces $f \subset v$. Now, consider such a gauge transformation for a by -colored face $f \subset v$. Then, we study how both sides of Eq. (88) change if we change A to A' ,

$$\begin{aligned} & \frac{1}{8} \sum_{p \in v} \text{sgn}_p \overline{A'_p} - \frac{1}{8} \sum_{p \in v} \text{sgn}_p \overline{A_p} \\ & \stackrel{(74)}{=} \frac{1}{8} \sum_{p \in v} \text{sgn}_p ((df)_p + 2(df)_p A_p) \\ & = \frac{1}{8} \sum_{p \in f} \text{sgn}_p + \frac{1}{4} \sum_{p \in f} \overline{A_p} \stackrel{(83)}{=} \frac{1}{2} (A^r \cup_{2D} A^g)(f) \\ & \quad (89) \end{aligned}$$

$$\begin{aligned} & \stackrel{*}{=} \frac{1}{2} ((A')^r \cup (A')^g \cup (A')^b)(v) \\ & \quad - \frac{1}{2} (A^r \cup A^g \cup A^b)(v). \quad (90) \end{aligned}$$

We find that they change by the same amount. For the equation labeled (*), we have used that $(A')^r = A^r$ and $(A')^g = A^g$, since each $\bar{r}$ -colored or $\bar{g}$ -colored edge shares two or no vertices with the by -colored face f . $(A')^b$ differs from A^b by all edges that share one vertex with f . Thus, the closed-loop configurations on v for $(A')^b$ and A^b differ by a closed loop surrounding f . Adding a closed loop around f to A^b leads to an additional triple-membrane crossing inside v precisely if A^r and A^g have an odd number of line crossings inside f , which is measured by $(A^r \cup_{2D} A^g)(f)$. Equation (89) holds analogously if f has color yr or yg . Since any color 1-cocycle A on the boundary of v can be obtained by a sequence of gauge transformations starting with $A = 0$, Eq. (88) holds for all A .

D. Flux and charge defects

In this section, we will equip the color path integral with *flux and charge defects*. While the color path integral is equal to the twisted color circuit from Sec. II where all measurement outcomes are postselected to $+1$, the color path integral with flux and charge defects correspond to the circuit postselected onto some nontrivial configuration of outcomes. Adding flux and charge defects is therefore essential for turning the path integral into a circuit of non-post-selected measurements, as well as for discussing decoding as in Sec. V. Inserting flux and charge defects means modifying the path integral such that it depends on a *flux configuration* b , and a *charge configuration* c . b is a color 2-cochain on the 3-colex, and c is a color 1-chain.

1. Untwisted case: Definition

We start by introducing flux and charge defects in the untwisted color path integral in Eq. (75),

$$Z_{\text{untw}}[b, c] = \sum_{A \in \mathbb{Z}_2^{\Delta_0} : dA = b} \prod_{p \in \Delta_0} (-1)^{\frac{1}{2} A_p c_p}. \quad (91)$$

The flux and charge configurations modify the path integral in two ways: First, instead of summing over all color 1-cocycles A , we sum over all color 1-cochains whose boundary is the flux configuration, $dA = b$. Second, there is an additional path-integral weight of $(-1)^{A_p c_p}$ at every vertex $p \in \Delta_0$, which depends on the charge configuration [75]. Note that we recover the original path integral for $b = 0$ and $c = 0$, $Z_{\text{untw}}[0, 0] = Z_{\text{untw}}$.

2. Untwisted case: Properties

The flux and charge defects each obey two central properties, which are important for decoding and fault tolerance of the untwisted color circuits. The first property is that the flux configuration must be a color 2-cocycle, otherwise the untwisted color path integral evaluates to zero:

$$db \neq 0 \Rightarrow Z_{\text{untw}}[b, c] = 0. \quad (92)$$

The simple reason for this is that if $db \neq 0$, then there exists no color 1-cochain A such that $dA = b$ due to Eq. (29), so the summation in Eq. (91) is empty. In fact, this equation holds locally, meaning that if $db \neq 0$ at some location in the 3-colex, then the path integral evaluated on a small neighborhood of this location is zero. Specifically, if $(db)_{(v,c)} \neq 0$ for some volume $v \in \Delta_3$ and color $c \neq \mathbf{c}_v$, then the path integral only on v is zero, where all A_p with $p \in v$ are state-boundary variables. The analogous property for the charges is that the charge configuration must

be a color 1-cycle,

$$d^T c \neq 0 \Rightarrow Z_{\text{untw}}[b, c] = 0. \quad (93)$$

Again, this equation holds locally: Assume that $(d^T c)_v \neq 0$ on some volume v , and evaluate the path integral on any neighborhood such that all vertices $p \in v$ are not boundary variables, and thus summed over. Then, the summation over all color 1-cocycles A cancels

$$\begin{aligned} Z_{\text{untw}}[b, c] &= \sum_A \prod_{p \in \Delta_0} (-1)^{A_p c_p} \\ &= \frac{1}{2} \sum_A \left(\prod_{p \in \Delta_0} (-1)^{A_p c_p} + \prod_{p \in \Delta_0} (-1)^{(A+dv)_p c_p} \right) \\ &= \frac{1}{2} \sum_A \prod_{p \in \Delta_0} (-1)^{A_p c_p} \left(1 + \prod_{p \in \Delta_0} (-1)^{(dv)_p c_p} \right) \\ &= \frac{1}{2} \sum_A \prod_{p \in \Delta_0} (-1)^{A_p c_p} \left(1 + \prod_{p \in v} (-1)^{c_p} \right) \\ &= \frac{1}{2} \sum_A \prod_{p \in \Delta_0} (-1)^{A_p c_p} \left(1 + (-1)^{(d^T c)_v} \right) = 0. \end{aligned} \quad (94)$$

The second property is that the path integral is invariant under adding the coboundary of a 1-chain β to b ,

$$Z_{\text{untw}}[b + d\beta, c] \propto Z_{\text{untw}}[b, c]. \quad (95)$$

As usual, this holds locally: If β consists of a single vertex p' , then it holds for the path integral on a small neighborhood of p' ,

$$\begin{aligned} Z_{\text{untw}}[b + dp', c] &= \sum_{A: dA=b+dp'} \prod_{p \in \Delta_0} (-1)^{A_p c_p} \\ &\stackrel{*}{=} \sum_{A: dA=b} \prod_{p \in \Delta_0} (-1)^{(A+p')_p c_p} \\ &= Z_{\text{untw}}[b, c] \prod_{p \in \Delta_0} (-1)^{(p')_p c_p} \\ &= Z_{\text{untw}}[b, c] (-1)^{c_{p'}}. \end{aligned} \quad (96)$$

In the equation labeled $*$, we have performed a change of summation variable from A to $A + p'$. An analogous property holds for the charges: The path integral is invariant

under adding the boundary of a 2-chain γ to c ,

$$Z_{\text{untw}}[b, c + d^T \gamma] \propto Z_{\text{untw}}[b, c]. \quad (97)$$

Again, this equation holds locally: If γ consists of a single face f , then the equation holds for Z on a constant-size neighborhood of f ,

$$\begin{aligned} Z_{\text{untw}}[b, c + d^T f] &= \sum_{A: dA=b} \prod_{p \in \Delta_0} (-1)^{A_p (c+d^T f)_p} \\ &= \sum_{A: dA=b} \prod_{p \in \Delta_0} (-1)^{A_p c_p} \prod_{p \in \Delta_0} (-1)^{A_p (d^T f)_p} \\ &= Z_{\text{untw}}[b, c] (-1)^{(dA)_f} \\ &= Z_{\text{untw}}[b, c] (-1)^{b_f}. \end{aligned} \quad (98)$$

3. Relation to anyons and 1-form symmetries

The flux and charge configurations b and c can be viewed as *anomalous 1-form symmetry defects* [28,76,77] in the untwisted color path integral. Equations (95) and (97) make the defects “symmetric.” The “anomaly” corresponds to the fact that the equations only hold up to a global prefactor, as shown in Eqs. (96) and (98). The symmetries are “1-form” because color 2-cocycles and 1-cycles can be mapped to (three copies of) cellular 2-cocycles and 1-cycles, as shown in Sec. III C. These can be pictured as a closed-loop pattern inside the 3D space-time. In contrast, an ordinary global “0-form” symmetry would correspond to domain walls in spacetime, that is, closed-membrane pattern. Physically, the closed loops in spacetime can be interpreted as anyon worldlines, and the $(-1)^{c_\beta}$ and $(-1)^{b_\gamma}$ factors in Eqs. (96) and (98) correspond to their braiding.

4. Defects and Pauli flows

Let us briefly discuss what charge and flux defects look like if we describe the untwisted color path integral as a ZX tensor network as in Sec. II and relate them to the formalism of *Pauli flows* (also called “Pauli webs” or “stabilizer flows”) [25,34,78]. If a face f carries a flux defect, $b_f = 1$, then this means that the path-integral weight $\delta_{(dA)_f=0}$ is replaced by $\delta_{(dA)_f=1}$. Thus, this modified weight is represented by an X -tensor with a π -phase. If a vertex p carries a charge defect, $c_p = 1$, this corresponds to an additional weight $(-1)^{A_p}$, which can be implemented in the ZX diagram by adding a π -phase to the according Z-tensor.

In the context of ZX tensor networks, flux (charge) defects are related to *Pauli-X (Pauli-Z) insertions*. A Pauli-X (Pauli-Z) insertion corresponds to a set of bonds in the tensor network, where we insert a Pauli-X (Pauli-Z) operator. Note that Pauli-Z and X operators are 2-index Z

and X -tensors with a π -phase,

$$\begin{aligned} \text{Red circle with } a \text{ below and } b \text{ above} &= \delta_{a=b}(-1)^a = \langle b | Z | a \rangle, \\ \text{Blue circle with } a \text{ below and } b \text{ above} &= \delta_{a+b=1} = \langle b | X | a \rangle, \end{aligned} \quad (99)$$

using the notation in Eq. (7). Each Pauli- X (Pauli- Z) operator can be combined with the adjacent X -tensor (Z -tensor),

$$\begin{aligned} \text{Red circle with } a \text{ below and } b \text{ above} &= \text{Red circle with } a \text{ below and } b \text{ above}, \\ \text{Blue circle with } a \text{ below and } b \text{ above} &= \text{Blue circle with } a \text{ below and } b \text{ above}. \end{aligned} \quad (100)$$

This way, each Pauli- X (Pauli- Z) insertion is mapped to a flux configuration b (charge configuration c).

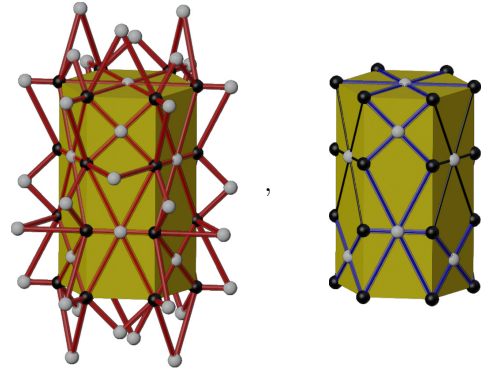
Pauli insertions can be deformed using the ZX rewrite rules: We can add a π -phase to the X -tensors at all 6 faces adjacent to a vertex p , by

$$\begin{aligned} \text{Vertex } p \text{ with 6 incident edges} &= \text{Vertex } p \text{ with 6 incident edges, each edge has a red circle at the vertex } p, \\ &= \text{Vertex } p \text{ with 6 incident edges, each edge has a blue circle at the vertex } p. \end{aligned} \quad (101)$$

This is the tensor-network formulation of the invariance in Eq. (95) for $\beta = p$. Analogously, we can add a π -phase to all the X -tensors at all the vertices around a face f , corresponding to the invariance in Eq. (95) for $\gamma = f$.

Now, we can also consider applying a set of invariances that leave b (c) unchanged, namely a set of vertices β (faces γ) which is a 1-cocycle with $d\beta = 0$ (or a 2-cycle with $d^T\gamma = 0$). Such sets of invariances are an example for *Pauli flows* [25,34,78], which have been used extensively for the construction of fault-tolerant circuits. Pauli flows can be defined for an arbitrary *stabilizer tensor network*, whose tensors are stabilizer states [34], and which is the kind of tensor network corresponding to circuits of

Clifford gates and postselected Pauli measurements. Pauli flows are usually drawn by coloring each bond in the tensor network by the Pauli operator ($\{I, X, Y, Z\}$ in general) that is inserted (and removed) at that bond during the application of the invariances. In our ZX tensor network, the Pauli flows consist either purely of X , or of Z operators, which we color in red and blue, respectively. For every volume v , there is one X Pauli flow given by $\beta = dv$. That is, β consists of all vertices of v , and the Pauli flow is given by marking all bonds adjacent to the according Z -tensors in red. Similarly, for each volume v and color $c \neq c_v$, there is one Z Pauli flow with $\gamma = d^T(v, c)$. That is, γ consists of all faces of v whose color does not contain c , and the Pauli flow is given by marking all bonds adjacent to the according X -tensors in blue. Such local Pauli flows are called *detector flows*. The following shows an example of the X detector flow, and one of the Z detector flows at a yellow volume in the 3-colex,



$$(102)$$

If we insert a Pauli- Z operator at a red edge of a (Pauli- X) detector flow, applying the according invariance yields a global factor of -1 due to the anticommutativity of X and Z . The analogous holds when inserting a Pauli- X operator at a blue edge of a (Pauli- Z) detector flow. In this sense, X detector flows (red) detect Pauli- Z (and Y) insertions and Z detector flows (blue) detect Pauli- X (and Y) insertions. We find that any local Pauli insertion that is not equivalent to the trivial insertion via ZX rewrite rules, is detectable, and hence, the protocols have a macroscopic fault distance. Note that this also is guaranteed more generally by the fixed-point property of the path integral, which manifests in terms of linear constraints on the flux and charge configurations as discussed above.

Pauli flows also describe the logical action of a Clifford protocol [25,34]. We have established that the local equivalences among Pauli insertions in the ZX diagram, that combine to Pauli flows, agree with 1-cocycle and 2-cycle condition in the color path integral. This means that the X (Z) logical flows are given by nontrivial color 1-cocycles (2-cycles).

5. Twisted case: Definition

Let us now return to the original, twisted, color path integral used to construct twisted color circuits. For each nontrivial flux configuration b and charge configurations c we define a modified (twisted) color path integral, similar to the untwisted case,

$$Z[b, c] = \sum_{A \in \mathbb{Z}_2^{\Delta_0}} \prod_{dA=b} \prod_{p \in \Delta_0} e^{2\pi i (\frac{1}{8} \text{sgn}_p \overline{A_p} + \frac{1}{2} A_p c_p)}. \quad (103)$$

Note that we recover the color path integral in Eq. (76) for $b, c = 0$. Compared to the untwisted case, the properties in Eqs. (92), (93), (95), and (97) are modified due to the nontrivial action entering $Z[0, 0]$. Equation (92) holds unchanged, since it only relies on the fact that $db = ddA = 0$. Equation (95) does not hold anymore, and it is unclear whether there is a meaningful way to fix it. Luckily, we do not need to rely on this property for fault tolerance. The constraint and invariance in Eqs. (93) and (97) are modified in the twisted case in the presence of nontrivial flux, $b \neq 0$: While the constraint in Eq. (93) gets replaced by a weaker constraint, the invariance in Eq. (97) can be extended to a stronger invariance. Note that b and c correspond to configurations of measurement outcomes in the twisted color circuits, and the modified c constraints correspond to modified detectors generating a modified syndrome. Using this modified syndrome may be important for effective decoding, as we discuss in Sec. V. In the following we study these modified constraints, which we refer to as *twisted charge constraints*.

6. Twisted charge constraints

As we saw earlier, the (generating) charge constraints correspond to detectors in the context of quantum error correction. We will therefore refer to the generating twisted charge constraints as *twisted charge detectors* and denote their set by X . The twisted charge constraint is of the form

$$K^b d^T c = k^b, \quad (104)$$

where the matrix $K^b \in \mathbb{Z}_2^{X \times \Delta_3}$ and the vector $k^b \in \mathbb{Z}_2^X$ are defined as follows. First, consider the symmetric matrix $M^b \in \mathbb{Z}_2^{\Delta_3 \times \Delta_3}$, defined by

$$M_{vv'}^b = \sum_{f \in \Delta_2, f \in v, v'} b_f. \quad (105)$$

That is, if v and v' do not share a face, then $M_{vv'}^b = 0$. If they share a face f , then $M_{vv'}^b$ equals the value of b on f . For $v = v'$, M_{vv}^b is equal to the sum of b over all faces of v .

Given M^b , K^b can be chosen to be any maximum-rank matrix such that

$$(K^b M^b)_{xv} = \sum_{v' \in \Delta_3} K_{xv'}^b M_{v'v}^b = 0 \quad \forall x \in X, v \in \Delta_3. \quad (106)$$

In other words, we choose K^b such that the vectors $\{K_x^b\}_{x \in X}$ form a complete basis for the kernel of M^b . Finally, in order to define k^b , we need to choose a fixed A_0 such that $dA_0 = b$. Then, we set

$$k_x^b := \frac{1}{2} \sum_{v \in \Delta_3} \sum_{p \in \Delta_0, p \in v} (A_0)_p K_{xv}^b + \sum_{v < v'} M_{vv'}^b K_{xv'}^b K_{xv''}^b. \quad (107)$$

The expression above takes integer values, as we show below, and is interpreted mod 2.

Before we verify that the twisted charge constraint in Eq. (104) indeed holds, let us discuss how it differs from the charge constraint in the untwisted color path integral. First, we find that for every volume v such that $b_f = 0$ on all faces $f \in v$, we have $M_{vv'}^b = 0 \forall v'$. Thus, each such volume v also gives rise to a twisted detector x with $K_{xv}^b = \delta_{x=v}$. More generally, consider a connected component $V \subset \Delta_3$, where two volumes are considered connected if they share a face f with $b_f = 1$. Then, we have $M_{vv'}^b = 0 \forall v \in V, v' \neq v$. Thus, we can choose a basis of twisted detectors, such that twisted detector is supported only on one connected component. On a high level, the twisted charge constraint can be summarized as follows: The volumes adjacent to a nontrivial flux face do not yield twisted charge detectors individually as in the untwisted case, but instead each connected component of such volumes together yields a small number of twisted charge detectors.

After stating the twisted charge constraints, let us now show that they hold. We begin by defining the *gauge variance* of the action S ,

$$\Delta S_\alpha[A] = S[A + d\alpha] - S[A], \quad (108)$$

for any color 1-cocycle A and 0-cochain α . We find for any pair of volumes v and v' ,

$$\begin{aligned}
\Delta S_{v+v'}[A] - \Delta S_v[A] - \Delta S_{v'}[A] &\stackrel{(108)}{=} S[A + dv + dv'] - S[A + dv] - S[A + dv'] + S[A] \\
&= \frac{1}{8} \sum_p \text{sgn}_p \left(\overline{(A + dv + dv')_p} - \overline{(A + dv)_p} - \overline{(A + dv')_p} + \overline{A_p} \right) \\
&\stackrel{(74)}{=} \frac{1}{8} \sum_p \text{sgn}_p \left(\overline{(dv')_p} - 2\overline{(A + dv)_p} \overline{dv'_p} - \overline{(dv')_p} + 2\overline{A_p} \overline{(dv')_p} \right) \\
&\stackrel{(74)}{=} \frac{1}{4} \sum_p \text{sgn}_p \left(-\overline{A_p} + \overline{(dv)_p} - 2\overline{A_p} \overline{(dv)_p} \overline{(dv')_p} + \overline{A_p} \overline{(dv')_p} \right) \\
&= \sum_p \text{sgn}_p \left(-\frac{1}{4} \overline{(dv)_p} \overline{(dv')_p} + \frac{1}{2} \overline{A_p} \overline{(dv)_p} \overline{(dv')_p} \right) \\
&\stackrel{*}{=} \frac{1}{2} \sum_p \overline{A_p} \overline{(dv)_p} \overline{(dv')_p} = \frac{1}{2} \sum_{f: f \in v, f \in v'} \sum_{p \in f} A_p \\
&= \frac{1}{2} \sum_{f: f \in v, f \in v'} b_f = \frac{1}{2} M_{vv'}^b. \tag{109}
\end{aligned}$$

For the equality labeled (*), we have used that two volumes v and v' share either no vertices p , or they share a face and all adjacent vertices. Since each face has equal number of vertices with $\text{sgn}_p = 1$ and $\text{sgn}_p = -1$, the first term vanishes. In words, the equation says that $\Delta S_\alpha[A]$, as a function of α for a fixed A , is a *quadratic refinement* [79] of the $\mathbb{R}/\mathbb{Z}$ -valued bi-linear form defined by the matrix M^b . Using this, we find

$$\begin{aligned}
\Delta S_{K_x^b}[A_0] &\stackrel{(109)}{=} \sum_v K_{xv}^b \Delta S_v[A_0] + \frac{1}{2} \sum_{v < v'} M_{vv'}^b K_{xv}^b K_{xv'}^b \\
&\stackrel{(107)}{=} \frac{1}{2} k_x^b, \tag{110}
\end{aligned}$$

using the notation $K_x^b := \sum_v K_{xv}^b \cdot v$ for the x th row of the matrix K^b . We can now use Eq. (109) to show that (1) k^b is integral, and (2) it can only depend on the cohomology class of A_0 . For (1), we observe that

$$\begin{aligned}
-2\Delta S_{K_x^b}[A_0] &= \Delta S_0[A_0] - \Delta S_{K_x^b}[A_0] - \Delta S_{K_x^b}[A_0] \\
&\stackrel{(109)}{=} \frac{1}{2} (K_x^b)^T M^b K_x^b = 0, \tag{111}
\end{aligned}$$

which implies that $k_x^b = -2\frac{1}{2}k_x^b = 0 \bmod 1$, recalling that both equations above are in $\mathbb{R}/\mathbb{Z}$. For (2), we calculate

$$\begin{aligned}
\Delta S_{K_x^b}[A_0 + d\alpha] &\stackrel{(108)}{=} \Delta S_{K_x^b + \alpha}[A_0] - \Delta S_{K_x^b}[A_0] \\
&\quad - \Delta S_\alpha[A_0] + \Delta S_{K_x^b}[A_0]
\end{aligned}$$

$$\begin{aligned}
&\stackrel{(109)}{=} \frac{1}{2} \alpha^T M^b K_x^b + \Delta S_{K_x^b}[A_0] \\
&\stackrel{(106)}{=} \Delta S_{K_x^b}[A_0]. \tag{112}
\end{aligned}$$

To show the twisted charge constraint in Eq. (104), consider a patch of the spacetime 3-colex such that K_x^b is supported on its interior. If $(K^b d^T c)_x \neq k_x^b$, then we find

$$\begin{aligned}
Z[b, c] &= \sum_A e^{2\pi i (S[A] + \frac{1}{2} \sum_p A_p c_p)} \\
&= \frac{1}{2} \left(\sum_A e^{2\pi i (S[A] + \frac{1}{2} \sum_p A_p c_p)} \right. \\
&\quad \left. + e^{2\pi i (S[A + dK_x^b] + \frac{1}{2} \sum_p (A + dK_x^b)_p c_p)} \right) \\
&\stackrel{(108)}{=} \frac{1}{2} \sum_A e^{2\pi i S[A]} \left(1 + e^{2\pi i (\Delta S_{K_x^b} + \frac{1}{2} \sum_p (dK_x^b)_p c_p)} \right) \\
&\stackrel{(110)}{=} \frac{1}{2} \sum_A e^{2\pi i S[A]} \left(1 + (-1)^{k_x^b + (K^b d^T c)_x} \right) \\
&= \frac{1}{2} \sum_A e^{2\pi i S[A]} (1 - 1) = 0. \tag{113}
\end{aligned}$$

In addition to the weakened (twisted) constraints, the invariance in Eq. (97) can be extended: For any locally supported 1-chain Γ such that $K^b d^T \Gamma = 0$, we have

$$Z[b, c + \Gamma] \propto Z[b, c]. \tag{114}$$

This includes 1-chains of the form $\Gamma = d^T \gamma$, which form the invariance in the untwisted case.

7. Physical interpretation of twisted constraints

Let us briefly discuss the physics interpretation of our findings. The twisted constraint for the charge configuration in the (twisted) color path integral mirrors the fusion rules of the anyon theory of the underlying non-Abelian phase. Recall that in Sec. IV C, we showed the phase equivalence between the color path integral and the type-III twisted $\mathbb{Z}_2^3$ Dijkgraaf-Witten state sum, whose anyon model is known [71]. Using the cohomology mapping in Sec. III C, we can generate special charge or flux configurations that look like isolated strings, which we can interpret as *anyon worldlines*. Each of the three generating flux strings is associated with two non-Abelian anyons of quantum dimension 2 [80]. The fact that the flux constraint in Eq. (92) is unchanged mirrors the fact that the fusion of two non-Abelian flux anyons of the same type always yields an anyon with trivial flux. However, due to their non-Abelian nature, there is only a topological invariance for isolated flux worldlines, but no cohomological invariance for some (possibly dense) network of fluxes as for the untwisted case in Eq. (95) [81]. The charge configuration still corresponds to a pattern of Abelian anyon worldlines in the path integral. The fact that the charge constraint $d^T c = 0$ is weakened and does not need to hold on volumes with nontrivial flux faces mirrors the fact that the fusion of two non-Abelian fluxes can result in Abelian charges. Or, in other words, the Abelian charge worldlines can terminate on the flux worldlines.

8. Charges and fluxes in 3D color code

The color path integral is closely related to the 3D color code and its transversal T gate [31], see Sec. II. Let us briefly discuss to what kind of objects the charge and flux configurations and their constraints correspond in the 3D color code. The flux configuration b corresponds to a ± 1 configuration of measurement outcomes of the Z stabilizers S_f^Z , or in other words, a Z syndrome or Z excitation of the 3D color code. The charge configuration c , on the other hand, does not correspond to the syndrome for the X stabilizers S_v^X . Rather, the operator acting as Z^{c_p} at every 3-colex vertex p maps the code state of the 3D color code to one with X syndrome given by $d^T c$.

Let us next study what happens if we apply the transversal T gate $\bar{T} = \prod_{p \in \Delta_0} T_p^{\text{sgn}_p}$ to a color-code state with Z syndrome b and X syndrome $d^T c$ [51]. The action of the transversal T gate on the X syndrome is trivial, since the Z^{c_p} operator that creates it commutes with T . It thus suffices to study the action on a state $|b\rangle$ with a definite flux syndrome b but trivial charge syndrome. After applying the $\bar{T}$ gate, $\bar{T}|b\rangle$ is an eigenstate of the conjugated S_v^X stabilizers,

$$\tilde{S}_v^X := \bar{T} S_v^X \bar{T}^\dagger = \prod_{p \in v} S_p^{\text{sgn}_p} X_p, \quad (115)$$

where $S = i^{|\chi|}$ denotes the single-qubit Clifford phase gate. In order to determine the X syndrome of $\bar{T}|b\rangle$, we measure the original stabilizers S_v^X . If S_v^X and $\tilde{S}_v^X$ commute, then the measurement outcome will be deterministic, and if they anticommute it will be fully random. Computing the group commutator $[A, B]_G := ABA^{-1}B^{-1}$ yields

$$[S_v^X, \tilde{S}_{v'}^X]_G = \prod_{f: f \in v, f' \in v'} S_f^Z. \quad (116)$$

Accordingly, we find

$$\begin{aligned} \langle b | \prod_{f: f \in v, f' \in v'} S_f^Z | b \rangle &= (-1)^{\sum_{f: f \in v, f' \in v'} b_f} \\ &= (-1)^{M_{vv'}^b}, \end{aligned} \quad (117)$$

recovering the quadratic form from Eq. (109) that describes the gauge variance. S_v^X will yield a deterministic outcome on $\bar{T}|b\rangle$ exactly if it commutes with all $\tilde{S}_{v'}^X$, which is the case if $M_{vv'}^b = 0$ for all v' . More generally, a product

$$\prod_{v \in \Delta_3} (S_v^X)^{\alpha_v} \quad (118)$$

of generating S_v^X stabilizers yields a deterministic outcome if the vector $\alpha \in \mathbb{Z}^{\Delta_3}$ is in the kernel of M^b . So the subgroup of the original X -stabilizer group that yields deterministic outcomes is generated by

$$\left\{ \prod_{v \in \Delta_3} K_{xv}^b S_v^X \right\}_{x \in X}, \quad (119)$$

using the matrix K^b from Eq. (106). The deterministic outcome of the measurement does not have to be $+1$. Instead, it is given by k_x^b from Eq. (107),

$$\prod_{v \in \Delta_3} K_{xv}^b S_v^X \bar{T}|b\rangle = (-1)^{k_x^b} \bar{T}|b\rangle. \quad (120)$$

Concretely, such deterministic -1 outcomes can occur for configurations b consisting of linked loops, a phenomenon which was called *linking charge* in Ref. [51].

E. Boundaries

In this section, we revisit the boundaries of the path integral in Sec. II D in more detail and describe other constructions for topological boundaries. In order to define a boundary condition for the color path integral, we need to (1) extend color cohomology to 3-colexes with boundaries, and (2) extend the action in Eq. (77) to the boundaries. (1) and (2) need to satisfy two crucial conditions: (1) the

extension of color cohomology to the boundary needs to be locally exact, and (2) the extension of the action to the boundary needs to be gauge invariant. We will here only state the different boundaries, and derive them explicitly in Appendix A. There, we also discuss examples of boundaries in 2D and boundaries that do not fulfill local exactness or gauge invariance, and we show that we have found all possible boundaries using their equivalence to boundaries of a cellular twisted path integral.

Geometrically, we will define the boundaries on an arbitrary 2D surface formed by faces of the 3-colex. For any such surface, we introduce the following notation:

- (a) $\Delta_i^{\text{bd},c}$ for $i = 0, 1, 2$ denotes the set of i -cells of the 3-colex that are part of the boundary and have coloring c .
- (b) $\Delta_3^{\text{bdin},c} \subset \Delta_3$ denotes the set of volumes that are adjacent to a boundary face.
- (c) $\Delta_3^{\text{bdout},c} \not\subset \Delta_3$ denotes the set of “phantom” volumes that are not part of the 3-colex, but adjacent to a boundary face on the “outside.”
- (d) $\Delta_2^{\text{bdout}} \not\subset \Delta_2$ denotes the set of “phantom” faces that are adjacent to one or more boundary edges in Δ_1^{bd} but outside the boundary.
- (e) $\Delta_0^{\text{bdout},c} \subset \Delta_0^{\text{bd}}$ ($\Delta_0^{\text{bdin},c} \subset \Delta_0^{\text{bd}}$) denotes the set of boundary vertices that are adjacent to a single volume in $\Delta_2^{\text{bdout},c}$ ($\Delta_2^{\text{bdin},c}$), which is of color c .

1. All-rough boundary

The first boundary type is the *all-rough boundary*. The generator sets of the boundary cohomology are

$$\begin{aligned} X_0 &:= \Delta_3, & X_1 &:= \Delta_0, \\ X_2 &:= \Delta_2 + \Delta_2^{\text{bdout}}, & X_3 &:= \bigsqcup_{c \in \mathcal{C}_1} (\Delta_3^c + \Delta_3^{\text{bdout},c}) \times \bar{c}, \end{aligned} \quad (121)$$

and the boundary maps d_0, d_1, d_2 are simple restrictions of these in the bulk. The boundary action looks the same as the bulk one in Eq. (77), where the summation runs over all vertices $p \in \Delta_0$ including those on the boundary. The ZX tensor network for this boundary is discussed later in Sec. VI, where it is depicted in Fig. 23.

2. Color boundary

The second family of boundaries are *color boundaries*. There are four such boundaries, one for every color c , where we set $c = r$ (red) without loss of generality. The

cohomology theory is

$$\begin{aligned} X_0 &:= \Delta_3 - \Delta_3^{\text{bdin},r} + \Delta_3^{\text{bdout},g} + \Delta_3^{\text{bdout},b} + \Delta_3^{\text{bdout},y}, \\ X_1 &:= \Delta_0, \\ X_2 &:= \Delta_2 - \Delta_2^{\text{bd},gb} - \Delta_2^{\text{bd},gy} - \Delta_2^{\text{bd},by} + \Delta_1^{\text{bd},\bar{r}}, \\ X_3 &:= \bigsqcup_{c \in \mathcal{C}_1} \Delta_3^c \times \bar{c}. \end{aligned} \quad (122)$$

d_1^T maps an edge $e \in \Delta_1^{\text{bd},\bar{r}}$ to both of its endpoints. d_2 acts on an edge $e \in \Delta_1^{\text{bd},\bar{r}}$ as follows: First, we map e to all adjacent faces in $\Delta_2^{\text{bd},gb}$, $\Delta_2^{\text{bd},gy}$, and $\Delta_2^{\text{bd},by}$. Then, we act on these faces with the bulk coboundary map d_2 from Eq. (49). Otherwise, the coboundary maps are restrictions of the bulk ones in Eq. (49). The boundary action extends from Eq. (77) to the boundary as follows:

$$S^{\text{bd},r}[A] = \frac{1}{8} \sum_{p \in \Delta_0 - \Delta_0^{\text{bdin},r}} \text{sgn}_p \bar{A}_p. \quad (123)$$

This boundary is the one used in Sec. II.

3. Double-color boundary

The third family of boundaries are *double-color boundaries*. There are six such boundaries, one for each color pair κ . Without loss of generality, we choose $\kappa = rg$. The cohomology theory is

$$\begin{aligned} X_0 &:= \Delta_3 - \Delta_3^{\text{bdin},r} - \Delta_3^{\text{bdin},g} + \Delta_3^{\text{bdout},b} + \Delta_3^{\text{bdout},y}, \\ X_1 &:= \Delta_0, \\ X_2 &:= \Delta_2 - \Delta_2^{\text{bd}} + \Delta_2^{\text{bd},rg} + \Delta_1^{\text{bd},\bar{r}} + \Delta_1^{\text{bd},\bar{g}}, \\ X_3 &:= \bigsqcup_{c \in \mathcal{C}_1} \Delta_3^c \times \bar{c} + \Delta_2^{\text{bd},by}. \end{aligned} \quad (124)$$

d_2^T maps a boundary face $f \in \Delta_2^{\text{bd},by}$ to all of its edges. Apart from this, the coboundary maps are defined similarly to Eq. (122). The boundary action looks the same as in Eq. (123). The according ZX tensor network is shown in Sec. VI in Fig. 23.

In total, there are 11 distinct boundaries of the 3D twisted color path integral corresponding to three families, namely one all-rough, four color, and six double-color boundaries.

V. DECODING AND FAULT TOLERANCE

In this section, we discuss in more detail how to decode the twisted color circuit from Sec. II. We describe decoding in five different steps: In Sec. VA we explain how

to decode the untwisted color circuit without T gates. In Sec. VB, we discuss how the presence of T gates in the twisted color circuit makes it necessary to decode and apply X corrections while running the circuit, and how this can be achieved using a *just-in-time* decoder. While Z errors could be corrected in the same way as in the untwisted circuit, we show in Sec. VC how the output of just-in-time X decoding can be used to improve the Z decoding of the twisted circuits. In Sec. VD, we discuss global aspects of the decoding procedure for a logical circuit using our non-Clifford logical elements. We phrase the subroutines of the decoding problem in all these sections as hypergraph-matching problems, for which we propose concrete (efficient) algorithms in Sec. VE.

A. Decoding of the untwisted color circuits

Let us warm up by discussing how to perform decoding and corrections in the *untwisted* color circuits, where we omit the physical T gates. Similarly to other circuits for Abelian topological phases, 2D toric code [2], color code [30,82,83], or honeycomb Floquet code [84,85], it suffices to record all measurement outcomes while running the circuit and decode at a later time. This is the main difference to the *twisted* color circuits which we discuss in the following sections, where we need to perform decoding and corrections while executing the circuit.

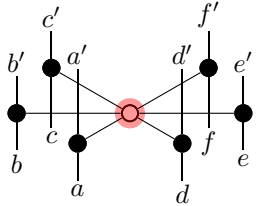
1. Decoding in 3 + 0D

Let us recall how decoding works for the 3 + 0D measurement-based protocol discussed in Sec. IIB, without the T gates. For concreteness, we assume a 3-colex of the topology $S_1 \times S_1 \times [0, 1]$, where the task is to reliably teleport the logical information from the input boundary $S_1 \times S_1 \times 0$ to the output boundary $S_1 \times S_1 \times 1$. After running the circuit and recording all measurement results, the task is to apply corrections to the output state such that it represents the logical state of the input.

2. Measurements in 3 + 0D

We start by translating the measurement outputs into a flux configuration b and charge configuration c in the color path integral, as discussed in Sec. IVD: We recall that, if all X and Z measurement outcomes are +1, the postselected circuit is equal to the untwisted color path integral as discussed in Sec. IIB. If a Z measurement outcome at a face is -1 , then this corresponds to adding a π phase to

the X -tensor at that face, for example, for a 6-gon face,



$$\begin{aligned}
 &= \delta_{a=a'} \delta_{b=b'} \delta_{c=c'} \delta_{d=d'} \delta_{e=e'} \delta_{f=f'} \delta_{a+b+c+d+e+f=1} \\
 &= \langle a', b', c', d', e', f' | \frac{1}{2} (1 - Z^{\otimes 6}) | a, b, c, d, e, f \rangle, \quad (125)
 \end{aligned}$$

using notation from Eq. (7). As discussed in Sec. IVD, this π phase corresponds to a nontrivial flux ($b_f = 1$) on the face f . Similarly, a -1 outcome of a destructive X measurement corresponds to adding a π phase to the according Z -tensor, which corresponds to a nontrivial charge ($c_p = 1$) on the vertex p ,

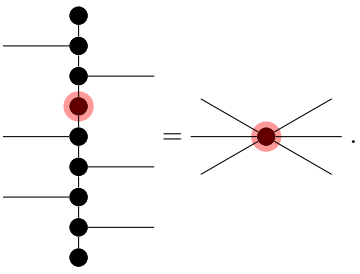


$$= (-1)^a = \sqrt{2} \langle -|a \rangle. \quad (126)$$

All in all, the Z measurement outcomes (relabeled as $\{1, -1\} \rightarrow \{0, 1\} \simeq \mathbb{Z}_2$) form a flux configuration b which is a color 2-chain, and the X measurement outcomes form a charge configuration c which is a color 1-chain.

3. Errors in 3 + 0D

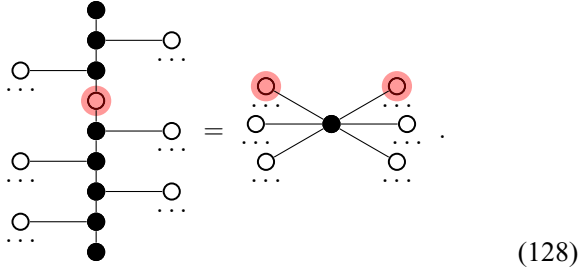
Let us next look at Pauli- X and Z errors in the circuit. Pauli- X and Z operators correspond to 2-index X or Z -tensors with a π phase as shown in Eq. (99). The circuit with Pauli- X and Z errors corresponds to the ZX tensor network flux and charge configurations b_E and c_E , similar to the Pauli insertions discussed in Sec. IVD. Concretely, using the relation between the protocol and the path integral in Eq. (13), a Z error at a qubit at any point in the protocol adds to the value c_E on the corresponding vertex,



$$\quad (127)$$

As another example, an X error occurring after the qubit has been involved in 4 of the 6 face measurements adds to

the value of b_E at the two remaining faces,



Equivalently, the X error contributes to the value of b_E on the four other faces. All in all, an X (Z) error configuration can be represented as a $\mathbb{Z}_2$ -valued vector b_{err} (c_{err}) with one entry for every location in the circuit where an error can occur. The redistribution of π -phases above defines a local $\mathbb{Z}_2$ -linear map which takes b_{err} (c_{err}) to a color 2-cochain b_E (1-chain c_E). This map itself is not unique, as we can locally change $b'_E = b_E + d\beta$ ($c'_E = c_E + d^T\gamma$) via the invariance in Eq. (95) [Eq. (97)]. However, the syndrome db_E ($d^T c_E$) is uniquely determined by b_{err} (c_{err}), due to $dd = 0$.

4. Decoding algorithm in 3 + 0D

The output state of the protocol at $S_1 \times S_1 \times 1$ is not a color code state, but a state with X (Z) syndrome given by the termination of $b + b_E$ ($c + c_E$) at the state boundary $S_1 \times S_1 \times 1$. So we need to apply a correction that maps us back into the code space. This correction is equal to inserting another flux (charge) configuration b_c (c_c) which is only supported near $S_1 \times S_1 \times 1$. The whole circuit with measurements, errors, and corrections forms a path integral with a total flux (charge) configuration given by $b + b_E + b_c$ ($c + c_E + c_c$). Due to the constraints in Eq. (92) [Eq. (93)], we know that this effective flux configuration $b + b_E + b_c$ (charge configuration $c + c_E + c_c$) is a color 2-cocycle (1-cycle), otherwise measuring b (c) has zero probability. Due to the invariance in Eq. (95) [Eq. (97)], a logical error occurs only if the effective flux (charge) configuration is (co)homologically nontrivial. In other words, no logical fault occurs if we choose b_c (c_c) to be cohomologically equivalent to $b + b_E$ ($c + c_E$). Unfortunately, we do not know b_E (c_E), but only its syndrome $db_E = db$ ($d^T c_E = d^T c$). So the best we can do is to replace it with an estimate $\tilde{b}$ ($\tilde{c}$) that also satisfies

$$d\tilde{b} = db, \quad (d^T \tilde{c} = d^T c). \quad (129)$$

It is the task of the decoder to find this estimate, and it succeeds if $\tilde{b}$ ($\tilde{c}$) is cohomologically equal to b_E (c_E). A strategy that generally works is to pick the $\tilde{b}$ ($\tilde{c}$) fulfilling Eq. (129) with the smallest weight, or at least as small as we can find within runtime limitations. Here, the weight is the number of faces f (vertices v) with $\tilde{b}_f = 1$ ($\tilde{c}_v = 1$).

We will refer to the problem of low-weight finding solutions to Eq. (129) as *hypergraph matching*, and provide concrete algorithms in Sec. V E. Very roughly, the reason why this works is that for low enough error rate, b_E (c_E) has low weight with high probability, thus if we choose $\tilde{b}$ with low weight it will have the same (co)homology class than b_E (c_E) with high probability. Finally, we note that the fault-tolerance argument can also be made for any arbitrary p -bounded local circuit-level noise model, without expressing it in terms of Pauli errors, which we do in Appendix B.

5. String-pattern picture

After applying the mapping in Sec. III C, coarse-graining and abstracting away the lattice, both b and c can be imagined as string patterns. More precisely, each of b and c corresponds to 3 independent string patterns of three different colors. db and $d^T c$ are the endpoints of these string patterns. b_E and c_E , as well as $\tilde{b}$ and $\tilde{c}$ are low-weight string patterns with the same endpoints. $b + \tilde{b}$ and $c + \tilde{c}$ are string patterns that are closed everywhere except for the points where they terminate on the output state boundary $S_1 \times S_1 \times 1$. b_c and c_c close off these termination points inside the output state boundary.

6. Decoding in 2 + 1D

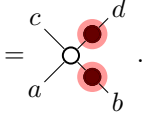
Decoding of a 2 + 1D untwisted color circuit (still without physical T gates) works essentially in the same way as described above for the 3 + 0D protocol. There are a few small differences which we list below. First, we do not get all the measurement outcomes at the same time, but we record them over time to then perform decoding at a later time step. This may be either when we pass the logical information into some other gadget that requires a clean corrected state, or when determining the classical outcome of a logical measurement, which is encoded in the charge configuration.

7. Measurement outcomes in 2 + 1D

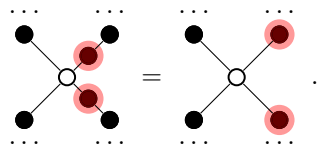
The second difference is that there is no one-to-one correspondence between individual Z -type (X -type) measurement outcomes in the circuit and the flux (charge) value at an individual face (vertex) in the 3-colex. Instead, the history of all Z measurement outcomes (X measurement outcomes) form a $\mathbb{Z}_2$ -valued vector b_{meas} (c_{meas}), which is related to the flux configuration b (charge configuration c) by a local $\mathbb{Z}_2$ -linear map, just like the relation between b_{err} and b_E (c_{err} and c_E) above. Again, b (c) is only defined up to local 2-coboundaries (1-boundaries), but its syndrome db ($d^T c$) is uniquely determined by b_{meas} (c_{meas}).

To get a map between b_{meas} and b (c_{meas} and c), we express the -1 -postselected projectors of each measurement in the circuit as a ZX diagram, and move the corresponding π -phases to the nearby tensors. Let us give a few

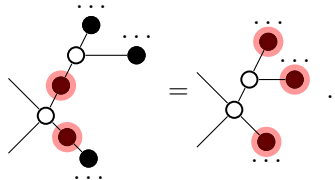
examples of this map for the circuit in Fig. 3 discussed in Sec. II. The -1 projector of a 2-qubit XX measurement can be written with two additional π -phase Z -tensors,

$$\langle c, d | (1 - XX) | a, b \rangle = (-1)^{d+b} \cdot \delta_{a+b+c+d=0}$$

(130)

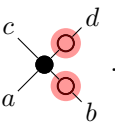
We can choose the π Z -tensors to be on either side. The π -phase Z -tensors can be fused with the nearby Z -tensors in the ZX path integral. With this, a -1 outcome of the XX measurement in Fig. 3 corresponding to a time-parallel square in Fig. 1 contributes a charge to two of its vertices,


(131)

Note that we could also have chosen the two vertices on the left rather than those on the right. Similarly, a -1 outcome of the XX measurement at a time-parallel brick hexagon contributes a π -phase to three of its vertices, since pushing the π -phase through part of the CX gate yields a third π -phase,


(132)

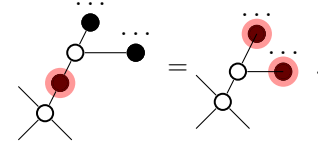
A -1 outcome of the $Z^{\otimes 6}$ measurement at a time-perpendicular hexagon still yields a single π -phase at the corresponding Z -tensor, as shown in Eq. (125). On the other hand, the -1 projector of a 2-qubit ZZ measurement can be written with two additional π -phase X -tensors,

$$\langle c, d | \frac{1}{2}(1 - ZZ) | a, b \rangle = (-1)^a \cdot \delta_{a=b=c=d} =$$

(133)

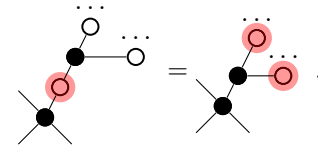
Again, we can move the π -phases of these X -tensors onto the nearby X -tensors. For example, the ZZ -measurement in Fig. 3 corresponding to a 3-colex vertex gives rise to a π -phase at three adjacent faces, namely one past square, one future brick hexagon, and the time-perpendicular hexagon.

8. Pauli errors in 2 + 1D

As for b_{meas} (c_{meas}), also the local $\mathbb{Z}_2$ -linear map from b_{err} to b_E (c_{err} to c_E) is different for the 2 + 1D untwisted color circuit. For example, consider a Z error acting on a qubit after an XX measurement and before the following CX gate in the untwisted color circuit (shown in Fig. 3 if we ignore the T gates). Recall that these XX measurements and CX gates originate from splitting up the X -tensor associated to brick hexagon face as shown in Eq. (18). Such a Z error yields π -phases at two of the vertices of the brick hexagon,


(134)

As another example, apply an X error in Fig. 3 to a qubit, after it is involved in the ZZ measurement (originating from splitting the Z -tensor at a vertex) and before a $Z^{\otimes 6}$ measurement. Such an X error gives rise to two π -phases at two of the 3-colex faces adjacent to the vertex, namely the future square as well as the time-perpendicular hexagon face,


(135)

At most other places in the circuit, a single-qubit X or Z error just corresponds to a π -phase at a single neighboring X or Z tensor.

Apart from the different maps from b_{err} to b_E (c_{err} to c_E) and from b_{meas} to b (c_{meas} to c), decoding works as in the 3 + 0D case.

B. Just-in-time decoding of flux configuration

So far, we have described the effect of (Pauli) errors and their decoding in an untwisted color circuit. Now, we consider how decoding of X errors changes for the *twisted* color circuit including the T gates. Roughly speaking, applying the T gates creates an additional random charge syndrome in $d^T c$ supported near the flux configuration $b + b_E$. In order to contain the damage caused by this additional charge syndrome, it is necessary to choose b and apply corrections while running the circuit, in contrast to the untwisted case described in the previous section.

1. Flux corrections in 3 + 0D

In the 3 + 0D protocol, we need to apply corrections before applying the transversal T^{sgn_p} gate. The reason for

this is that otherwise the protocol corresponds to the color path integral with an overall flux configuration $b + b_E$. As we have seen around Eq. (104), the flux configuration modifies the constraint for the charge configuration, effectively decreasing the fault distance for Z errors. While b_E is a low-weight configuration for low error rates, b is completely random apart from the constraint $db = db_E$. Thus, a typical flux configuration of $b + b_E$ is dense everywhere, and effectively decreases the Z fault distance to a constant, spoiling fault tolerance completely.

The correction operator we need to apply to trivialize the flux configuration is of the form $\prod_p X_p^{G_p}$, that is, we need to apply a Pauli- X operator at every vertex with $G_p = 1$, where G is a color 1-cochain which we call the *gauge-fixing configuration*. The correction correctly prepares a 3D color code state with trivial flux configuration for any choice of G such that $dG = b + b_E$. Since we do not know b_E , we have to replace it with its estimate $\tilde{b}$. In that case, applying a correction with G such that $dG = b + \tilde{b}$ does not yield an ideal code state, but one with a *residual flux configuration* of $\tilde{b} + b_E$. To see this precisely, we realize that the whole circuit including X errors and corrections, with measurement outcomes postselected onto b and c , is equal to the path integral

$$\begin{aligned} Z &= \sum_{A: dA=b+b_E} e^{2\pi i S[A+G] + \frac{1}{2} c \cdot A} \\ &\propto \sum_{A: dA=\tilde{b}+b_E} e^{2\pi i S[A] + \frac{1}{2} c \cdot A}, \end{aligned} \quad (136)$$

where $c \cdot A$ is the generator-wise product of c and A . In the second line, we have performed a change of summation variable $A \rightarrow A + G$, and ignored a global prefactor of $(-1)^{c \cdot G}$. All in all, the flux decoding does effectively reduce the code distance for Z errors, but only by a tolerable amount since $\tilde{b}$ and b_E are both low weight.

Note that without the transversal T^{sgn_p} gate, the X corrections are unnecessary since they have no effect when combined with the subsequent X measurements. Also note that instead of applying the X^{G_p} correction, we can replace the transversal T^{sgn_p} gate with $X^{G_p} T^{\text{sgn}_p} X^{G_p} \propto T^{\text{sgn}_p \cdot (-1)^{G_p}}$. That is, we swap between T and T^{-1} at all the vertices p with $G_p = 1$. If we do not apply the X corrections before performing the transversal T^{sgn_p} gate, this leads to an uncorrectable propagation of Z errors as discussed in Sec. IV D. Finally, note the following important difference between the untwisted (without T) and twisted protocols, which is especially important in the $2 + 1\text{D}$ case discussed in the following paragraphs: In the untwisted protocol, the map from the measurement outcomes b_{meas} to the flux configuration b is ambiguous as discussed in Sec. V A. For example, as discussed after Eq. (131), a -1 outcome of a ZZ measurement can contribute to b at two different pairs of faces. In general, only the coboundary

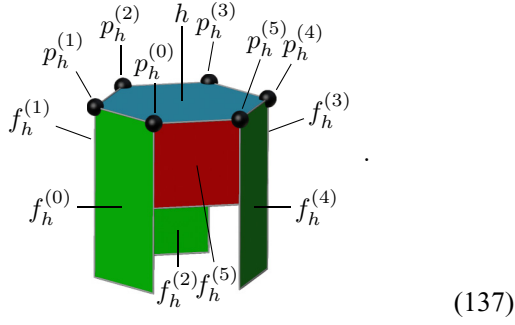
db is relevant for decoding and corrections, and we can make arbitrary changes of the form $b \rightarrow b' = b + d\beta$. In the twisted protocol, there is a unique configuration b associated to b_{meas} , as the nontrivial action in the path integral spoils invariance under changes of the form $b' = b + d\beta$. For example, if we add the $e^{2\pi i/8}$ -phase Z -tensor corresponding to the T gate in Eq. (131), we can only move the $e^{2\pi i/2}$ X -tensor phases to one particular side. Accordingly, the corrections do depend not only on db , but on b itself as $dG = b + \tilde{b}$.

2. Flux corrections in $2 + 1\text{D}$

In the $2 + 1\text{D}$ circuit from Sec. II, we have to apply the analogous corrections. Namely, we need to choose the gauge-fixing configuration G with $dG = b + \tilde{b}$, and apply $T^{\text{sgn}_p \cdot (-1)^{G_p}}$ instead of T^{sgn_p} . However, these corrections now need to be applied while we run the circuit: When we apply the $T^{\text{sgn}_p \cdot (-1)^{G_p}}$ gate in the circuit, we need to choose the value of G_p at the according vertex p . At this time we do not have access to the value of b on all faces of the 3-colex, but only on a subset. Roughly, the subset of faces with a known b value are those whose time coordinate is smaller or similar to that of p . We need to show that it is still possible to choose G_p given these limitations.

In the following, we show that this can be done for our example twisted color circuit from Sec. II. A similar strategy also works for the other examples of twisted color circuits in Sec. VI. Let P_t be the set of 3-colex vertices with the same time coordinate t , located inside a single common time-parallel plane. Let F_t be the set of faces f such that the vertices of f with the largest time coordinate are in P_t . Write $P_{<t} := \bigcup_{t' < t} P_{t'}$, $F_{\leq t} := \bigcup_{t' \leq t} F_{t'}$, and so on. Also, write $G_{<t}$ for G restricted to $P_{<t}$ and zero otherwise, b_t for b restricted to F_t , and so on. We need to choose G_t such that $dG = b + \tilde{b}$ holds on all faces $f \in F_t$. This is because dG does not depend on G_p for any vertex $p \in P_{>t}$. Luckily, at the moment where we apply the $T^{\text{sgn}_p \cdot (-1)^{G_p}}$ gate corresponding to the vertices $p \in P_t$, we do already know the value of b on all faces $f \in F_{\leq t}$. This is because $(b + \tilde{b})_f$ for a face f only depends on the ZZ measurement outcomes performed at the vertices $p \in f$, and the $Z^{\otimes 6}$ measurement outcome of f itself if f is a time-perpendicular hexagon. Let us now concretely describe how we choose G_p on all vertices $p \in P_t$ in our 3-colex: Each vertex $p \in P_t$ is part of one unique time-perpendicular hexagon h . We label the 6 vertices that are part of h by $p_h^{(0)}, \dots, p_h^{(5)}$ in clockwise order. Each face $f \in F_t$ is either equal to some time-perpendicular hexagon h , or shares an edge with h . We label the six faces $f \in F_t$ that share an edge with h by $f_h^{(0)}, \dots, f_h^{(5)}$ in clockwise order, such that both $f_h^{(i)}$ and $f_h^{(i-1 \bmod 6)}$ contain the vertex $p_h^{(i)}$. The following picture

shows the vertices $p_h^{(i)}$ and faces h and $f_h^{(i)}$,



In the example above, all faces belong to the boundary of a γ -colored volume and are colored according to the adjacent volumes. We now choose

$$G_{p_h^{(i)}} = \sum_{0 \leq x < i} (b + \tilde{b} + dG_{< t})_{f_h^{(x)}}. \quad (138)$$

We indeed find that

$$\begin{aligned} & (b + \tilde{b} + dG_{\leq t})_{f_h^{(i)}} \\ &= (b + \tilde{b})_{f_h^{(i)}} + (dG_{< t})_{f_h^{(i)}} + (dG_t)_{f_h^{(i)}} \\ &= (b + \tilde{b})_{f_h^{(i)}} + (dG_{< t})_{f_h^{(i)}} + G_{p_h^{(i+1 \bmod 6)}} + G_{p_h^{(i)}} \\ &\stackrel{(138)}{=} (b + \tilde{b})_{f_h^{(i)}} + (dG_{< t})_{f_h^{(i)}} + (b + \tilde{b} + dG_{< t})_{f_h^{(i)}} = 0 \end{aligned} \quad (139)$$

and

$$\begin{aligned} & (b + \tilde{b} + dG_{\leq t})_h \\ &= (b + \tilde{b})_h + \sum_i G_{p_h^{(i)}} \\ &\stackrel{(138)}{=} (b + \tilde{b})_h + \sum_{i \in \{1, 3, 5\}} (b + \tilde{b} + dG_{< t})_{f_h^{(i)}} \\ &= \sum_{f \in \{h, f_h^{(1)}, f_h^{(3)}, f_h^{(5)}\}} (b + \tilde{b} + dG_{< t})_f \\ &\stackrel{*}{=} \sum_{f \in \Delta_2: \mathbf{c}_f \in \{yb, yg\}, \partial_{v_h, f} = 1} (b + \tilde{b} + dG_{< t})_f \\ &= d(b + \tilde{b} + dG_{< t})_{v_h, r} \stackrel{**}{=} 0. \end{aligned} \quad (140)$$

For the equality labeled $*$, we have assumed that finding G was successful at all previous steps, and we thus have $(b + \tilde{b} + dG_{< t})_f = 0$ for all faces $f \in F_{< t}$. $v_h \in \Delta_3$ denotes the volume whose top face is h . For the equation labeled $**$, we have used that $d(b + \tilde{b}) = 0$ by construction. Summarizing, we have shown that it is possible to choose the gauge fixing configuration G while running the circuit.

3. 2 + 1D: Just-in-time decoding

Since we have to choose G as we go, we also need to choose $\tilde{b}$ as we go. In particular, our choice of G_t at time step t according to Eq. (138) depends on $\tilde{b}_{\leq t}$. At the same time step t , we do not know the full flux configuration b but only the portion that is determined by the measurement outcomes collected so far. This portion is given by $b_{\leq t}$, since the b_f on some face f only depends on the ZZ measurements associated with the vertices of f , as well as the $Z^{\otimes 6}$ measurement associated with f itself if f is a time-perpendicular hexagon. So the challenge for the decoder is to find a low-weight estimate for $\tilde{b}$ under the “causality” constraint that each portion $\tilde{b}_t$ can only depend on $b_{\leq t}$. A solution to this challenge was first proposed in Ref. [22], namely so-called *just-in-time decoding*. In Sec. II G, we have given an intuitive explanation of just-in-time decoding when picturing flux configurations as string patterns. Here, we describe a detailed decoding algorithm using the language introduced since then.

4. Just-in-time algorithm

The just-in-time decoding algorithm is executed once at every time step t . It takes the measured flux configuration $b_{\leq t}$ as input, and computes $\tilde{b}_t$ as output. For convenience and performance, it also remembers the collective outputs $\tilde{b}_{< t}$ from all previous time steps, even though these could in principle be recomputed.

The algorithm proceeds in two steps: The first step is to find a 2-cochain b_{glob} which is defined on the overall 3-colex, such that $(b_{\text{glob}})_{\leq t}$ has low weight, and $db_{\text{glob}} = db_{\leq t}$. In other words, we decode the flux configuration as if we were doing global decoding at time t like in Sec. V A, ignoring all choices of $\tilde{b}$ from previous time steps. The second step is to find a low-weight 2-cochain b_{join} that is only supported on $F_{\geq t}$ such that $db_{\text{join}} = d(\tilde{b}_{< t} + (b_{\text{glob}})_{\leq t})$. In other words, we consider the choices $\tilde{b}_{< t}$ committed in previous time steps, and the current global choice b_{glob} . $\tilde{b}_{< t}$ terminates at time $t - 1$, and b_{glob} terminates at time t . We extend $\tilde{b}_{< t}$ by one time step, such that it terminates at time t in the same way as b_{glob} . After this, the output is given by $\tilde{b}_t := (b_{\text{join}})_t$. The procedure is summarized in Algorithm 1. We defer the actual algorithm for the two hypergraph-matching problems to Sec. V E.

We should confirm that a solution for b_{join} in the second step exists even if we restrict ourselves to $F_{\geq t}$. This is the case because db_{glob} coincides with db at times earlier than t , and $d(\tilde{b}_{< t} + (b_{\text{glob}})_{\leq t})$ is only supported on volumes that contain faces in $F_{\geq t}$. More formally, we observe

$$\begin{aligned} d(\tilde{b}_{< t} + (b_{\text{glob}})_{\leq t}) &= db_{< t} + db_{\leq t} + d(b_{\text{glob}})_{\geq t} \\ &= d(b_t + (b_{\text{glob}})_{\geq t}), \end{aligned} \quad (141)$$

so $b_{\text{join}} := b_t + (b_{\text{glob}})_{\geq t}$ is a valid solution supported on $F_{\geq t}$, though not necessarily a low-weight one.

ALGORITHM 1. Just-in-time flux decoding.

At each time step $t \geq 0$:

Input: $(db)_{\leq t}$.

Output: G_t .

Memorize from previous steps: $\tilde{b}_{<t}$.

- 1: Using Algorithm 2 with open/smooth boundary conditions at time t , find a 2-chain b_{glob} which is defined on the overall 3-colex, such that $(b_{\text{glob}})_{\leq t}$ has low weight, and $db_{\text{glob}} = db_{\leq t}$.
 - 2: Using Algorithm 2 with closed/rough boundary conditions at time t , find a low-weight 2-chain b_{join} that is only supported on $F_{\geq t}$ such that $db_{\text{join}} = d(\tilde{b}_{<t} + (b_{\text{glob}})_{\leq t})$.
 - 3: Set $\tilde{b}_t := (b_{\text{join}})_t$. Set $\tilde{b}_{\leq t} := \tilde{b}_{<t} + \tilde{b}_t$, and memorize for time $t+1$.
 - 4: The output G_t is given by Eq. (137), using the notation from Eq. (136).
-
-

Finally, let us briefly give some intuition for how this algorithm works when imagining b and $\tilde{b}$ as string configurations. Consider a configuration of db consisting of two points at the same time t_0 but spatial separation x , like the bottom two points in Fig. 12. At each time step t with $t_0 \leq t < t_0 + \frac{1}{2}x$, b_{glob} matches each of the two points to the time- t state boundary separately, since this has less weight than matching the two points to each other, $2(t - t_0) < x$. Thus, $\tilde{b}_t$ consists of two strings propagating from the two points in time direction. At the first time step with $t > t_0 + \frac{1}{2}x$, b_{glob} instead consists of a string matching the two points in db . So $\tilde{b}_{<t} + (b_{\text{glob}})_{\leq t}$ now has a form shaped like $\sqcup$ (if time flows upward as in Fig. 12), and b_{join} connects the two end points at time t . After this, $\tilde{b}_{\leq t}$ will be shaped like $\sqcap$.

C. Global decoding of charge configuration

Let us now describe how to decode the charge syndrome, created by Z errors, in our twisted color circuits. This decoding can be done globally, like for the untwisted color circuit in Sec. V A. In fact, we could still decode by solving the same hypergraph-matching problem $\tilde{d}^T \tilde{c} = d^T c$. Even though the residual flux configuration $\tilde{b} + b_E$ effectively decreases the fault distance for Z errors, it is of low weight, so we still expect this method to have a fault-tolerance threshold. We can, however, mitigate the decrease of fault distance to some extent by using the information we get from the just-in-time flux decoding.

Let us assume for a moment that we are told the actual X error configuration b_E , after just-in-time decoding with $\tilde{b}$. Then, the residual flux configuration $\tilde{b} + b_E$, which leads to missing charge detectors around its support, can be treated similar to a heralded erasure. In order to use the heralded information to optimize charge decoding, we simply replace the charge constraint $d^T c = 0$ with the twisted charge constraint in Eq. (104). That is, instead of $d^T \tilde{c} = dc$, we solve the twisted low-weight hypergraph-matching

problem

$$K^{b_E + \tilde{b}} d^T \tilde{c} = K^{b_E + \tilde{b}} d^T c + k^{b_E + \tilde{b}}, \quad (142)$$

where K and k are defined in Eqs. (106) and (107).

In reality, we only know the estimated error $\tilde{b}$, but not the actual error b_E . With this partial information, it is not so clear how to best use it to optimize charge decoding, and we have to resort to some heuristics. First of all, we note that our interpretation of $M^{\tilde{b}}$ in Eq. (105) in terms of gauge variances of the color path integral only makes sense when $db = 0$, since the path integral evaluates to zero otherwise. Hence, we need to choose a heuristic to define $M^{\tilde{b}}$ if $\tilde{d}\tilde{b} \neq 0$. A choice that seems sensible is $(M^{\tilde{b}})_{vv'} = 0$ if $(\tilde{d}\tilde{b})_v \neq 0$, or if $(\tilde{d}\tilde{b})_{v'} \neq 0$. Given such a heuristic, we would proceed in the same way as in the $db = 0$ case: We define the matrix $K^{\tilde{b}}$ via the kernel of $M^{\tilde{b}}$. When $\tilde{b}$ terminates at some volume v ($(\tilde{d}\tilde{b})_v = 1$), then the choice above implies that any twisted charge detector can be truncated at v .

Also in our definition of $k^{\tilde{b}}$ in Eq. (107), the assumption $db = 0$ was essential, since it required a choice of A_0 such that $dA_0 = b$. Again, we propose a heuristic, namely, to use a global decoder to obtain a low-weight $\tilde{b}_{\text{glob}}$ with $\tilde{d}\tilde{b}_{\text{glob}} = db$, and then use A_0 such that $dA_0 = d(\tilde{b} + \tilde{b}_{\text{glob}})$ in order to define $k^{\tilde{b}}$. With these two choices, we then solve the hypergraph-matching problem

$$K^{\tilde{b}} d^T \tilde{c} = K^{\tilde{b}} d^T c + k^{\tilde{b}}. \quad (143)$$

We do not provide a detailed algorithm for how to efficiently solve this hypergraph-matching problem here, but we do provide some ideas in Sec. V E.

D. Decoding in a logical circuit

In this section we describe how decoding works when we use twisted color circuits as logical blocks that realize non-Clifford gates in a logical circuit. Concretely, we assume an end-to-end logical circuit with only classical inputs and outputs, consisting of two kinds of logical blocks: First, we use logical Clifford gates and Pauli measurements that are implemented by 2D CSS-type Abelian topological protocols, for example, by transversal Clifford gates or lattice surgery in the 2D color or surface code. Second, we use logical non-Clifford gates or non-Pauli measurements which are implemented by twisted color circuits, such as the $|T\rangle$ -state measurement from Sec. II. The full physical circuit can be pictured as one overall 3D spacetime diagram whose volumes are labeled by either Abelian syndrome-extraction or twisted color circuits, which can have physical boundaries and are interfaced at domain walls.

The X and Z measurement outcomes of both the Abelian parts of the protocol and the twisted color circuits yield

a flux configuration b and a charge configuration c . b (c) is a configuration of $\mathbb{Z}_2$ -variables on the overall space-time lattice and can be viewed as a 2-cochain (1-chain) of some cohomology theory defined on this spacetime lattice [86]. The different 3D cohomology theories have different boundaries and are coupled at their domain walls.

1. Just-in-time decoding

We need to apply just-in-time decoding to choose the gauge-fixing membrane G inside all volumes of twisted color circuit. A strategy that generally works is to apply just-in-time decoding to the full flux configuration b during all of the protocol, including both the Abelian and the twisted-color-circuit volumes. The resulting G is then a 1-cocycle in the overall cohomology theory defined on all of spacetime. However, G will only have an effect inside the volumes with twisted color circuit, as only these contain the $T^{\text{sgn}_p} \cdot (-1)^{G_p}$ gates. As we will discuss later, this allows us to restrict just-in-time decoding to the vicinity of the spacetime volumes with twisted color circuits, but in general we have to include a large-enough buffer zone of abelian circuits.

2. Global decoding for logical measurements

In the end-to-end logical circuit, the only reason why we need to perform global decoding is to fault-tolerantly deduce logical measurement outcomes from b and c . After choosing global low-weight X and Z error estimates $\tilde{b}$ and $\tilde{c}$, the configuration of logical measurement outcomes is encoded in the cohomology class of $b + \tilde{b}$ and $c + \tilde{c}$. In other words, each logical outcome corresponds to one generating cohomology class, which is usually supported locally on one logical block. Note that we could also use the $\tilde{b}$ resulting from just-in-time decoding instead of a global $\tilde{b}$. However, it is better to use a $\tilde{b}$ from global decoding, because the latter leads to a lower logical error rate as it does not suffer from the causality constraints of just-in-time decoding. Note that if the just-in-time decoded $\tilde{b}$ and the global $\tilde{b}$ disagree on a twisted-color-circuit volume, this indicates that we probably chose the wrong G and already made a logical error. If just-in-time and global decoding disagree on the Abelian volumes, this does not imply a likely logical error, and we get a lower logical error rate if we take the global decoding. We further note that for the optimized decoding of the charge syndrome c in the twisted-color-circuit volumes as described in Sec. VC, we need to use the $\tilde{b}$ from just-in-time decoding, not the one from the global decoding.

We need to perform global decoding at two kinds of times t :

- (1) At the very end of the circuit where the logical measurement outcome is a final classical output, or

- (2) at an intermediate time where a logical measurement outcome is used as a classical control, for an adaptive logical circuit.

In the second case, the value of the according generating (co)homology class must be deducible from $b_{\leq t}$ or $c_{\leq t}$. In this case, we perform the global decoding with a state boundary condition at time t , that is, $b + \tilde{b}$ and $c + \tilde{c}$ are allowed to freely terminate at time t .

3. Computational cost of decoding

While the described decoding procedure has a runtime that is polynomial in the size of the overall circuit, there are two aspects that require a large amount of classical computation resources: First, for a large logical circuit, global decoding in all of spacetime is slow, in particular if we consider decoding algorithms based on minimum-weight perfect matching which scale much worse than linear in the overall lattice size. This can be solved by decomposing the overall spacetime into overlapping blocks and decoding each block separately. As long as the blocks are large enough and have sufficient overlap, the decomposition will not increase the logical error rate by much [2,87].

The second aspect is that just-in-time decoding requires running a global decoder (in fact, two) at every single time step. It may be possible to recycle some information of one global decoding to speed up the global decoding in the next time step, but this is highly nontrivial and depends on the implementation details of the decoder. However, if we decompose spacetime into blocks, then we only need to apply just-in-time decoding inside blocks that contain twisted color circuits. Note that in general, it is not possible to restrict just-in-time decoding to precisely the twisted-color-circuit volumes: If a twisted color circuit has a domain wall with an Abelian circuit that is oriented parallel or diagonal to the time direction in spacetime, then we need to use just-in-time decoding also for the Abelian circuit, at least inside some buffer zone that scales with the fault distance. Note that we do not describe any protocols with such domain walls in this paper, but examples can be found in Refs. [17,22,23].

4. Fault distance

In order to suppress the logical error rate by scaling up a family of protocols, we need to make sure that we proportionally scale up the fault distance of the overall physical circuit. We can define a notion of fault distance as follows: The X or flux (Z or charge) fault distance is given by the lowest weight of a 2-cocycle (1-cycle) of nontrivial (co)homology in the overall spacetime cohomology theory. So the scaling of the fault distance is a purely topological property, depending only on the underlying cohomology theory or path integral.

5. Closure condition

There are some further requirements for a scalable logical error suppression, which come from the just-in-time decoder rather than from the path integral. Accordingly, these requirements are of a geometric rather than topological nature. Consider a domain wall or boundary in the overall circuit that is positioned time-perpendicular. Then such a boundary must not have a cohomology that allows for cocycles in the future volume to freely terminate. This is similar to a condition called *closure condition* in Ref. [22]. More precisely, any Abelian circuit is in the same phase as some number of toric codes. Each domain wall can be classified by a subgroup $H \subset \mathbb{Z}_2^i \times \mathbb{Z}_2^o$, where i (o) is the number of toric codes in the Abelian circuit before (after) the domain wall. We also set $i = 3$ or $o = 3$ for a twisted color circuit (as if it was untwisted), and boundaries are domain walls with $i = 0$ or $o = 0$ on one side. With this, we must not have a time-perpendicular domain wall for which H contains an element of the form $(0, x)$ with $0 \in \mathbb{Z}_2^i$ and $x \in \mathbb{Z}_2^o$.

The reason for this is that such a domain wall creates a random initial configuration of b , which leads to a high-weight $\tilde{b}$, which spoils fault tolerance inside a twisted color circuit. In fact, such domain walls do maintain the macroscopic fault distance if $\mathbb{Z}_2^o$ does not correspond to a twisted color circuit, but only to an Abelian circuit. In this case the domain wall needs to have a macroscopic distance to any volume with twisted color circuit. Concretely, we can turn our $|T\rangle$ -state measurement protocol from Sec. II A into a $|T\rangle$ -state preparation protocol by initializing the 2D color code in a $|+\rangle$ state [88]. This initialization corresponds to a time-perpendicular smooth boundary for the 2D-color-code volume at the bottom of Eq. (1). Thus, as a consequence of the above considerations, we need to keep a macroscopic distance between this smooth boundary and the twisted-color-circuit volume. In other words, after the $|+\rangle$ state initialization, we have to perform 2D-color-code syndrome extraction for a time that scales linear in the fault distance before introducing the domain wall with the twisted color circuit discussed in Sec. II E.

E. Hypergraph-matching algorithms

In order to implement the decoding algorithms from the previous sections in practice, we need efficient subroutines that find approximate-minimum-weight solutions to hypergraph-matching problems related to color cohomology. In this section, we propose such subroutines. At the end of the section we comment on related decoding problems in the literature and on the expected thresholds and possible improvements.

On a high level our hypergraph-matching algorithms works as follows: We first apply the mappings from Sec. III C to map the syndrome color (co-)chains db or $d^T c$ to a triple $\vec{y}$ of cellular chains. Then we solve three

decoding problems of the form $d^{\text{cell}} \vec{x} = \vec{y}$, where d^{cell} is the cellular (co-)boundary map. Since the cellular (co-)boundary is an ordinary graph (instead of a hypergraph), we can use minimum-weight graph matching algorithms [2,89,90] to find the exact minimum-weight solution for $\vec{x}$. Finally, we map the cellular triple $\vec{x}$ back to obtain a color chain $\tilde{b}$ or $\tilde{c}$.

1. Hypergraph matching for fluxes

We start with the algorithm that finds an approximate-minimum-weight $\tilde{b}$ with $d\tilde{b} = db$, on a 3-colex without any boundaries. Up to dealing with the boundaries, this algorithm can be used as a global decoder for the flux syndrome in the untwisted color circuit, but also as a subroutine in every just-in-time decoding step for the twisted color circuit. The algorithm depends on some choices: First, we need to choose a “special” color, which we take to be y . Second, for every y -colored volume v , we need to choose a special ry face $f_r(v) \subset v$, a special gy face $f_g(v) \subset v$, and a special by face $f_b(v) \subset v$. The algorithm starts by defining the color 2-cochain

$$\begin{aligned} \bar{b} := & \sum_v ((db)_{(v,r)}(db)_{(v,g)}f_b(v) \\ & + (db)_{(v,r)}(db)_{(v,b)}f_g(v) + (db)_{(v,g)}(db)_{(v,b)}f_r(v)). \end{aligned} \quad (144)$$

That is, each y volume can have either no flux syndrome points, or a pair (r, g) , (r, b) , or (g, b) of syndrome points. If it has a (r, g) pair, then we add the special $f_b(v)$ face to $\bar{b}$, and analogously for other color pairs. $\bar{b}$ is constructed such that the syndrome of $b + \bar{b}$ is supported only on the non- y colored volumes,

$$(d(b + \bar{b}))_{(v,c)} = 0 \quad \forall c \in \{r, g, b\}, \forall v \in \Delta_3 : \mathbf{c}_v = y. \quad (145)$$

Next, for each color $c \in \{r, g, b\} \subset \mathcal{C}_1$, we apply the chain map M^c from Eq. (58) to obtain a cellular 3-chain $s^c := M^c d(b + \bar{b})$, or

$$(s^c)_v := \begin{cases} (d(b + \bar{b}))_{(v,c)} & \text{if } \mathbf{c}_v \neq c, \\ 0 & \text{if } \mathbf{c}_v = c. \end{cases} \quad (146)$$

Due to Eq. (145), s^c can only be nonzero on the volumes v with $\mathbf{c}_v \notin \{c, y\}$. Then, for each c , we find a minimum-weight cellular 2-chain $\tilde{b}^c$ supported on the $\overline{cy}$ -colored faces, such that $d^{\text{cell}} \tilde{b}^c = s^c$. This can be done using algorithms as in Refs. [2,89,90] on the graph whose vertices are the volumes v with $\mathbf{c}_v \notin \{c, y\}$ and whose edges are the $\overline{cy}$ faces. Finally, we set

$$\tilde{b} = \bar{b} + \sum_{c \in \{r, g, b\}} \tilde{b}^c. \quad (147)$$

The procedure is summarized in Algorithm 2.

ALGORITHM 2. Hypergraph matching for fluxes.Input: Syndrome color 3-cochain db .Output: Low-weight color 2-cochain $\tilde{b}$ such that $d\tilde{b} = db$.

- 1: Determine $\bar{b}$ according to Eq. (143).
- 2: For every $c \in \{r, g, b\}$, set $s^c := M^c d(b + \bar{b})$ (c.f. Eq. (145)).
- 3: For every $c \in \{r, g, b\}$, use matching algorithms [2, 66, 67] to compute a cellular 2-cochain $\tilde{b}^c$ supported on $\bar{c}y$ -colored faces such that $d^{\text{cell}}\tilde{b}^c = s^c$.
- 4: The output is given by $\tilde{b} = \bar{b} + \sum_{c \in \{r, g, b\}} \tilde{b}^c$.

Let us briefly argue why Eq. (147) indeed satisfies $d\tilde{b} = db$. For a generator (v, b) consisting of a g -colored volume v , and the color b (blue), we indeed find

$$\begin{aligned}
 (d\tilde{b})_{(v,b)} &= (d\bar{b})_{(v,b)} + \sum_{f \subset v, \mathbf{c}_f \in \{gr, gy\}} \sum_{c \in \{r, g, b\}} (\tilde{b}^c)_f \\
 &= (d\bar{b})_{(v,b)} + \sum_{f \subset v, \mathbf{c}_f = gr} (\tilde{b}^b)_f \\
 &= (d\bar{b})_{(v,b)} + (d^{\text{cell}}\tilde{b}^b)_v = (d\bar{b})_{(v,b)} + (s^b)_v \\
 &= (d\bar{b})_{(v,b)} + (d(b + \bar{b}))_{(v,b)} = (db)_{(v,b)}. \quad (148)
 \end{aligned}$$

For all other generators, the equation holds analogously, except for $\mathbf{c}_v = y$ where it holds trivially.

2. Hypergraph matching for charges

Let us now describe the algorithm that finds an approximate-minimum-weight $\tilde{c}$ with $d^T\tilde{c} = d^Tc$, on a 3-colex without any boundaries. Up to dealing with boundaries, this algorithm can be used to perform the charge decoding in the untwisted color circuit, but also for a naive suboptimal charge decoding in the twisted color circuit. The algorithm is equivalent to the 3D restriction decoder proposed in Ref. [30]. After choosing a special color to be y , the algorithm proceeds as follows. For each of color pair $\kappa \in \{ry, gy, by\} \subset \mathcal{C}_2$, we use the chain map $\bar{M}^\kappa$ from Eq. (64) to obtain a cellular 3-chain $s^\kappa := \bar{M}^\kappa d^Tc$, or

$$(s^\kappa)_v = \begin{cases} (d^Tc)_v & \text{if } \mathbf{c}_v \in \kappa, \\ 0 & \text{if } \mathbf{c}_v \notin \kappa. \end{cases} \quad (149)$$

Then, for each κ , we find a cellular 2-chain $\tilde{c}^\kappa$ whose support is restricted to the κ -colored faces, such that $d_{\text{cell}}\tilde{c}^\kappa = s^\kappa$. We do this using algorithms as in Refs. [2, 89, 90] on the graph whose vertices are volumes v with $\mathbf{c}_v \in \kappa$ and whose edges are the κ -colored faces. We proceed by considering the sum

$$\tilde{c}^\Sigma := \tilde{c}^{ry} + \tilde{c}^{gy} + \tilde{c}^{by}. \quad (150)$$

Then, on each y -colored volume v , let $\tilde{c}^\Sigma|_v$ denote the 2D color 2-chain obtained from restricting $\tilde{c}^\Sigma$ to the boundary

ALGORITHM 3. Hypergraph matching for charges.Input: Syndrome color 0-chain d^Tc .Output: Low-weight color 1-chain $\tilde{c}$ such that $d^T\tilde{c} = d^Tc$.

- 1: For every $\kappa \in \{ry, gy, by\}$, compute the cellular 3-cocycle $(s^\kappa)_v := (d^Tc)_v \delta_{\mathbf{c}_v \in \kappa}$.
- 2: For every $\kappa \in \{ry, gy, by\}$, use minimum-weight matching [2, 66, 67] to compute a cellular 2-cochain $\tilde{c}^\kappa$ supported on the κ -colored faces, such that $d_{\text{cell}}\tilde{c}^\kappa = s^\kappa$.
- 3: For each y -colored volume v , find a 2D color 1-chain $\tilde{c}_v$ on the boundary 2-colex of v , such that $d_{2D}\tilde{c}_v = (\tilde{c}^{ry} + \tilde{c}^{gy} + \tilde{c}^{by})|_v$.
- 4: The output is given by $\tilde{c} := \sum_{v \in \Delta_3: \mathbf{c}_v = y} \tilde{c}_v$.

2-colex of v . We find a color 1-chain $\tilde{c}_v$ on the boundary 2-colex of v such that $d_{2D}\tilde{c}_v = \tilde{c}^\Sigma|_v$. The choice of $\tilde{c}_v$ does not affect the cohomology class of the final output, and we can find one by Gaussian elimination. The output is then given by the sum of all the $\tilde{c}_v$,

$$\tilde{c} := \sum_{v \in \Delta_3: \mathbf{c}_v = y} \tilde{c}_v. \quad (151)$$

The procedure is summarized in Algorithm 3.

Let us verify that the above indeed satisfies $d^T\tilde{c} = d^Tc$. For a y -colored volume v , we find

$$\begin{aligned}
 (d^T\tilde{c})_v &= (d^T\tilde{c}_v)_v = (d_{\text{cell}}(d_{2D}\tilde{c}_v))_v = (d_{\text{cell}}\tilde{c}^\Sigma)_v \\
 &= (d_{\text{cell}}(\tilde{c}^{ry} + \tilde{c}^{gy} + \tilde{c}^{by}))_v = (s^{ry} + s^{gy} + s^{by})_v \\
 &= 3(d^Tc)_v = (d^Tc)_v. \quad (152)
 \end{aligned}$$

For a g -colored volume v , we have

$$\begin{aligned}
 (d^T\tilde{c})_v &= \sum_{p \in v} \sum_{v' \in \Delta_3: \mathbf{c}_{v'} = y, p \in v'} (\tilde{c}_{v'})_p \\
 &= \sum_{f \subset v: \mathbf{c}_f = gy} \sum_{p \in f} (\tilde{c}_{v'(f)})_p = \sum_{f \subset v: \mathbf{c}_f = gy} (d_{2D}\tilde{c}_{v'(f)})_f \\
 &= \sum_{f \subset v: \mathbf{c}_f = gy} (\tilde{c}^{gy})_f = (s^{gy})_v = (d^Tc)_v, \quad (153)
 \end{aligned}$$

where $v'(f)$ denotes the y -colored volume containing the gy -colored face f . For r and b -colored volumes, the equation holds analogously.

3. Matching with boundaries

After describing decoding algorithms in the bulk, let us sketch how they can be extended to the various types of boundaries that may occur in our protocols, including both physical and state boundaries. As we have discussed in Sec. IV E, each boundary type corresponds to a boundary condition for color cohomology. So the approximate-minimum-weight hypergraph-matching problems that we

need to solve look the same as before, $\tilde{d}\tilde{b} = db$ and $d^T\tilde{c} = d^Tc$, only the (co)boundary operators are that of a different cohomology theory. Let us now consider some examples for boundary conditions from Eq. (IV E) and discuss how the algorithms above can be adapted to incorporate these boundaries.

As a first example, let us consider flux decoding in the presence of an r color boundary. The hypergraph-matching problem is given by the boundary map d_2 in Eq. (122). When choosing $f_r(v)$, $f_g(v)$, and $f_b(v)$ for Eq. (144) for a volume $v \in \Delta_3^{\text{bdin}}$, we need to choose a face in X_2 . We solve the three cellular-cohomology matching problems $d_{\text{cell}}(\tilde{b}^r, \tilde{b}^g, \tilde{b}^b) = (s^r, s^g, s^b)$. Looking at Fig. 24, the corresponding boundary of the cellular cohomology is determined by the subgroup $H = \langle (0, 1, 0), (0, 0, 1) \rangle$. Thus, matching of s^r is with respect to a rough boundary condition, whereas matching of s^g and s^b is with respect to a smooth boundary condition. d_{cell} with rough boundary still defines a graph, so we can directly use algorithms like in Refs. [2,89,90]. d_{cell} with smooth boundary defines a hypergraph, where some edges near the boundary are hyperedges with only a single vertex. Concretely, the hyperedges are the rb -colored (rg -colored) boundary faces, and the missing vertex is the missing X_3 generator (v, g) for the adjacent volume $v \in \Delta_3^{\text{bdout}}$ ((v, b) for the adjacent volume $v \in \Delta_3^{\text{bdout}}$). In this case, we have to add one global vertex to the hypergraph connected to all the single-vertex hyperedges, which turns the hypergraph into a graph. After solving the three minimum-weight graph matching problems, the final result is obtained via Eq. (147) as before.

As a second example, consider a the charge decoding for the same r color boundary. The hypergraph matching problem is given by the boundary map d_0^T in Eq. (122). We compute the cellular 3-cochains s^{ry} , s^{gy} , and s^{by} as in Eq. (149), restricting ourselves to the generators in X_0 . Then we solve the three cellular-cohomology matching problems $d_{\text{cell}}(\tilde{c}^{ry}, \tilde{c}^{gy}, \tilde{c}^{by}) = (s^{ry}, s^{gy}, s^{by})$. The boundary condition for this cellular cohomology is described by the subgroup $H^\perp = \langle (1, 0, 0) \rangle$, orthogonal to the subgroup H in the flux decoding. So s^{ry} is matched with respect to a smooth boundary condition, whereas s^{gy} and s^{by} are matched with respect to a rough boundary condition. Concretely, the missing vertices in the hypergraph for the smooth boundary condition are the r -volumes missing from X_0 in Eq. (122). Finally, when computing $\tilde{c}$ via Eq. (151), $\tilde{c}_v$ is defined on all y -colored volumes of X_0 including these in $\Delta_3^{\text{bdin},y}$ and $\Delta_3^{\text{bdout},y}$. The boundary of the latter two kinds of volumes forms itself a 2-colex with boundary. So in the equation $d_{2D}\tilde{c}_v = \tilde{c}^\Sigma|_v$, d_{2D} is the 2D color coboundary map on a 2-colex with a r color boundary as shown in Eq. (A4).

As a third example, consider flux decoding for the y color boundary. This case is different from the r boundary,

because y coincides with the “special” color in our flux-decoding algorithm. Of course, we could just pick a different special color that is not y . However, if we consider a protocol that includes all 4 different color boundaries such as the logical $\bar{T}$ gate in Eq. (162), then one of the color boundaries must coincide with the special color. The hypergraph-matching problem is given by the boundary map d_2 in Eq. (122), with $c = r$ exchanged by $c = y$. According to Fig. 24, the cellular-cohomology matching problem has a boundary condition corresponding to the subgroup $H = \langle (1, 1, 0), (0, 1, 1) \rangle$. So instead of solving three uncoupled cellular-cohomology matching problems, the three problems are coupled at the y color boundary. The three coupled cellular boundary maps still define a graph (as opposed to a general hypergraph), so we can still apply algorithms like in Refs. [2,89,90].

As a fourth example, consider the subroutine of the just-in-time decoder in Algorithm 1 for finding b_{glob} at every time step. The hypergraph-matching problem is modified by (1) restricting $\tilde{b}$ to $F_{\leq t}$, and (2) removing all (triples of) flux detectors at volumes containing faces in $F_{> t}$. That is, it is given by the coboundary map d_2 of the all-smooth boundary in Eq. (A7). As the name suggests, the associated cellular cohomology corresponds to the full subgroup $H = \mathbb{Z}_2^3$. So the matching of each cellular 3-cochain s^c is with respect to a smooth boundary condition. For each s^c , the matching can be turned into a graph-matching problem by adding one global vertex connected to all the cy -colored faces in $F_{> t}$.

As a fifth example, consider the subroutine of the just-in-time decoder for finding b_{join} . The hypergraph-matching problem is modified by restricting $\tilde{b}$ to $F_{\geq t}$. That is, the hypergraph is the coboundary map d_2 in Eq. (121). The according cellular cohomology on which we match s^c has a rough boundary condition for every $c \in \{r, g, b\}$ and thus corresponds to three separate graphs. Concretely, the graph can be obtained from the boundary-free graph by removing all faces in $F_{< t}$.

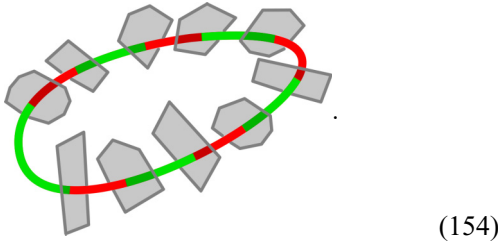
For a general boundary, the hypergraph-matching problem is defined by one of the (co-)boundary operators d_2 or d_0^T in Eqs. (A7), (122), (124), or (121). The corresponding cellular cohomology in which we have to match s^c or s^k is determined by some subgroup H listed in Fig. 24, or its orthogonal complement $H^\perp$. If H or $H^\perp$ is not generated by elements of the form $(1, 0, 0)$, $(0, 1, 0)$, or $(0, 0, 1)$, the three cellular matching problems for s^c or s^k are coupled at the boundary. With one exception, the cellular matching problem can be turned into a graph by adding one global vertex for every smooth component, that is, for every $(1, 0, 0)$, $(0, 1, 0)$, or $(0, 0, 1)$ generator. The exception is the charge decoding for the y color boundary, whose cellular boundary cohomology is determined by $H^\perp = \langle (1, 1, 1) \rangle$. In this case, the cellular boundary faces form hyperedges adjacent to three vertices, and there is no

easy way to turn the hypergraph into a graph. A possible but probably suboptimal solution for this case is to first match as if the boundary condition was $H^\perp = \langle(1, 1, 0)\rangle$ and then add match the third component consistently with this choice.

4. Twisted charge matching

After discussing boundaries, let us now make some comments on solving the optimized twisted hypergraph-matching problem for the charges in Eq. (143), which takes into account the output $\tilde{b}$ of the just-in-time flux decoding. Note that the charge hypergraph-matching decoder proposed above works in principle, and the following discussion is only to further improve its performance.

First of all, we note that the twisted charge constraint in Eq. (143) is independent of a choice of generating constraints, or twisted charge detectors. However, for designing an efficient hypergraph-matching algorithm, it is essential to understand the structure of the twisted constraints, and to choose some “canonical” set of generators. Let us now study the structure of the twisted constraints at hand of some examples. First of all, we recall that every volume without $b_f = 1$ faces defines a twisted constraint, and it is natural to take all these volumes as generators, like in the untwisted case. The remaining volumes $b_{\text{vol}} \subset \Delta_3$ with at least one $b_f = 1$ face are no longer individual generators, but combine to larger generators. Now, consider a simple flux configuration that consists of an isolated loop of rg faces, schematically depicted as



In that case, $M_{v,v'}^b = 1$ if v and v' are neighboring volumes along the loop and 0 for other pairs of volumes. The volumes with flux faces together define two combined generating twisted constraints which we can label by 0 and 1,

$$K_{0v}^b := (b_{\text{vol}})_v \delta_{\mathbf{c}_v=r}, \quad K_{1v}^b := (b_{\text{vol}})_v \delta_{\mathbf{c}_v=g}. \quad (155)$$

In other words, the generator 0 consists of all red volumes on the flux loop, and the generator 1 consists of all green volumes. We further find

$$k_0^b = k_1^b = 0. \quad (156)$$

For this simple flux configuration b , the twisted hypergraph matching still works in the same way as for the untwisted

case, apart from the following modification: In each of the three graph-matching problems, we simply unify all volumes of K_{0v}^b into a single vertex. We do the same for all volumes in K_{1v}^b . When unifying vertices, the graph remains a graph. The method works for any flux configuration b that consists of loops of faces, each for a different color pair, which are sufficiently isolated. Note that if two loops are isolated but linked, we can get $k_x^b = 1$ for some generators x due to the linking charge phenomenon [51].

In contrast, a generic flux configuration b does not consist of isolated color-pair loops, but looks instead like a network of strings. We do not know how to choose a natural set of twisted charge detectors for a generic flux configuration, and how to incorporate these twisted charge detectors into the hypergraph-matching algorithm. Note that for every connected component of b , there is one combined twisted charge constraint for every color pair κ , consisting of all volumes v with $\mathbf{c}_v \in \kappa$. Up to three of these constraints are independent. However, a connected component can give rise to more than three linearly independent charge constraints, including ones involving volumes of more than two colors. We leave a more detailed analytical study of the twisted charge constraints for a generic flux configuration and their influence on the Z -error threshold to future work.

Finally, we remark that we may be able to improve the threshold we get from decoding with the untwisted charge constraint, without using the full precise twisted charge constraints. For example, we could simply remove all untwisted detectors of the volumes in b_{vol} , without adding any new combined detectors. Or, we could only add the combined detector for every color pair κ at every connected component, as discussed above.

5. Expected and ideal performance

Let us finally speculate what thresholds we could expect for the untwisted color circuit using the decoders above, and how they might compare to the threshold for an ideal maximum-likelihood decoder [91]. Our (untwisted) charge decoding problem is equivalent to that for the $3 + 1D$ color code for Z errors in the absence of measurement errors. Reference [92] uses a statistical-mechanics mapping and Monte-Carlo simulation to estimate the threshold for a maximum-likelihood decoder under iid on-site Z noise to be $\sim 1.9\%$, which is similar to that of the $3D$ toric code at $\sim 3.3\%$. This coincides with the maximum-likelihood threshold of our untwisted color circuit under a phenomenological noise model where every vertex in the underlying 3-colex experiences one potential Z error. Unfortunately, there are no known efficient decoding algorithms that come close to this ideal threshold. Reference [19] benchmarks the $3D$ restriction decoder from Ref. [30] on a $3D$ color code with boundaries and on-site Z noise and finds a threshold of 0.8% which is only half

of the maximum-likelihood threshold of 1.9%. For a more realistic circuit-level noise model, one would expect the maximum-likelihood threshold to be a factor of the order of 2–5 smaller. One reason for this is that in the 3 + 1D on-site or 2 + 1D phenomenological noise model, each Z error contributes to c at a single vertex, but for circuit-level noise in our untwisted color circuit multiple error locations may contribute to c at a vertex.

Our flux-decoding problem is equivalent to the X syndrome fixing in the single-shot decoding of the 3D color code. We are not aware of any results on the maximum-likelihood threshold for this decoding problem in the literature, but we expect that it will be of a similar order of magnitude to that of the charge decoding at $\sim 1.9\%$ [93]. Again, efficiently implementable decoders for our flux-decoding problem achieve a much lower threshold than the (expected) ideal one. Reference [94] benchmarks a clustering-based decoder for on-site X noise and finds a threshold of 0.3%. Another decoding strategy for the same problem was sketched in Appendix F.5 of Ref. [82], but not benchmarked. We hope that our proposed flux decoder has a somewhat higher threshold under this noise model, but we do not expect it to come close to the (unknown) maximum-likelihood threshold.

More generally, designing efficient decoders for decoding problems related to the 2D or 3D color code with thresholds close to the maximum-likelihood threshold is an ongoing topic of research. For the measurement-error-free 2D color code, this goal has been achieved using the restriction decoder in Ref. [30]: Under on-site X or Z noise, the restriction-decoder threshold of 10.2% is very close to the maximum-likelihood threshold of 11% [95]. Other decoders for other colorful decoding problems (including higher dimensions) have been proposed [19,30,53,82,83] and partially benchmarked but perform significantly worse than a maximum-likelihood decoder.

VI. MORE EXAMPLES OF TWISTED COLOR CIRCUITS

In this section we briefly provide some further examples of twisted color circuits.

A. $x + y$ -direction in bitruncated cubic lattice

For the first example, it turns out to be more convenient to consider a different geometric realization of the 3-colex in Sec. II. That is, the 3-colex is combinatorially equivalent to the *bitruncated cubic honeycomb* [38] as shown in Fig. 16. Note that the Poincaré dual 3-triangulation to this lattice is a body-centered cubic lattice, consisting of two superimposed cubic lattices (one shifted by $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$), with additional edges between nearest-neighbor vertices from the two different lattices. We choose to work with the basis vectors x , y , and z of this dual cubic lattice as a coordinate system. The first example circuit we consider is obtained

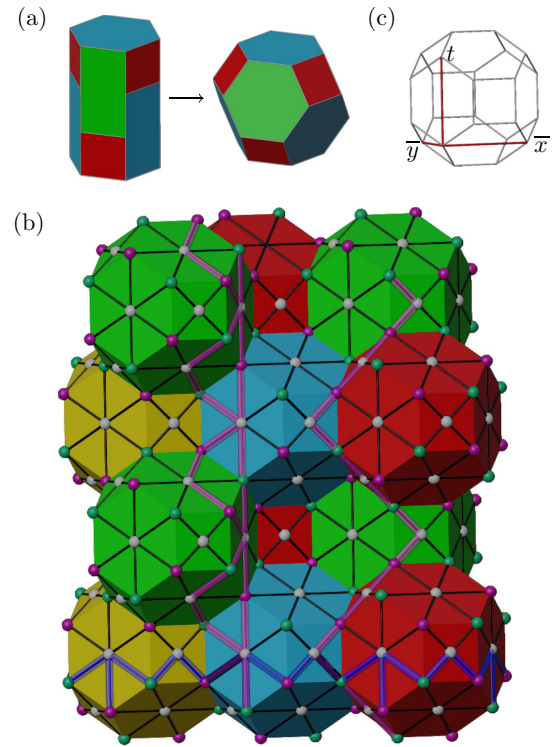


FIG. 16. (a) The hexagon-prism lattice from Sec. II is related to a bitruncated cubic lattice by deforming the hexagon prisms into a more “sphere-like” shape. (b) Snippet of the 3-colex in the bitruncated cubic geometry, together with the color path integral. Three representative qubit worldlines are marked in purple, and one qubit timeslice is shown in blue. (c) Coordinate system with space and time coordinates.

from traversing the path integral in the $t = x + y$ direction in this coordinate system. Figure 16 shows a snippet of the corresponding ZX diagram, as well as three representative qubit worldlines.

From the qubit timeslices and qubit worldlines shown in Fig. 16, we can read off how the qubits are distributed over a two-dimensional spatial lattice as shown in Fig. 17.

Next, we discuss how to locally apply rewrite rules to split some of the ZX tensors, and regroup them, in order to turn the path integral into a circuit. There is one 6-index X -tensor at every time-parallel hexagon, which we split into two 4-index X tensors,

$$\begin{array}{c} \text{Diagram of a 6-index } X \text{-tensor (a hexagon with 6 lines)} \end{array} = \begin{array}{c} \text{Diagram of two 4-index } X \text{-tensors (two hexagons sharing a diagonal line)} \end{array} \quad (157)$$

As shown, this yields two M_{XX} -measurements. There is one 6-index X -tensor at every time-diagonal hexagon, which we split into one 4-index X tensor and two 3-index

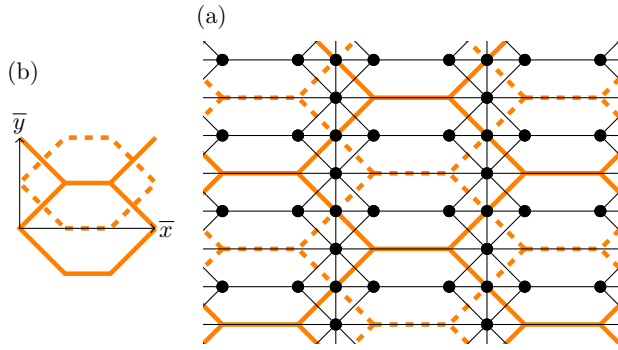


FIG. 17. (a) Qubits (shown as dots) distributed over the 2D spatial lattice (shown as orange lines). The 2D lattice is obtained by projecting the 3-colex along the t direction onto the perpendicular $\bar{x}$ and $\bar{y}$ directions, which yields two superimposed stretched hexagonal lattices. (b) Coordinate system, obtained from projecting the one in Fig. 16.

X -tensors,

$$\text{Vertex} = \text{3-index } X\text{-tensor} \otimes \text{3-index } X\text{-tensor} \otimes M_{XX} \quad (158)$$

As depicted, it gives rise to a M_{XX} measurement and contributes half of two CX gates. The 4-index X -tensor at each time-parallel square simply becomes a M_{XX} measurement. The 4-index X -tensor at each time-diagonal square is split into two 3-index X -tensors,

$$\text{Vertex} = \text{3-index } X\text{-tensor} \otimes \text{3-index } X\text{-tensor} \quad (159)$$

As depicted, each of these 3-index X -tensors contributes half of one CX gate in the circuit. Next, there are three types of vertices in the 3-colex: Type-1 vertices are adjacent to two time-parallel hexagons [one in the past (0) and one in the future (1)], two time-diagonal squares (2 and 3), as well as two time-diagonal hexagons (4 and 5). The 6-index Z -tensor at each type-1 vertex p is split into four phaseless 3-index Z tensors, and one 2-index Z tensor with a $\text{sgn}_p 2\pi/8$ phase,

$$\text{Vertex} = 3 \text{ 3-index } Z\text{-tensors} \otimes 2 \text{ 3-index } Z\text{-tensors} \otimes 2\text{-index } Z\text{-tensor} \quad (160)$$

As depicted, each of these 3-index Z -tensors contributes the control part of one CX operation in the circuit. The other halves of the CX gates are these at the according time-diagonal hexagons and time-diagonal squares in Eqs. (158) and (159). Type-2 vertices are adjacent to one past time-parallel square (0), two past time-parallel hexagons (1 and 2), one future time-diagonal square (3), and two future time-diagonal hexagons (4 and 5). The according 6-index tensor is split into two 4-index tensors, and one phase- $\text{sgn}_v 2\pi/8$ 2-index Z -tensor,

$$\text{Vertex} = 2 \text{ 4-index tensors} \otimes 2\text{-index } Z\text{-tensor} \quad (161)$$

As shown, it gives rise to two M_{ZZ} measurements and one T^{sgn_v} -gate. Type-3 vertices are the same as type-2 vertices after reflecting along the t direction. They are split up in the same way as type-2 vertices in Eq. (161). All in all, the circuit consists of

- (a) two M_{XX} -measurements at every time-parallel hexagon,
- (b) one M_{XX} -measurement at every time-parallel square,
- (c) two M_{ZZ} -measurements at every type-2 and type-3 vertex,
- (d) one CX gate for each pair of a type-1 vertex and adjacent time-diagonal hexagon or time-diagonal square.

In order to construct the circuit, we need to read off from Fig. 16 in which order these operators act on the according qubit worldlines. The resulting circuit is shown in Fig. 18 and has the qubit connectivity shown in Fig. 17. Notably, the circuit already naturally involves only 2-qubit measurements and CX gates. The spacetime unit cell of the circuit is that of the uncolored 3-colex and can be chosen as spanned by $t + \bar{y}$, $2\bar{x}$, and $2\bar{y}$. This unit cell is the one used in Fig. 18. A shift by $t + \bar{y}$ swaps the vertex signs, which is why we have to exchange T and T^{-1} in addition to shifting by $\bar{y}$. Note that a shift by $2t$ also leaves the 4-coloring of the 3-colex invariant, so the circuit already implements the logical identity already after two of the rounds shown in Fig. 18.

B. x direction in bitruncated cubic lattice

As a next example, let us take the same bitruncated cubic lattice from Fig. 16 and traverse it in the $t = x$ direction. The color path integral on this lattice with representative qubit worldlines and qubit timeslices is shown in Fig. 19.

There are three types of faces: (1) Hexagons, all of which are neither perpendicular nor parallel to t , (2) time-perpendicular squares, (3) time-parallel squares (oriented

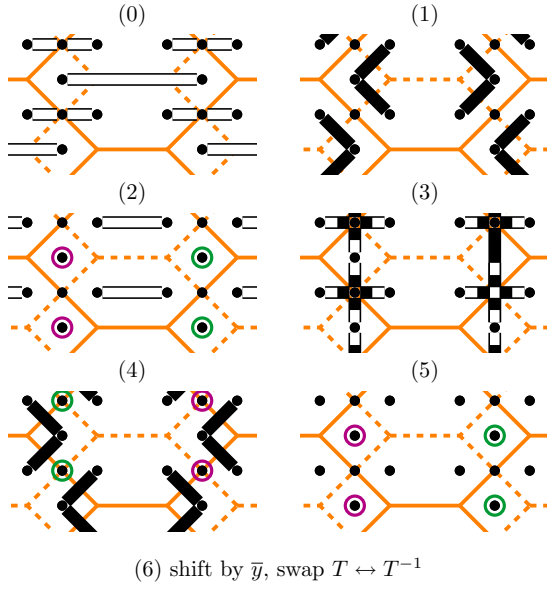


FIG. 18. (0)–(5): Twisted color circuit derived from the ZX diagram, time direction, and qubit worldlines in Fig. 16. The symbols for Pauli measurements, CX gates, and T gates are the same as in Fig. 3. Some of the time steps above contain multiple gates or measurements that act on the same qubits. We can draw these overlapping gates because they commute, and thus the order in which we apply them does not matter. However, in practice, it may not be possible to apply these gates simultaneously, and we will have to distribute them over multiple steps.

in a diamond-like fashion with respect to time). There are two types of vertices: (1) “Horizontal” vertices that are adjacent to a time-perpendicular square, and (2) “vertical” vertices that are not. Horizontal vertices have half-integer t -coordinates $\{\frac{1}{2}k\}_{k \in \mathbb{Z}}$ with respect to the coordinate system in Fig. 19, and vertical vertices have t -coordinates $\{\frac{1}{4} + \frac{1}{2}k\}_{k \in \mathbb{Z}}$. The circuit consists of the following operators and measurements:

- (a) One M_{XX} measurement at every hexagon,
- (b) one $M_{Z^{\otimes 4}}$ measurement at every time-perpendicular square,
- (c) one CX gate and one M_{ZZ} measurement at every horizontal vertex,
- (d) four CX gates at every vertical vertex, and
- (e) one T^{sgn_p} gate at every vertex p .

Let us now describe how the tensors of the ZX path integral are reshaped to give rise to the operators above. The X -tensor at a hexagon is traversed by two qubit worldlines, is split up like in Eq. (158). It yields the M_{XX} measurement of the hexagon and contributes the X -tensors of the CX gates at the two adjacent vertical vertices. The X -tensor at a time-perpendicular square is not traversed by any qubit worldlines and yields the central X -tensor of

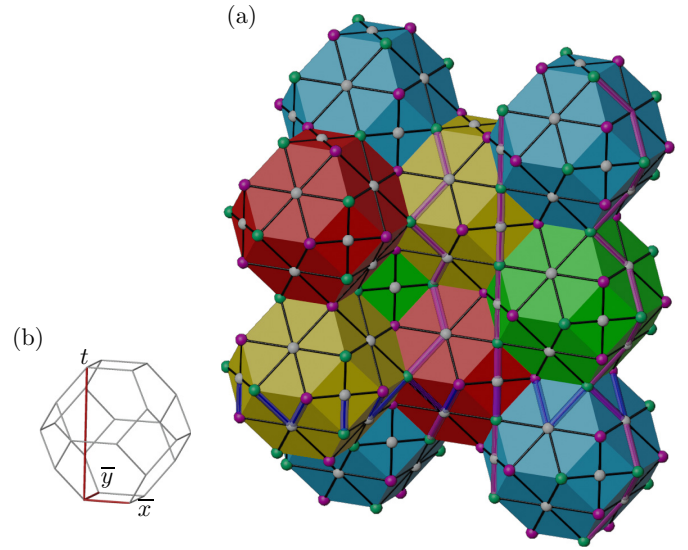


FIG. 19. (a) Color path integral on a snippet of the bitruncated cubic lattice. Three representative qubit worldlines for the chosen t direction are marked in purple. One representative qubit timeslice is marked in dark blue. (b) Coordinate system including the time coordinate $t = x$, and two orthogonal spatial coordinates $\bar{x}$ and $\bar{y}$.

the $M_{Z^{\otimes 4}}$ measurement at the square. The X -tensor at a time-parallel square is traversed by one qubit worldline. It contributes the X -tensor of the two CX gates at the adjacent horizontal vertices. The Z -tensor at a horizontal vertex is traversed by two qubit worldlines. It is split up like in Eq. (17) and gives rise to the M_{ZZ} measurement at the vertex, one Z -tensor of the $M_{Z^{\otimes 4}}$ measurement at the adjacent time-perpendicular square, the Z -tensor of the CX gate at the adjacent time-parallel square, and the T^{sgn_p} gate at the vertex. The Z -tensor at a vertical vertex is traversed by one qubit worldline. It gives rise to the Z -tensor of the four CX gates at the vertex, and the T^{sgn_p} gate at the vertex.

With this, we can write down the circuit by reading off from Fig. 19 in which order the gates and measurements above act. The resulting circuit is shown in Fig. 21 and the required qubit connectivity is shown in Fig. 20.

It suffices to write down the circuit for one spacetime unit cell. We choose this unit cell to be spanned by $4\bar{x}$, $2\bar{x} + 2\bar{y}$, and $\frac{1}{2}t + 2\bar{x}$. Note that a shift by $\frac{1}{2}t + 2\bar{x}$ flips the vertex signs, which is why we need to swap T and T^{-1} in step (6) of Fig. 21.

C. Chamfered cubic lattice

In this section, we briefly consider a combinatorially different 3-colex. It consists of two kinds of volumes: (1) smaller cubes, and (2) larger cubes whose edges have been truncated, yielding a polyhedron with square and hexagon faces called a *chamfered cube*. The 3-colex with the associated ZX color path integral is shown in Fig. 22. The spatial

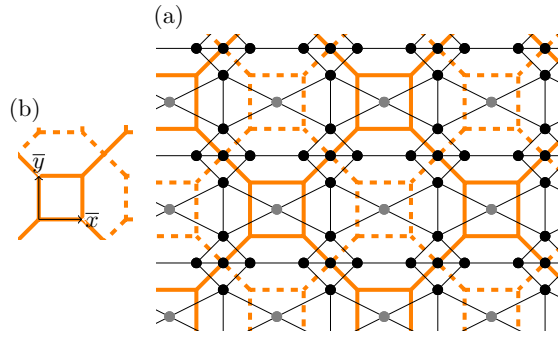


FIG. 20. (a) Projection of the 3-colex along the t direction in orange: It consists of two 4-8-8 lattices, one of which we draw dashed. Qubits and their connectivity are depicted in black. Gray dots represent ancilla qubits for performing the weight-4 Pauli measurements. (b) Coordinate system.

projection and qubit distribution is combinatorially equivalent to the one shown in Fig. 20. The circuit is similar to Fig. 21, with the difference being that steps (1)–(3) are repeated a second time (with $T \leftrightarrow T^{-1}$ exchanged).

D. Circuits for other boundaries

In the previous examples, we have restricted ourselves to the bulk circuits for simplicity. Here, we will describe additional boundary conditions for twisted color circuits,

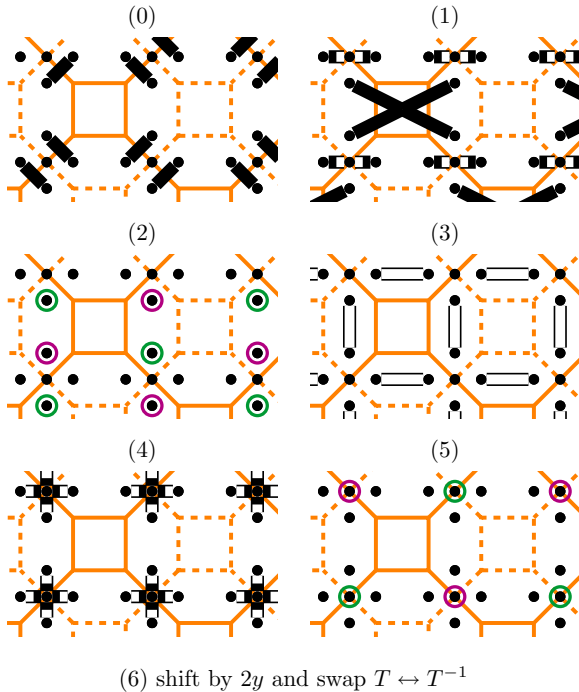


FIG. 21. Twisted color circuit derived from the 3-colex, time direction, and qubit worldlines shown in Fig. 19. Symbols for gates are the same as in Fig. 3. Some steps contain multiple operations which overlap on qubits, but commute.

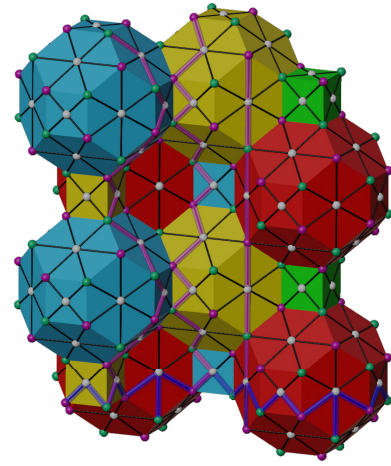
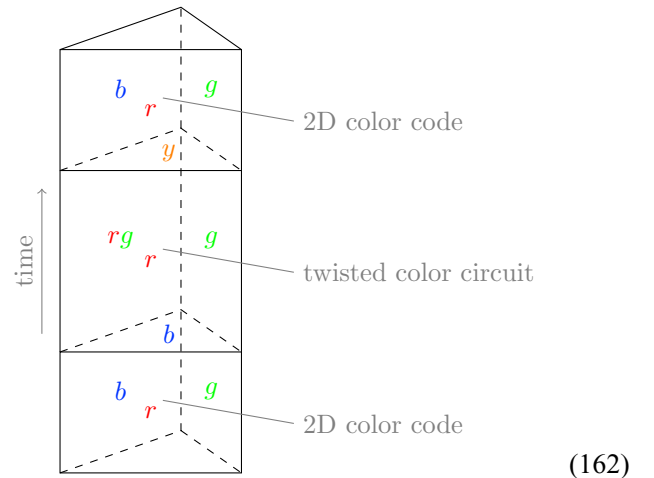


FIG. 22. ZX color path integral on a snippet of the chamfered cubic 3-colex. Time goes upward. Two representative qubit worldlines are marked in purple, and one representative qubit timeslice is marked in dark blue.

for the geometry considered in Sec. II. In Sec. IID, we have only shown how to implement color boundaries of the color path integral. Let us now briefly describe how to implement the other types of boundary conditions discussed in Sec. IVE, namely the double-color and all-rough boundaries. Figure 23 shows the according ZX diagrams, along with representative qubit worldlines. Deriving the according circuits is straightforward using the methods developed in this paper.

Using the 2-color boundary, we can also obtain a gate for a unitary T gate (instead of a $|T\rangle$ -state measurement). The according spacetime diagram (cf. Ref. [17]) for the global protocol is as follows:



(162)

The twisted color circuit in the middle has two color boundaries and also one double-color boundary. The input and output domain walls couple the 2D color code with two different color boundaries. It should be noted that at the bottom domain wall, the 3-colex restricts to a 2-colex

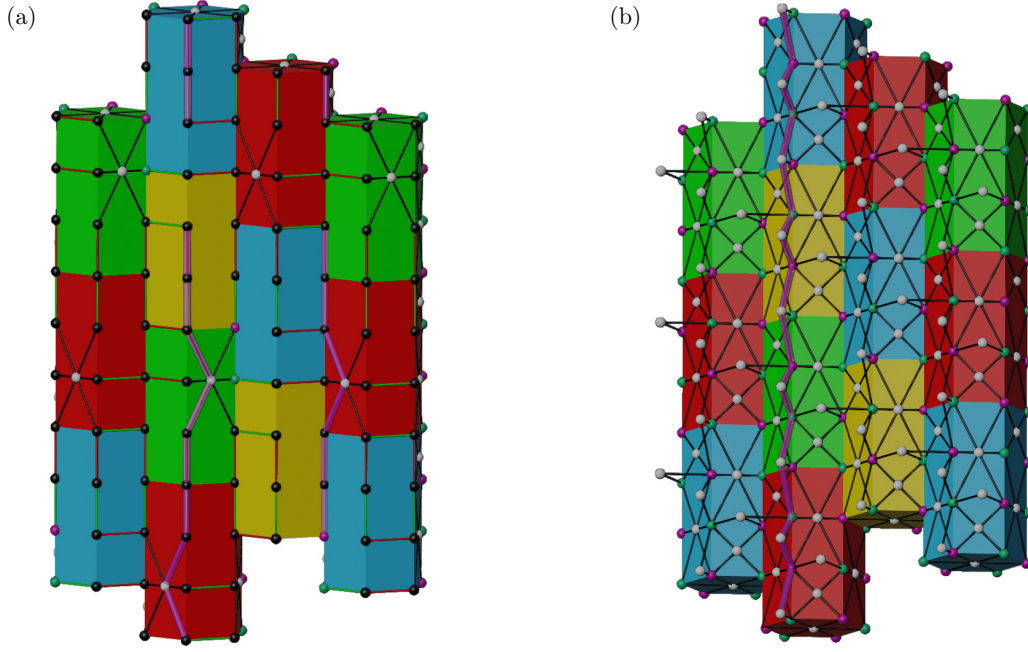


FIG. 23. ZX diagrams for the double-color boundary (a) and all-rough boundary (b) of the color path integral. The Z and X -tensors correspond the generators in X_1 and X_2 of the according cohomology theories in Eq. (124) for the double-color boundary and Eq. (121) for the all-rough boundary. The bonds correspond to the coboundary map d_1 . Note that in (b) we should have put a 2-index X -tensor at every $\bar{r}$ and $\bar{g}$ -colored edge, but a 2-index X -tensor is the same as a bond without tensor. One representative qubit worldline has been marked in purple in each [(a) and (b)].

with colors r , g , and y , which we have to identify with r, g, b .

VII. DISCUSSION

A. Summary

We have constructed *twisted color circuits*, a new class of fault-tolerant scalable planar circuits that realize logical non-Clifford gates on the 2D color code. The circuits are natively described in spacetime by a *color path integral* and can be derived from an arbitrary 3-colex by traversing it in an arbitrary time direction. We have constructed physical boundaries of the twisted color circuits, and domain walls with the 2D color code, to realize various logical non-Clifford gates, using spacetime geometries presented in Ref. [17]. These include a logical $|T\rangle$ -state preparation via XS measurement, a logical unitary T gate, and a logical CCZ gate.

We have given a formulation of the color path integral as a $2 + 1$ D lattice gauge theory, which is based on a cohomology theory on 3-colexes that we call *color cohomology*. We have directly proven the gauge invariance of the underlying action, and its equivalence to a known Dijkgraaf-Witten theory, or twisted quantum double. This establishes that the path integral represents a non-Abelian topological phase, which coincides with that of the (untwisted) Kitaev quantum double for the dihedral group D_4 . We have related the flux and charge defects that

are used for the decoding of our circuits to the anyons of the non-Abelian phase.

Further, we have described the decoding of our circuits in detail, consisting of a *just-in-time* decoder for the X errors and a global decoder for the Z errors. We have shown how to concretely implement the ideas in Ref. [22] for just-in-time decoding by using two global decoders at every time step. We have made extensive use of color cohomology to construct such efficient, matching-based global decoders, extending previous work on color-code decoders [30]. In particular, we have proposed a decoder for the X errors, which could also be used for single-shot decoding of the 3D color code. We have also described how knowledge of the just-in-time X corrections can be used to optimize the global decoding for Z errors.

B. Outlook

The most important remaining open task is to benchmark our circuits, in order to obtain a better estimate for realistic fault-tolerance thresholds of our protocols under circuit-level noise. While this is beyond the scope of this paper, we have established the essential theoretical tools for simulating the logical performance under local Pauli- X and Z noise: Our analysis of flux and charge defects in the color path integral allows us to correctly sample syndrome configurations and to determine the correctness of the decoder, despite the non-Clifford T gates in the circuit.

We believe that the steps we have taken have potential for greatly improving the modest logical performance of the protocols found in simulations so far [24,96]. Two reasons that indicate the possibility of an improved performance are (1) the comparatively low complexity of our circuits, (2) our proposed matching-based decoder which is more fine-tuned to the underlying lattice microscopics than previously used RG-based decoders as in Refs. [23,24].

A second open question is to which extent the methods in this paper carry over to nongeometrically local quantum low-density parity check (qLDPC) codes. There is a variety of recent constructions for transversal CCZ gates in qLDPC codes based on cup products or triple-intersection forms [97–100]. The equivalence that we establish between color and cellular cohomology can be used to translate between the transversal CCZ gate of three 3D toric codes and the transversal T gate of the 3D color code. It will be interesting to see if this cohomological perspective can be used to find transversal logical T gates in more general qLDPC codes.

ACKNOWLEDGMENTS

We would like to thank M. Davydova, D. Williamson, M. Webster, and B. Brown for collaboration on Ref. [17] and B. Brown and M. Vasmer for feedback on an earlier draft of this manuscript. A.B. was supported by the U.S. Army Research Laboratory and the U.S. Army Research Office under Contract/Grant No. W911NF2310255 and by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, and the Co-design Center for Quantum Advantage (C2QA) under Contract No. DE-SC0012704. J.C.M. is supported by the DFG (CRC 183).

DATA AVAILABILITY

There are no publicly available research data or software supporting this manuscript. Requests for further information or data should be sent to the authors.

APPENDIX A: PATH-INTEGRAL BOUNDARIES: DERIVATION

In this appendix, we derive the boundaries stated in Sec. IV E in a detailed and pedagogical way. We start with boundaries of the 2D color path integral and then generalize to three dimensions. The boundaries of the color path integral from which we construct the twisted color circuits only make use of the coboundary map d_1 of color cohomology. However, if we want to describe the decoding problem in terms of color cohomology we have to also define d_0 and d_2 in the presence of boundaries. The boundary maps d_{-1} and d_3 are less natural (and also not locally exact), especially in the presence of boundaries, so we will not define these.

1. 2D boundaries

We start by discussing some 1D physical boundaries for the 2D color path integral in Eq. (70). These provide a useful exercise to get some intuition for the construction and are related to the boundaries of the 2D color code used in Sec. II E. Geometrically, the boundary is defined along any path of edges of the 2-colex. We will use the following notation:

- (a) Δ_i denotes the set of i -cells of the 2-colex including its boundary edges and vertices.
- (b) $\Delta_0^{\text{bd}} \subset \Delta_0$ ($\Delta_1^{\text{bd},c} \subset \Delta_1$) denote the sets of vertices and edges (of color c) that form the boundary.
- (c) $\Delta_2^{\text{bdin}} \subset \Delta_2$ denotes the set of faces adjacent to some boundary edge.
- (d) $\Delta_2^{\text{bdout}} \not\subset \Delta_2$ denotes the “phantom” faces that are adjacent to a boundary edge on the outside of the boundary.
- (e) For every color c , let $\Delta_0^{\text{bdout},c}$ ($\Delta_0^{\text{bdin},c}$) denote the set of boundary vertices that are adjacent to one single face in Δ_2^{bdout} (Δ_2^{bdin}), which has of color c .

2. 2D state boundary

The most naive way to extend 2D color cohomology to the boundary is by restricting the generator sets and coboundary maps of some larger 2-colex-without-boundary containing the 2-colex-with-boundary,

$$X_0 := \Delta_2, \quad X_1 := \Delta_0, \quad X_2 := \Delta_2. \quad (\text{A1})$$

This does not define a physical boundary for the path integral (or, at least not a topological one), since it lacks the local exactness condition discussed in Sec. III B. That is, there are locally supported 1-cocycles that are not 1-coboundaries, namely one for every face in Δ_2^{bdout} . Rather than a physical boundary, the above can be viewed as a state boundary of the path integral as discussed in Sec. IV A. It also does not define a topological boundary for the 2D color code, where the lack of local exactness implies low-weight logical operators.

3. 2D all-smooth boundary

Local exactness can be regained by trivializing each local 1-cocycle by adding an according generator to X_0 ,

$$\begin{aligned} X_0 &:= \Delta_2 + \Delta_2^{\text{bdout}}, & X_1 &:= \Delta_0, \\ X_2 &:= \Delta_2, \end{aligned} \quad (\text{A2})$$

where $+$ denotes set union. The coboundary maps look the same as for the bulk.

The chain complex does define a topological boundary for the 2D untwisted color path integral. However, it does not yield a boundary for the 2D (twisted) color path integral, since there is no way to extend the action in Eq. (71) to the boundary in a gauge-invariant way. One can see this by mapping the boundary to a boundary for cellular cohomology as shown at the end of this appendix: It corresponds to the full subgroup of $\mathbb{Z}_2^2$, and the action is a group 2-cocycle that is nontrivial on this subgroup. The chain complex also yields a boundary for the 2D color code, which represents the same boundary topological phase as the Pauli- X boundary [101], though it is defined in a geometrically slightly different way. In analogy to the path-integral action being incompatible with this boundary, the transversal $\bar{S}$ gate of the 2D color code cannot be extended to this boundary.

4. 2D all-rough boundary

Instead of adding generators to X_0 , we can also achieve local exactness by adding generators to X_2 ,

$$\begin{aligned} X_0 &:= \Delta_2, & X_1 &:= \Delta_0, \\ X_2 &:= \Delta_2 + \Delta_2^{\text{bdout}}. \end{aligned} \quad (\text{A3})$$

This chain complex defines a boundary for both the untwisted and twisted 2D color path integral. The action looks the same as in Eq. (71), where the summation runs over all vertices p including these inside the boundary. The analog 2D-color code boundary is the Pauli- Z boundary [101].

5. 2D color boundary

There is one additional boundary cohomology that is in between the all-smooth and all-rough boundaries: We can define this boundary for each color c , but will set $c = r$ without loss of generality below:

$$\begin{aligned} X_0 &:= \Delta_2 - \Delta_2^{\text{bdin},r} + \Delta_2^{\text{bdout},g} + \Delta_2^{\text{bdout},b}, \\ X_1 &:= \Delta_0, \\ X_2 &:= \Delta_2 + \Delta_1^{\text{bd},\bar{r}}, \end{aligned} \quad (\text{A4})$$

where $-$ denotes set subtraction. d_1^T maps every edge $e \in \Delta_1^{\text{bd},\bar{r}}$ to both of its endpoints, otherwise the coboundary maps are like in the bulk. Note that after adding $\Delta_1^{\text{bd},\bar{r}}$ to X_2 we must exclude $\Delta_2^{\text{bdin},r}$ from X_0 , otherwise we would have $(d_1 d_0)_{ef} = 1$ for an r boundary face $f \in X_0$ and an $\bar{r}$ boundary edge $e \in X_2$ sharing a vertex.

The above chain complex yields a boundary for the untwisted as well as for the twisted color path integral. In the twisted case, the action in Eq. (71) is extended to the

boundary as follows:

$$\begin{aligned} S_{2D}^{\text{bd},r}[A] &= \frac{1}{4} \sum_{p \in \Delta_0 - \Delta_0^{\text{bdin},r}} \text{sgn}_p \bar{A}_p \\ &= \frac{1}{4} \sum_{p \in \Delta_0 - \Delta_0^{\text{bd}} + \Delta_0^{\text{bdout},r}} \text{sgn}_p \bar{A}_p. \end{aligned} \quad (\text{A5})$$

The second equation holds because $(dA)_e = 0$ for every edge $e \in \Delta_1^{\text{bd},\bar{r}}$, so we have $A_{p_0} = A_{p_1}$ for the two endpoints p_0 and p_1 of e , and their action cancels, $\frac{1}{4}(\text{sgn}_{p_0} \bar{A}_{p_0} + \text{sgn}_{p_1} \bar{A}_{p_1}) = 0$. It suffices to test gauge invariance for α consisting of a single X_0 generator f . For faces $f \in \Delta_2^{\text{bdin},g}$ or $f \in \Delta_2^{\text{bdin},b}$, the action on the adjacent vertices is the same as for any bulk face, if we use the first line in Eq. (A5). For faces $f \in \Delta_2^{\text{bdout},g}$ or $f \in \Delta_2^{\text{bdout},b}$, the action is zero if we use the first line in Eq. (A5).

The chain complex also defines a boundary for the 2D color code, which we use in Sec. II in Fig. 7. This boundary is in the same phase as the *color boundary* [4,101], though defined in a microscopically slightly different way. Further, the phases in Eq. (A5) of the form $e^{2\pi i(1/4) \text{sgn}_p \bar{A}_p}$ define a transversal $\bar{S}$ gate for the 2D color-code boundary.

6. 3D boundaries

Let us continue with boundaries of the 3D color path integral. We recall the notations like Δ_3^{bdout} or Δ_2^{bd,g^b} from Sec. IV E.

7. 3D state boundary

As in 2D, simply restricting the bulk cohomology,

$$\begin{aligned} X_0 &:= \Delta_3, & X_1 &:= \Delta_0, \\ X_2 &:= \Delta_2, & X_3 &:= \bigsqcup_{c \in C_1} \Delta_3^c \times \bar{c}, \end{aligned} \quad (\text{A6})$$

is not locally exact and yields a state boundary rather than a physical (and topological) boundary. Concretely, there are locally supported 1-cocycles that are not 1-coboundaries, namely one for every volume in Δ_3^{bdout} .

8. 3D all-smooth boundary

Local exactness can be fixed by adding the one generator to X_0 for each local 1-cocycle [102],

$$\begin{aligned} X_0 &:= \Delta_3 + \Delta_3^{\text{bdout}}, & X_1 &:= \Delta_0, \\ X_2 &:= \Delta_2, & X_3 &:= \bigsqcup_{c \in C_1} \Delta_3^c \times \bar{c}. \end{aligned} \quad (\text{A7})$$

The coboundary maps are still restrictions of the bulk ones in Eq. (49). This chain complex defines a boundary of the

untwisted color path integral. We call it “all-smooth,” since it corresponds to a stack of three smooth boundary conditions for the equivalent cellular cohomology. However, the chain complex does *not* yield a boundary for the (twisted) color path integral, as there is no way to extend the action of Eq. (77) in a gauge-invariant way. The chain complex also yields a boundary for the 3D color code, which represents the same topological phase as the Pauli- X boundary [103]. The fact that the action is not gauge invariant on the boundary is analogous to the fact that this color-code boundary does not admit a transversal $\bar{T}$ gate [104].

9. 3D all-rough boundary

To obtain a different boundary type we can, instead of adding generators to X_0 , regain local exactness by adding generators to X_2 and X_3 . This yields the cohomology theory shown in Eq. (121). In this case, the resulting chain complex does define a boundary for the untwisted as well as the twisted color path integral. The boundary action looks the same as the bulk one in Eq. (77), where the summation runs over all vertices $p \in \Delta_0$ including those on the boundary. Gauge invariance follows directly, since it is equivalent to the bulk gauge invariance for the special case where all vertices p beyond the boundary are set to $A_p = 0$. The chain complex also defines a boundary for the 3D color code, which is equivalent to a Pauli- Z boundary [103].

10. 3D color boundary

Instead of adding all locally supported generators to either X_0 or to X_2 and X_3 , we can add or remove some generators for both. There are two such families of boundaries. The first boundary can be defined for every color c , where we set $c = r$ (red) without loss of generality. The cohomology theory is discussed around Eq. (122). The chain complex obtained in this way defines a boundary for the untwisted as well as the twisted color path integral. The resulting gauge-invariant action extends from Eq. (77) to the boundary as shown in Eq. (123). As in the 2D case, we can rewrite the action as

$$\begin{aligned} S^{\text{bd},r}[A] &= \frac{1}{8} \sum_{p \in \Delta_0 - \Delta_0^{\text{bdin},r}} \text{sgn}_p \bar{A}_p \\ &= \frac{1}{8} \sum_{p \in \Delta_0 - \Delta_0^{\text{bd}} + \Delta_0^{\text{bdout},r}} \text{sgn}_p \bar{A}_p, \end{aligned} \quad (\text{A8})$$

since $(dA)_e = 0$ for any $e \in \Delta_1^{\text{bd},\bar{r}}$. It suffices to verify the gauge invariance $S^{\text{bd},r}[A + \alpha] = S^{\text{bd},r}[A]$ for α equal to a single generator $v \in X_0$. For volumes $v \in \Delta_3^{\text{bdin},g}$, the action on the adjacent vertices is the same as for any bulk volume if we use the first line in Eq. (A8), so gauge invariance follows as in the bulk. For volumes $v \in \Delta_3^{\text{bdout},g}$, the action on the adjacent vertices is zero if we use the second

line in Eq. (A8), so gauge invariance follows trivially. The analogous holds for b or y instead of g .

The chain complex above also yields a boundary for the 3D color code, which is in the same phase as the *color boundary* [20]. Conventionally, the color boundary is defined on a 3-colex whose boundary forms the surface of a single large r volume on the outside, which is a 2-colex. In this case, we have $\Delta_1^{\text{bd},\bar{r}} = \emptyset$, $\Delta_3^{\text{bdin},r} = \emptyset$, and $\Delta_0^{\text{bdout},r} = \emptyset$, and the chain complex and action are slightly simplified. The color boundaries we have defined generalize these boundaries, by defining them on arbitrary surfaces in the 3-colex. The action in Eq. (A8) is analogous to the phases introduced by a transversal $\bar{T}$ gate acting on the 3D color code and its color boundary.

Finally, we note that an alternative form of the chain complex is given by exchanging

$$\begin{aligned} X_2 &:= \Delta_2 + \Delta_1^{\text{bd},\bar{r}}, \\ X_3 &:= \bigsqcup_{c \in \mathcal{C}_1} \Delta_3^c \times \bar{c} + \Delta_2^{\text{bd},gb} + \Delta_2^{\text{bd},gy} + \Delta_2^{\text{bd},by}. \end{aligned} \quad (\text{A9})$$

That is, instead of removing faces from X_2 , we add the same faces to X_3 .

The ZX tensor network corresponding to the r color boundary is shown in Sec. II in Fig. 4. In fact, Fig. 4 corresponds to a mix of Eqs. (122) and (A9), where some of the faces of $\Delta_2^{\text{bd},gb} + \Delta_2^{\text{bd},gy} + \Delta_2^{\text{bd},by}$ have been removed from X_2 but not all. The $\bar{Z}$ -tensor phases in Fig. 4 correspond to the second line in Eq. (A8).

11. 3D double-color boundary

The second type of boundary that is neither the all-smooth nor all-rough can be defined for every color pair κ . Without loss of generality, we choose $\kappa = rg$. The cohomology theory is discussed around Eq. (124). Compared to the all-smooth boundary in Eq. (A7), we have removed all boundary faces from X_2 and added both the $\bar{r}$ and $\bar{g}$ -colored boundary edges. Again, this chain complex defines a boundary of both the untwisted and twisted color path integral, and the gauge-invariant boundary action looks the same as in Eq. (A8).

In total, we have found 11 distinct boundaries of the 3D twisted color path integral corresponding to three families, namely one all-rough, four color, and six double-color boundaries.

12. Boundaries of the twisted cellular path integral

In order to show that boundaries defined above capture all boundary topological phases, we relate them to well-studied boundaries of the twisted cellular path integral, which we briefly review as follows. There are two possibilities of extending $\mathbb{Z}_2$ -valued 3D cellular cohomology to a boundary in a locally exact way: The *smooth*

boundary cohomology is defined on a 3-cellulation with boundary and looks the same as in Eq. (39), where X_i also includes the i -cells inside the boundary. In contrast, for the *rough boundary cohomology*, X_i does not include the i -cells inside the boundary. The rough boundary cohomology is equivalent to the smooth boundary cohomology on the Poincaré dual lattice. When we picture 1-cocycles and 2-cocycles as closed-membrane and closed-loop patterns, respectively, then the smooth boundary corresponds to one where these membrane and loops can freely terminate at the boundary.

If we have three copies of $\mathbb{Z}_2$ -valued cellular cohomology, each copy may either have a rough or a smooth boundary, yielding eight different boundaries. However, there are additional boundaries that couple the different copies. In general, there is one boundary cohomology for every subgroup $H \subset \mathbb{Z}_2^3$: It is defined by restricting any i -(co)chain to H on all i -cells that are part of the boundary. Not all boundary cohomology theories yield boundaries of the twisted cellular path integral in Eq. (84), since there may be no gauge-invariant way of extending the action in Eq. (85) to the boundary. For example, this is not possible for the triple-smooth boundary corresponding to $H = \mathbb{Z}_2^3$. In the intuitive closed-membrane picture discussed in Sec. IV C, this can be shown as follows: Consider a point on the boundary where the termination lines of the membranes A^r and A^g intersect. There is a gauge transformation that adds a small semisphere to the closed-membrane pattern A^b around this point on the boundary. Such a gauge transformation adds a triple-intersection between the A^r , A^g , and A^b membranes and thus adds $\frac{1}{2}$ to the action in Eq. (85) [105]. The subgroups H that allow for a gauge-invariant action are given by

- (a) the trivial subgroup $\langle \emptyset \rangle$,
- (b) the 6 $\mathbb{Z}_2$ subgroups generated by

$$\begin{aligned} &\langle (1, 0, 0) \rangle, \quad \langle (0, 1, 0) \rangle, \quad \langle (0, 0, 1) \rangle, \\ &\langle (1, 1, 0) \rangle, \quad \langle (0, 1, 1) \rangle, \quad \langle (1, 0, 1) \rangle, \end{aligned} \quad (\text{A10})$$

- (c) and the 4 $\mathbb{Z}_2^2$ subgroups generated by

$$\begin{aligned} &\langle (1, 0, 0), (0, 1, 0) \rangle, \quad \langle (1, 0, 0), (0, 0, 1) \rangle, \\ &\langle (0, 1, 0), (0, 0, 1) \rangle, \quad \langle (1, 1, 0), (0, 1, 1) \rangle. \end{aligned} \quad (\text{A11})$$

All in all, we find that there are 11 distinct boundary phases for the type-III twisted cellular path integral [106,107].

13. Equivalence to cellular boundaries

We have found 11 different boundaries for the 3D color path integral and also 11 different boundaries of the twisted cellular path integral. Indeed, if we fix one way to map between the cellular and the color path integral, then there is a one-to-one correspondence between the two sets of

Color boundary	Cellular boundary
All-rough boundary	$H = \langle \emptyset \rangle$
r -color boundary	$H = \langle (0, 1, 0), (0, 0, 1) \rangle$
g -color boundary	$H = \langle (1, 0, 0), (0, 0, 1) \rangle$
b -color boundary	$H = \langle (1, 0, 0), (0, 1, 0) \rangle$
y -color boundary	$H = \langle (1, 1, 0), (0, 1, 1) \rangle$
rg -double-color boundary	$H = \langle (0, 0, 1) \rangle$
rb -double-color boundary	$H = \langle (0, 1, 0) \rangle$
gb -double-color boundary	$H = \langle (1, 0, 0) \rangle$
ry -double-color boundary	$H = \langle (0, 1, 1) \rangle$
gy -double-color boundary	$H = \langle (1, 0, 1) \rangle$
by -double-color boundary	$H = \langle (1, 1, 0) \rangle$

FIG. 24. Equivalence between boundaries of the color path integral (left) and boundaries of the type-III twisted path integral (right), defined by subgroup $H \subset \mathbb{Z}_2^3$.

boundaries. If we use the mappings M^r , M^g , and M^b from Sec. III C, then the one-to-one correspondence is shown in Fig. 24. We will not give full chain maps for all the equivalences but instead illustrate how color 1-cocycles A are mapped to cellular 1-cocycles $A^r := M^r A$, $A^g := M^g A$, and $A^b := M^b A$. The r -color boundary adds the $\bar{r}$ -colored edges to X_2 , which forces $A^r = 0$ on the boundary as A^r is defined on these edges. A^g and A^b on the other hand are unrestricted, so this corresponds to the cellular boundary with $H = \langle (0, 1, 0), (0, 0, 1) \rangle$. The analogous holds for the g and b -color boundaries. Since the sum $M^r + M^g + M^b + M^y$ is trivial, the y -color boundary corresponds to one where $A^y = A^r + A^g + A^b = 0$ at the boundary, corresponding to $H = \langle (1, 1, 0), (0, 1, 1) \rangle$. The rg -color boundary adds both the $\bar{r}$ and $\bar{g}$ edges to X_2 , which forces $A^r = 0$ and $A^g = 0$ on the boundary, corresponding to $H = \langle (0, 0, 1) \rangle$. The ry -color boundary forces $A^r = 0$ and $A^y = A^r + A^g + A^b = 0$ on the boundary, corresponding to $H = \langle (0, 1, 1) \rangle$. The other double-color boundaries are analogous. The equivalence establishes that we have found path integrals for all 11 boundary phases of the color path integral.

APPENDIX B: FAULT TOLERANCE FOR ARBITRARY LOCAL NOISE

In this appendix, we will present a fault-tolerance argument for arbitrary p -bounded local noise in the untwisted color circuits. Such an argument can be made *without* expressing the noisy circuit in terms of Pauli errors, following the proof in Ref. [28]. We assume a general noise model where the unitary channel or measurement instrument $C_0^{(l)}$ at the spacetime location $l \in \mathcal{L}$ in the circuit may be replaced with one of a list of faulty channels or instruments $\{C_x^{(l)}\}_{1 \leq x < n}$. Let $\vec{x}$ be an error configuration, that is, at each location $l \in \mathcal{L}$, we apply $C_{\vec{x}_l}^{(l)}$ instead of the noise-free channel $C_0^{(l)}$. In other words, all locations with $\vec{x}_l \neq 0$ experience an error. The noisy circuit is given by applying

the error configuration $\vec{x}$ with probability $P(\vec{x})$. We assume that P is p -bounded, meaning that

$$\sum_{\vec{x}: \vec{x}_l \neq 0 \forall l \in L} P(\vec{x}) < p^{|L|} \quad \forall L \subset \mathcal{L} \quad (\text{B1})$$

for some small enough p . Note that p -bounded noise includes on-site circuit-level noise, but also includes correlated noise to some extent.

Now, the conditions in Eqs. (92) and (93) imply that the measured flux and charge configuration always obeys $(db)_{(v,c)} = 0$ or $(d^T c)_v = 0$ on every volume v that is separated from any error location l ($\vec{x}_l \neq 0$) by more than some constant lattice distance. That is, if $R_{\vec{x}}$ denotes the region obtained by padding the set of error locations $\{l : \vec{x}_l \neq 0\}$ by some constant-size error-free margins, then db and $d^T c$ are only supported on $R_{\vec{x}}$. This is because outside of $R_{\vec{x}}$, the circuit equals the path integral, which is zero if $(db)_{(v,c)} \neq 0$ or $(d^T c)_v \neq 0$, and so the probability for measuring such a b or c is zero as well. This allows us to write down sufficient conditions for when a noisy circuit does not introduce a logical error:

- (1) $R_{\vec{x}}$ decomposes into topologically trivial components each contained inside a 3-ball.
- (2) $\tilde{b}$ ($\tilde{c}$) are homologically equal to a color 1-cochain b' (1-chain c') supported only inside $R_{\vec{x}}$.

This is because the fixed-point condition in Eq. (66) of the underlying path integral together with (1) implies that the circuit with errors is equal to the circuit without errors, with measurement outcomes given by $b + b'$ ($c + c'$). (2) then implies that $\tilde{b} + b + b_c$ ($\tilde{c} + c + c_c$) has the same (co)homology class as $b' + b + b_c$ ($c' + c + c_c$), which by construction is the trivial one. Due to the invariance in Eqs. (95) and (97), the postselected circuit with corrections is thus equal to the clean, error-free, path integral without any flux or charge configuration.

It is left to be shown that the configurations of $\vec{x}$, b , and $\tilde{b}$ that violate the conditions (1) or (2) are exponentially unlikely in the lattice size L . Here we assume that the hypergraph-matching algorithm really yields the exact minimum-weight configuration, but the argument can be adapted if this is only approximately the case. Assume that we have a configuration of $\vec{x}$, b , and c violating (1) or (2). Then, $b' + \tilde{b}$ ($c' + \tilde{c}$) is a 2-cocycle (1-cycle) of nontrivial (co)homology such that $|b' + \tilde{b}| \cap R_{\vec{x}} < \frac{1}{2}|b' + \tilde{b}|$, where $|\bullet|$ denotes the weight. This is because otherwise, b' (c') has a lower weight than $\tilde{b}$ ($\tilde{c}$) which contradicts the assumption that $\tilde{b}$ ($\tilde{c}$) is of minimum weight. The cohomologically nontrivial configuration $b' + \tilde{b}$ ($c' + \tilde{c}$) can be pictured as a noncontractible loop configuration. The number of closed loops of length L is at most α^L for some constant α , and the probability for a loop that is covered half by errors scales like $p^{L/\eta}$ for some constant η , due to

Eq. (B1). So the overall probability of a length- L loop scales like $(p^{1/\eta}\alpha)^L$. Since the size of a noncontractible loop scales like the lattice size, $L \sim L$, the probability of a logical error vanishes exponentially quickly in L if $p^{1/\eta}\alpha < 1$.

-
- [1] A. Y. Kitaev, Fault-tolerant quantum computation by anyons, *Ann. Phys.* **303**, 2 (2003).
 - [2] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, Topological quantum memory, *J. Math. Phys.* **43**, 4452 (2002).
 - [3] S. B. Bravyi and A. Y. Kitaev, Quantum codes on a lattice with boundary, [arXiv:quant-ph/9811052](https://arxiv.org/abs/quant-ph/9811052).
 - [4] H. Bombin and M. A. Martin-Delgado, Topological quantum distillation, *Phys. Rev. Lett.* **97**, 180501 (2006).
 - [5] N. Lacroix *et al.*, Scaling and logic in the color code on a superconducting quantum processor, *Nature* **645**, 614 (2025).
 - [6] S. Krinner, N. Lacroix, A. Remm, A. Di Paolo, E. Genois, C. Leroux, C. Hellings, S. Lazar, F. Swiadek, J. Herrmann, G. J. Norris, C. K. Andersen, M. Müller, A. Blais, C. Eichler, and A. Wallraff, Realizing repeated quantum error correction in a distance-three surface code, *Nature* **605**, 669 (2022).
 - [7] R. Acharya *et al.*, Quantum error correction below the surface code threshold, *Nature* **638**, 920 (2025).
 - [8] D. Horsman, A. G. Fowler, S. Devitt, and R. V. Meter, Surface code quantum computing by lattice surgery, *New J. Phys.* **14**, 123011 (2012).
 - [9] D. Litinski, A game of surface codes: Large-scale quantum computing with lattice surgery, *Quantum* **3**, 128 (2019).
 - [10] A. J. Landahl and C. Ryan-Anderson, Quantum computing by color-code lattice surgery, [arXiv:1407.5103](https://arxiv.org/abs/1407.5103).
 - [11] F. Thomsen, M. S. Kesselring, S. D. Bartlett, and B. J. Brown, Low-overhead quantum computing with the color code, *Phys. Rev. Res.* **6**, 043125 (2024).
 - [12] C. Ryan-Anderson *et al.*, Implementing fault-tolerant entangling gates on the five-qubit code and the color code, [arXiv:2208.01863](https://arxiv.org/abs/2208.01863).
 - [13] D. Bluvstein *et al.*, Logical quantum processor based on reconfigurable atom arrays, *Nature* **626**, 58 (2024).
 - [14] S. Bravyi and A. Kitaev, Universal quantum computation with ideal Clifford gates and noisy ancillas, *Phys. Rev. A* **71**, 022316 (2005).
 - [15] S. Bravyi and J. Haah, Magic state distillation with low overhead, *Phys. Rev. A* **86**, 052329 (2012).
 - [16] D. Litinski, Magic state distillation: Not as costly as you think, *Quantum* **3**, 205 (2019).
 - [17] M. Davydova, A. Bauer, J. C. Magdalena de la Fuente, M. Webster, D. J. Williamson, and B. J. Brown, Universal fault tolerant quantum computing in 2D without getting tied in knots, [arXiv:2503.15751](https://arxiv.org/abs/2503.15751).
 - [18] H. Bombin, Dimensional jump in quantum error correction, *New J. Phys.* **18**, 043038 (2016).
 - [19] M. E. Beverland, A. Kubica, and K. M. Svore, The cost of universality: A comparative study of the overhead of state distillation and code switching with color codes, *PRX Quantum* **2**, 020341 (2021).

- [20] H. Bombin and M. A. Martin-Delgado, Topological computation without braiding, *Phys. Rev. Lett.* **98**, 160502 (2007).
- [21] M. Vasmer and D. E. Browne, Three-dimensional surface codes: Transversal gates and fault-tolerant architectures, *Phys. Rev. A* **100**, 012312 (2019).
- [22] H. Bombin, 2D quantum computation with 3D topological codes, [arXiv:1810.09571](https://arxiv.org/abs/1810.09571).
- [23] B. J. Brown, A fault-tolerant non-Clifford gate for the surface code in two dimensions, *Sci. Adv.* **6**, eaay4929 (2020).
- [24] T. R. Scruby, D. E. Browne, P. Webster, and M. Vasmer, Numerical implementation of just-in-time decoding in novel lattice slices through the three-dimensional surface code, *Quantum* **6**, 721 (2022).
- [25] H. Bombin, D. Litinski, N. Nickerson, F. Pastawski, and S. Roberts, Unifying flavors of fault tolerance with the ZX calculus, *Quantum* **8**, 1379 (2024).
- [26] A. Townsend-Teague, J. M. de la Fuente, and M. Kesselring, Floquetifying the colour code, *EPTCS* **384**, 265 (2023).
- [27] A. Bauer, Topological error correcting processes from fixed-point path integrals, *Quantum* **8**, 1288 (2024).
- [28] A. Bauer, Low-overhead non-Clifford topological fault-tolerant circuits for all non-chiral Abelian topological phases, *Quantum* **9**, 1673 (2025).
- [29] A. Bauer, The $x + y$ Floquet code: A simple example for topological quantum computation in the path integral approach, *Phys. Rev. A* **111**, 032413 (2025).
- [30] A. Kubica and N. Delfosse, Efficient color code decoders in $d \geq 2$ dimensions from toric code decoders, *Quantum* **7**, 929 (2023).
- [31] H. Bombin and M. A. Martin-Delgado, Exact topological quantum order in $D = 3$ and beyond: Branyons and brane-net condensates, *Phys. Rev. B* **75**, 075103 (2007).
- [32] C. Gidney, A pair measurement surface code on pentagons, *Quantum* **7**, 1156 (2023).
- [33] A. Kissinger, Phase-free ZX diagrams are CSS codes (...or how to graphically grok the surface code), [arXiv:2204.14038](https://arxiv.org/abs/2204.14038).
- [34] J. C. Magdalena de la Fuente, J. Old, A. Townsend-Teague, M. Rispler, J. Eisert, and M. Müller, XYZ ruby code: Making a case for a three-colored graphical calculus for quantum error correction in spacetime, *PRX Quantum* **6**, 010360 (2025).
- [35] R. Raussendorf, J. Harrington, and K. Goyal, Topological fault-tolerance in cluster state quantum computation, *New J. Phys.* **9**, 199 (2007).
- [36] N. Nickerson and H. Bombin, Measurement based fault tolerance beyond foliation, [arXiv:1810.09621](https://arxiv.org/abs/1810.09621).
- [37] A 3D triangulation is a decomposition of a 3-manifold into tetrahedron volumes, which are glued at their triangle faces. A 3D cellulation is defined similarly, without requiring the volumes and faces to be tetrahedra and triangles.
- [38] Wikipedia, Bitruncated cubic honeycomb, https://en.wikipedia.org/wiki/Bitruncated_cubic_honeycomb.
- [39] B. Coecke and A. Kissinger, *Picturing Quantum Processes: A First Course in Quantum Theory and Diagrammatic Reasoning* (Cambridge University, Cambridge, 2017).
- [40] J. van de Wetering, ZX-calculus for the working quantum computer scientist, [arXiv:2012.13966](https://arxiv.org/abs/2012.13966).
- [41] C. Chamberland and M. E. Beverland, Flag fault-tolerant error correction with arbitrary distance codes, *Quantum* **2**, 53 (2018).
- [42] J. C. Magdalena de la Fuente, Dynamical weight reduction of Pauli measurements, [arXiv:2410.12527](https://arxiv.org/abs/2410.12527).
- [43] B. Rodatz, B. Poór, and A. Kissinger, Floquetifying stabiliser codes with distance-preserving rewrites, [arXiv:2410.17240](https://arxiv.org/abs/2410.17240).
- [44] Alternatively, we could have chosen a unit cell spanned by t , $\frac{1}{3}t + x$, and $\frac{1}{3}t + y$. In this case we would specify the circuit on a smaller patch of the spatial lattice but for 12 instead of 4 time steps.
- [45] A fourth y color boundary is completely analogous but not used in the protocol.
- [46] One could derive these boundaries from analogous color boundaries of the 3D color code. Unfortunately, r color boundaries are commonly defined along a surface inside the triangulation dual to the 3-colex, which cuts only through $\bar{r}$ -colored edges of the primal 3-colex [31]. Our color boundaries are defined on an arbitrary surface inside the primal 3-colex, and the additional bonds at the red boundary edges would correspond to additional ZZ stabilizers at the boundary of the 3D color code. Reference [103] defines similar looking boundaries with ZZ stabilizers but still considers surfaces in the dual triangulation.
- [47] This is different from the usual way of defining a color boundary along a path in the dual lattice that only cuts perpendicular through $\bar{r}$ -colored edges. Such boundaries are usually placed along the “arm-chair” direction of the hexagonal lattice, which is rotated by 60° relative to the zig-zag direction shown in Fig. 7.
- [48] The logical action on $S_1 \times S_1 \times [0, 1]$ will not be a $\bar{T}$ gate, and the logical action of the protocol in Eq. (1) will be yet different, see Ref. [17]. We are not interested in the logical action of the $3 + 0D$ protocol here, but use it as tool to derive the interface of the color path integral.
- [49] More precisely, the 3D color code is equivalent to three copies of the 3D toric code, so b maps to three separate string configurations as we show in Sec. III. We will pretend that there is only one string configuration to simplify the following discussion.
- [50] H. Bombin, Single-shot fault-tolerant quantum error correction, *Phys. Rev. X* **5**, 031043 (2015).
- [51] H. Bombin, Transversal gates and error propagation in 3D topological codes, [arXiv:1810.09575](https://arxiv.org/abs/1810.09575).
- [52] More precisely, there are three sorts of flux and charge strings each, and the charge strings of one sort can terminate on the residual flux strings of the other two sorts [51].
- [53] N. Delfosse, Decoding color codes by projection onto surface codes, *Phys. Rev. A* **89**, 012317 (2014).
- [54] Usually, the cochains are defined with coefficients in the Pontryagin dual group of the chains. For us, the coefficient group is $\mathbb{Z}_2$ which can be identified with its Pontryagin dual via the pairing (or “Fourier transform” or “Hadamard gate”) $(a, b) \rightarrow (-1)^{ab}$, so we identify the cochains with chains. If we sometimes call a chain “cochain,” this is only

- to signal that we intend to apply the coboundary operator d_i to it instead of the boundary operator d_{i-1}^T .
- [55] We could define homology to be the cohomology of the Pontryagin dual chain complex consisting of $(\mathbb{Z}_2^{X_i})^*$ and d_i^* , then we simply have $H_i = (H^i)^*$. Since we define it via the transpose matrices, there is a canonical pairing between H_i and H^i .
- [56] More precisely, $(\partial_{ij})_{xy}$ is the mod 2 number of times y is contained in x , but this is only relevant for very irregular lattices or very small lattices with periodic boundary conditions.
- [57] Note that a chain complex is called *exact* if every (co)cycle is a (co)boundary, that is, if the (co)homology is trivial. Here we demand that this is true only for locally supported (co)cycles, such that the (co)homology can be nontrivial, but can only depend on the global properties (usually, the topology) of the lattice.
- [58] V. G. Turaev and O. Y. Viro, State sum invariants of 3-manifolds and quantum 6j-symbols, *Topology* **31**, 865 (1992).
- [59] J. W. Barrett and B. W. Westbury, Invariants of piecewise-linear 3-manifolds, *Trans. Amer. Math. Soc.* **348**, 3997 (1996).
- [60] M. Fukuma, S. Hosono, and H. Kawai, Lattice topological field theory in two dimensions, *Commun. Math. Phys.* **161**, 157 (1994).
- [61] R. Dijkgraaf and E. Witten, Topological gauge theories and group cohomology, *Commun. Math. Phys.* **129**, 393 (1990).
- [62] L. Crane and D. N. Yetter, in *Quantum Topology*, edited by L. Kauffman and R. Baadhio (World Scientific, Singapore, 1993).
- [63] A. Bauer, J. Eisert, and C. Wille, A unified diagrammatic approach to topological fixed point models, *SciPost Phys. Core* **5**, 38 (2022).
- [64] A. Bauer, Topological phases, fixed-point models, extended TQFT, and fault-tolerant quantum computation-tensors in spacetime, Ph.D. thesis, Freie Universität Berlin, https://refubium.fu-berlin.de/bitstream/handle/fub188/45937/Dissertation_Andreas_Bauer.pdf?sequence=4, 2023.
- [65] If we divide the boundary into an *input* and an *output* component, the state can also be interpreted as a vectorized linear map from the input to the output vector space.
- [66] Physically, our path integrals have the same qualitative properties as the imaginary-time evolution of a 2D Hamiltonian [63]. It should be noted though, that our path integrals are defined in discrete time, whereas the imaginary-time evolution is defined in continuous time. One could in principle obtain a path integral from an imaginary-time evolution by Trotterization, but our path integrals do not directly arise in this way.
- [67] Note that topological phases can be defined directly on the level of path integrals: Two path integrals are in the same phase if they are related by a continuous deformation that does maintain a constant spectral gap of the transfer operator (corresponding to a 2D layer of the 3D path integral). We believe that the resulting classification of topological phases is closely related to that for lattice Hamiltonians.
- [68] For usual examples, d_0 is very small, for example it might suffice if the interior and exterior are separated by at least one lattice edge.
- [69] Here, we mean the connected correlator, $\langle O_0 O_1 \rangle - \langle O_0 \rangle \langle O_1 \rangle$.
- [70] Since we sum over all boundary variables, evaluating the path integral with physical boundary yields again a number Z . We can also evaluate the path integral on a lattice whose boundary is divided into a physical-boundary part and a state-boundary part, yielding a state supported only on the state-boundary part. In this case, the state itself has a physical boundary, which is the region separating state and physical boundary in spacetime.
- [71] M. D. W. Propitius, Topological interactions in broken gauge theories, Ph.D. thesis, University of Amsterdam, <http://arxiv.org/abs/hep-th/9511195>.
- [72] N. E. Steenrod, Products of cocycles and extensions of mappings, *Ann. Math.* **48**, 290 (1947).
- [73] Y.-A. Chen and S. Tata, Higher cup products on hypercubic lattices: Application to lattice models of topological phases, *J. Math. Phys.* **64**, 091902 (2023).
- [74] If there is an edge e with both $(A_r)_e = 1$ and $(A_g)_e = 1$, it becomes ambiguous whether there is a red-green intersection at the edge center or not, which can be resolved by shifting the red and green edge centers with respect to another. However, this case does not appear when showing the equivalence of the color and twisted cellular path integrals.
- [75] Alternatively, we can interpret c as taking values in the Pontryagin dual $\mathbb{Z}_2^*$. Then this weight is equal to the evaluation of c on A .
- [76] S. Roberts and D. J. Williamson, 3-fermion topological quantum computation, *PRX Quantum* **5**, 010315 (2024).
- [77] D. Bulmash and M. Barkeshli, Absolute anomalies in $(2+1)$ D symmetry-enriched topological states and exact $(3+1)$ D constructions, *Phys. Rev. Res.* **2**, 043033 (2020).
- [78] M. McEwen, D. Bacon, and C. Gidney, Relaxing hardware requirements for surface code circuits using time-dynamics, *Quantum* **7**, 1172 (2023).
- [79] nCat Lab, Quadratic refinement, <https://ncatlab.org/nlab/show/quadratic+refinement>.
- [80] In fact, a flux string corresponds to the direct sum of both non-Abelian flux anyons.
- [81] If the fluxes in the model would correspond to Abelian worldlines, then the invariance in Eq. (95) can be rescued by adding further terms to the action as it was done in Ref. [28].
- [82] K. Sahay and B. J. Brown, A decoder for the triangular color code by matching on a Möbius strip, *PRX Quantum* **3**, 010310 (2022).
- [83] C. Gidney and C. Jones, New circuits and an open source decoder for the color code, [arXiv:2312.08813](https://arxiv.org/abs/2312.08813).
- [84] M. B. Hastings and J. Haah, Dynamically generated logical qubits, *Quantum* **5**, 564 (2021).
- [85] A. Paetznick, C. Knapp, N. Delfosse, B. Bauer, J. Haah, M. B. Hastings, and M. P. da Silva, Performance of planar Floquet codes with Majorana-based qubits, *PRX Quantum* **4**, 010310 (2023).
- [86] Concretely, this cohomology looks like 3D color cohomology on spacetime volumes with twisted color circuits,

- and like 3D cellular cohomology on volumes with surface-code syndrome-extraction circuits. For volumes with 2D-color-code syndrome-extraction circuits, the cohomology theory corresponds to some sort of “skeletal color cohomology,” defined on a “skeletal 3-colex” which is dual to a cellulation with only triangle faces and 3-colorable vertices but arbitrary volumes. Transversal H or S gates on Abelian codes correspond to domain walls with an action in addition to the cohomology theory. In contrast to the action of a T or CCZ gate, this action is quadratic and leads to a deterministic coupling between the constraints for b and c , which can be decoded globally using methods similar to these in Ref. [28].
- [87] H. Bombín, C. Dawson, Y.-H. Liu, N. Nickerson, F. Pastawski, and S. Roberts, Modular decoding: Parallelizable real-time decoding for quantum computers, [arXiv:2303.04846](#).
 - [88] This has the benefit that such magic states can be prepared offline, in parallel to the remaining logical circuit.
 - [89] J. Edmonds, Paths, trees, and flowers, *Can. J. Math.* **17**, 449 (1965).
 - [90] O. Higgott and C. Gidney, Sparse blossom: Correcting a million errors per core second with minimum-weight matching, *Quantum* **9**, 1600 (2025).
 - [91] We expect the twisted color circuits to have a somewhat lower threshold since the just-in-time decoded X errors effectively lead to further Z errors.
 - [92] A. Kubica, M. E. Beverland, F. Brandao, J. Preskill, and K. M. Svore, Three-dimensional color code thresholds via statistical-mechanical mapping, *Phys. Rev. Lett.* **120**, 180501 (2018).
 - [93] Note that this decoding has nothing to do with the flux decoding in the measurement-error-free case for the 3 + 1D color code, whose maximum-likelihood threshold is much higher.
 - [94] B. J. Brown, N. H. Nickerson, and D. E. Browne, Fault-tolerant error correction with the gauge color code, *Nat. Commun.* **7**, 12302 (2016).
 - [95] H. G. Katzgraber, H. Bombin, and M. A. Martin-Delgado, Error threshold for color codes and random three-body Ising models, *Phys. Rev. Lett.* **103**, 090501 (2009).
 - [96] T. R. Scruby and Z. Cai, Fault-tolerant quantum computation without distillation on a 2D device, [arXiv:2412.12529](#).
 - [97] N. P. Breuckmann, M. Davydova, J. N. Eberhardt, and N. Tantivasadakarn, Cups and gates I: Cohomology invariants and logical quantum operations, [arXiv:2410.16250](#).
 - [98] G. Zhu, A topological theory for qLDPC: non-Clifford gates and magic state fountain on homological product codes with constant rate and beyond the $n^{1/3}$ distance barrier, [arXiv:2501.19375](#).
 - [99] P.-S. Hsin, R. Kobayashi, and G. Zhu, Classifying logical gates in quantum codes via cohomology operations and symmetry, [arXiv:2411.15848](#).
 - [100] T.-C. Lin, Transversal non-Clifford gates for quantum LDPC codes on sheaves, [arXiv:2410.14631](#).
 - [101] M. S. Kesselring, F. Pastawski, J. Eisert, and B. J. Brown, The boundaries and twist defects of the color code and their applications to topological quantum computation, *Quantum* **2**, 101 (2018).
 - [102] In the language of subsystem codes this is similar to *gauge fixing* an associated (spacetime) logical operator.
 - [103] Z. Song and G. Zhu, Magic boundaries of 3D color codes, *Quantum* **9**, 1831 (2025).
 - [104] Even though the same chain complexes can be used to define boundaries for both the color path integral and the 3D color code, their classifications are fundamentally different: In general, the boundaries of the 3D color code correspond to the boundaries of a 4D path integral.
 - [105] Note that the cellular “twist” $(-1)^{A^r \cup A^g \cup A^b}$ is described by a group 3-cocycle $\omega \in H^3(B\mathbb{Z}_2^3, U(1))$, and a boundary is described by a subgroup H and a group 2-cocycle $\psi \in H^2(B\mathbb{Z}_2^3, U(1))$ such that $d\psi = \omega|_H$. So we only get boundaries if $\omega|_H$ is a cohomologically trivial group 3-cocycle, which is the case for subgroups $H \subset \mathbb{Z}_2^3$ that do not contain $(1, 1, 1) \in \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2$. Note that for $H \sim \mathbb{Z}_2^2$, there are two distinct group 2-cohomology classes corresponding to two distinct boundaries, but these distinct boundaries turn out to belong to the same boundary topological phase nonetheless [17].
 - [106] A. Bullivant, Y. Hu, and Y. Wan, Twisted quantum double model of topological orders with boundaries, *Phys. Rev. B* **96**, 165138 (2017).
 - [107] J. C. Magdalena de la Fuente, J. Eisert, and A. Bauer, Bulk-to-boundary anyon fusion from microscopic models, *J. Math. Phys.* **64**, 111904 (2023).