Topological subsystem codes

H. Bombin

Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts, 02139 USA and
Perimeter Institute for Theoretical Physics, 31 Caroline St. N., Waterloo, Ontario N2L 2Y5, Canada
(Received 26 October 2009; published 3 March 2010)

We introduce a family of two-dimensional (2D) topological subsystem quantum error-correcting codes. The gauge group is generated by two-local Pauli operators, so that two-local measurements are enough to recover the error syndrome. We study the computational power of code deformation in these codes and show that boundaries cannot be introduced in the usual way. In addition, we give a general mapping connecting suitable classical statistical mechanical models to optimal error correction in subsystem stabilizer codes that suffer from depolarizing noise.

PACS number(s): 03.67.Pp

I. INTRODUCTION

Quantum error correction [1–4] and fault-tolerant quantum computation [5–9] promise to allow almost perfect storage, transmission, and manipulation of quantum information. Without them, quantum information processing will be doomed to failure due to the decoherence produced by interactions with the environment and the unavoidable inaccuracies of quantum operations.

The key concept in quantum error correction is the notion of quantum code. This is a subspace of a given quantum system where quantum information can be safely encoded in the sense that the adverse effects of noise can be erased through an error correction procedure. In practice this procedure is also subject to errors and thus it should be as simple as possible to minimize them. Naturally, the meaning of "simple" will depend on particular implementations. A common situation is that interactions are restricted to quantum subsystems that are close to each other in space. In those cases, the locality of the operations involved in error correction becomes crucial.

The stabilizer formalism [10,11] provides a unified framework for many quantum codes. In stabilizer codes, the main step for error correction is the measurement of certain operators, which may be local or not. A class of codes where these measurements are intrinsically local is that of topological stabilizer codes [12–15]. In a different direction, locality can also be enhanced by considering more generally stabilizer subsystem codes [16,17]. The present work provides an example of a family of codes that can be labeled both as "topological" and "subsystem."

Topological codes were originally introduced with the goal of obtaining a self-protecting quantum computer [12]. This idea faces important difficulties in low dimensions since thermal instabilities are known to occur [13,18,19]. On the other hand, topological codes are local in a natural way and have very interesting features in the context of active error correction. For example, they do not only allow us to perform operations transversally [14,15], but also through code deformations [13,20]. Moreover, there exists a useful connection between error correction in topological codes and certain classical statistical models [13,21,22].

Stabilizer subsystem codes are the result of applying the stabilizer formalism to operator quantum error correction [23]. In subsystem codes, part of the logical qubits that form the code subspace are no longer considered as such, but rather as gauge qubits where no information is encoded. This not only allows the gauge qubits to absorb the effect of errors, but has interesting consequences for error correction. It may allow us to break up each of the needed measurements into several different ones that involve a smaller number of qubits [16,17]. An example of this is offered by the Bacon-Shor codes [16] in which the basic operators to be measured can have support on an arbitrarily large number of qubits, yet their eigenvalues can be recovered from two-local measurements that do not damage the encoded information. Moreover, the pairs of qubits to be measured together are always neighbors in a two-dimensional (2D) lattice. Thus, subsystem codes can have very nice locality properties.

The 2D topological subsystem codes introduced here show all the characteristic properties of topological codes and at the same time take profit from the advantages of subsystem codes. Some of them are:

- The codes are local in a 2D setting, which can be flat.
- The measurements needed for error correction only involve two neighboring qubits at a time, as in the Bacon-Shor codes. This is an important advantage with respect to other topological codes, such as surface codes that require measuring groups of at least four qubits.
- Most errors with a length up to cn are correctable, where c is some constant and n the number of physical qubits. This feature, common to topological codes, follows from the fact that logical operators are related to strings that wind nontrivially around the surface where the code is defined.
- Error correction must be done only "up to homology," an important simplification that allows the introduction of specific tools.
- One can naturally perform certain logical gates through "deformations" of the code. This feature, however, is less powerful than that in other topological codes because boundaries cannot be introduced in the usual way.

Since these codes are topological, it is natural to expect a connection between their correction procedures and suitable classical statistical models. However, the mapping between surface codes and random Ising models [13] makes strong use

of their CSS structure [24,25] and the same is true for the one between color codes and three-body random Ising models [22]. The CSS structure makes it possible to completely separate the correction of phase-flip and bit-flip errors, making the problem classical and enabling the connection. Indeed there exist similar mappings from classical codes to statistical models [26]. Fortunately, as explained in Sec. VI, the approach can be generalized even in the absence of this separation. Moreover, the subsystem structure is also compatible with the approach so that it can be applied to the family of codes of interest.

The article is organized as follows. Sections II and III review several aspects of quantum error correction and topological codes, respectively, setting up a framework for the rest of the article. Section IV introduces the family of topological subsystem codes and presents a thorough study of their properties. Section V offers the construction of a general mapping between error correction in subsystem codes and classical statistical models. Section VI is devoted to conclusions.

II. STABILIZER QUANTUM ERROR-CORRECTING CODES

This section summarizes the notions of quantum error correction that will be needed in the rest of the article. It mainly reviews stabilizer codes both in the subspace and the more general subsystem formulation. Ideal error correction procedures and their success probability are also considered.

A. Quantum error correction

Quantum error correction deals with the preservation of quantum information in the presence of noise. Both the noise $\mathcal{E}$ and the error recovery $\mathcal{R}$ are modeled as quantum operations or channels $\mathcal{E}, \mathcal{R} : L(H) \longrightarrow L(H)$, where $L(H)$ is the space of linear operators on H, the Hilbert space associated to the quantum system under consideration. Such maps can always be expressed in the operator-sum representation. For example, the noise is $\mathcal{E}(\rho) = \sum_i E_i \rho E_i^\dagger$ for some $E_i \in L(H)$, which will be denoted by $\mathcal{E} = \{E_i\}$.

In the original formulation of quantum error correction [1–4], quantum information is encoded in a subspace of H and the code subspace $C \subset H$. The system undergoes a noisy process $\mathcal{E}$ and afterward an error recovery operation $\mathcal{R}$ is performed. Then, given a code C, a noise source $\mathcal{E}$ is said to be correctable if there exists a recovery operation $\mathcal{R}$ such that $\mathcal{R} \circ \mathcal{E}(\rho) = \rho$ for any state $\rho \in L(C)$.

More generally, in the operator quantum error correction formalism [23], information is encoded in a subsystem A, with $C = A \otimes B$. Whatever happens to subsystem B is irrelevant. That is, error recovery is possible for a quantum channel $\mathcal{E}$ if there exists a recovery operation $\mathcal{R}$ such that for any $\rho^A \in L(A)$ and $\rho^B \in L(B)$, it gives $\mathcal{R} \circ \mathcal{E}(\rho^A \otimes \rho^B) = \rho^A \otimes \rho'^B$ for some arbitrary ρ'^B.

The sufficient and necessary condition for the noise process $\mathcal{E} = \{E_i\}$ to be correctable [3,4,23] is that $P E_i^\dagger E_j P = \mathbf{1}^A \otimes g_{ij}^B$ for every i and j, with P the projector onto the code subspace. When this condition holds, the set of errors $\{E_i\}$ is said to be correctable. Since adding a linear combination of the E_a to the set does not change correctability, it is natural to consider correctable sets of errors as linear subspaces and

to choose the most convenient operator basis. Generally the quantum system is composed of n qubits, $H \simeq (\mathbf{C}^2)^{\otimes n}$, and error operators are chosen to be Pauli operators; elements of the Pauli group $\mathcal{P}_n := \langle i\mathbf{1}, X_1, Z_1, \ldots, X_n, Z_n \rangle$. Here X_i, Z_i are, as usual, the Pauli operators on the ith qubit $X = |0\rangle\langle 1| + |1\rangle\langle 0|$, $Z = |0\rangle\langle 0| - |1\rangle\langle 1|$ in the orthonormal basis $\{|0\rangle, |1\rangle\}$.

Usually error models are such that errors that affect more qubits are less likely to happen. Then it makes sense to correct as many errors as possible among those that have support on (act nontrivially on) a smaller number of qubits. The weight $|E|$ of a Pauli operator $E \in \mathcal{P}_n$ is defined as the number of qubits that form its support. When a code can correct all Pauli errors E with $|E| \leqslant r$ it is said to correct r errors.

B. Stabilizer subspace codes

A formalism that was particularly successful for the development of quantum codes is the stabilizer formalism [10,11] in which the code C is described in terms of an Abelian subgroup $\mathcal{S} = \langle S_j \rangle \subset \mathcal{P}_n$ such that $-1 \notin \mathcal{S}$. Take the generators S_j to be independent and let s be the rank of $\mathcal{S}$. The n-qubit Hilbert space H can be partitioned according to the eigenvalues of the S_j into 2^s isomorphic orthogonal subspaces $H = \bigoplus_s C_s$. Here $s = (s_j)$ is the error syndrome, with $s_j = \pm 1$ the eigenvalue of S_j. By convention, the code subspace C is that with $s_j = 1$ for all j. It has dimension 2^k, with $k = n - s$ the number of encoded or logical qubits. The reason to call s the error syndrome is that in can be obtained by measuring the S_j and then used to infer which errors were produced.

It is easy to introduce a Pauli group for the k logical qubits. Let $N(\mathcal{S})$ be the normalizer of $\mathcal{S}$ in $\mathcal{P}_n$. Its elements are the Pauli operators that map the subspaces C_s onto themselves and the quotient group $N(\mathcal{S})/\mathcal{S}$ is isomorphic to $\mathcal{P}_k$. The logical Pauli operators are then generated by $\hat{X}_1, \hat{Z}_1, \ldots, \hat{X}_k, \hat{Z}_k \in N(\mathcal{S})$, some chosen representatives of the images of $X_1, Z_1, \ldots, X_k, Z_k \in \mathcal{P}_k$ under a given isomorphism.

It is also possible to characterize a stabilizer code with a pair (U, s), where U is an automorphism of $\mathcal{P}_n$ and s an integer $0 \leqslant s \leqslant n$. Let $\tilde{X}_i, \tilde{Z}_i$ denote the images of X_i, Z_i through U. Then the stabilizer is $\mathcal{S} = \langle \tilde{Z}_1, \ldots, \tilde{Z}_s \rangle$. This approach directly provides a choice for encoded Pauli operators $\hat{X}_1 := \tilde{X}_{s+1}, \hat{Z}_1 := \tilde{Z}_{s+1}, \ldots, \hat{X}_k := \tilde{X}_n, \hat{Z}_k := \tilde{Z}_n$.

Pauli errors E have a specially simple effect on the encoded states, as they map the subspaces C_s one onto another. Set $s_E = s$ when $E S_j E^\dagger = s_j S_j$. Then a Pauli error E maps C onto C_{s_E}. Pauli operators are divided into three categories. The elements of $\mathcal{P}_n - N(\mathcal{S})$ map the code to other subspaces and are termed detectable errors, as their effect can be detected by measuring the operators $\{S_j\}$. The elements of $N(\mathcal{S}) - \mathcal{S}'$, with $\mathcal{S}' := \langle i\mathbf{1} \rangle \mathcal{S}$, map, in a nontrivial way, the code to itself and are thus called undetectable errors. Finally, the elements of $\mathcal{S}'$ have no effect on the encoded states $\rho \in L(C)$. The distance d of the code is defined as the minimum weight among undetectable errors. It determines the number of corrected errors, which is $\lfloor (d-1)/2 \rfloor$. A code of n qubits that encodes k qubits and has distance d is denoted $[[n, k, d]]$.

C. Stabilizer subsystem codes

The stabilizer formalism can also be used in the context of operator quantum error correction [16,17]. Instead of being

characterized by a stabilizer group, subsystem stabilizer codes are almost determined by a subgroup $\mathcal{G} \subset \mathcal{P}_n$, called the gauge group, such that $i\mathbf{1} \in \mathcal{G}$. In addition, a stabilizer group $\mathcal{S}$ has to be chosen such that $\mathcal{S}'$, as defined earlier, is the center of $\mathcal{G}$. There are different choices for $\mathcal{S}$ because the sign of some of its generators can always be flipped. This amounts to different choices for C in the decomposition $H = \bigoplus_s C_s$.

The idea after the introduction of the gauge group is that gauge operations should not affect the encoded information. This forces us to identify states such as ρ and $G_j \rho G_j^\dagger$ as equivalent, giving rise to a subsystem structure $C_s = A_s \otimes B_s$. The decomposition is such that the gauge operators G_j act trivially in the A_s subsystems and generate the full algebra of operators of the B_s subsystems. Set $C = A \otimes B$, with A the logical subsystem where information is encoded and B the gauge subsystem that absorbs the effect of gauge operations. Since $\mathcal{G}/\mathcal{S} \simeq \mathcal{P}_r$ for some r, B consists of r qubits. Similarly, the Pauli operators for the k logical qubits are recovered from the isomorphism $N(\mathcal{G})/\mathcal{S} \simeq \mathcal{P}_k$ and $k + r + s = n$.

It is also possible to characterize a stabilizer subsystem code as a triplet (U, s, r), where U is an automorphism of $\mathcal{P}_n$ and $s, r \geqslant 0$ are integers with $r + s \leqslant n$. Using the same notation as earlier, the stabilizer and gauge groups are $\mathcal{S} = \langle \tilde{Z}_1, \ldots, \tilde{Z}_s \rangle$ and $\mathcal{G} = \langle i\mathbf{1}, \tilde{Z}_1, \ldots, \tilde{Z}_{s+r}, \tilde{X}_{s+1}, \ldots, \tilde{X}_r \rangle$. The chosen logical Pauli operators are $\hat{X}_1 := \tilde{X}_{s+r+1}, \hat{Z}_1 := \tilde{Z}_{s+r+1}, \ldots, \hat{X}_k := \tilde{X}_n, \hat{Z}_k := \tilde{Z}_n$.

In subsystem codes, detectable Pauli errors are the elements of $\mathcal{P}_n - N(\mathcal{S})$ and undetectable ones are those in $N(\mathcal{S}) - \mathcal{G}$. Undetectable errors are directly related to logical Pauli operators. Indeed $N(\mathcal{S})/\mathcal{G} \simeq N(\mathcal{G})/\mathcal{S}'$, as the following one-to-one correspondence shows. For any $E \in N(\mathcal{S})$ there exists a $G \in \mathcal{G}$ such that $EG \in N(\mathcal{G})$ and if $G' \in \mathcal{G}$ is such that $EG' \in N(\mathcal{G})$ then $GG' \in \mathcal{S}' = \mathcal{G} \cap N(\mathcal{G})$. The distance d of the code is defined as for subspace codes and has the same implications regarding error correction. A subsystem code of n qubits that encodes k qubits and has r gauge qubits and distance d is denoted $[[n, k, r, d]]$.

D. Syndrome measurement

An interesting property of stabilizer subsystem codes is that they may allow an easier measurement of the stabilizer generators. This is so because it is possible to substitute the direct measurement of a stabilizer element S by an indirect one, in which t self-adjoint gauge operators G_i such that $S = G_1 \cdots G_t$ are measured. It may be the case that the G_i has a smaller weight than S. For example, in the family of Bacon-Shor codes the gauge generators G_i always have weight $|G_i| = 2$, but the smallest stabilizer generators have an arbitrarily large weight. Such cases offer two important advantages. On the one hand, the smaller the weight of a Pauli operator, the simpler the operations needed to measure it. This is especially relevant in fault-tolerant quantum computing where error correction is considered a faulty process in itself because simpler operations imply less errors. On the other hand, it may be possible to measure the G_i in parallel, with the subsequent saving of time. This is again relevant for fault tolerance, where the ability to perform measurements faster entails less errors.

Since the G_i need not commute, the ordering of the measurements is relevant. In general, the ordered measurement

of a collection of t operators $E_1, \ldots, E_t \in \mathcal{P}_n^\dagger$ yields the effective measurement of an Abelian group of self-adjoint Pauli operators $\mathcal{M} \subset \mathcal{Z}$, with $\mathcal{Z}$ the center of $\langle -\mathbf{1}, E_1, \ldots, E_t \rangle$. In particular, $\mathcal{M} = \mathcal{N} \cap \mathcal{Z}$ with $\mathcal{N}$ the Abelian group of those self-adjoint Pauli operators with eigenvalues fixed by the sequence of measurements. $\mathcal{N}$ can be computed iteratively since adding an additional measurement E_{t+1} changes $\mathcal{N}$ to $\mathcal{N}' = \langle E_{t+1} \rangle [\mathcal{N} \cap N(E_{t+1})]$.

E. Error correction

Even if a noisy channel $\mathcal{E}$ is not correctable for a given subsystem code, there exists some probability of performing a successful error recovery. For example, if each of the physical qubits that compose a code undergoes a depolarizing channel $\{(1 - 3p)^{\frac{1}{2}}\mathbf{1}, p^{\frac{1}{2}}X, p^{\frac{1}{2}}Y, p^{\frac{1}{2}}Z\}$, then the noise is certainly not correctable, but the success probability can still be close to 1. This section quantifies this probability, which is of primary importance for topological codes [13].

Some notation is needed here. Set as equivalent $E \sim E'$ those operators $E, E' \in L(H)$ that have the same action up to gauge elements $E = E'G$ for some $G \in \mathcal{G}$. The corresponding equivalence classes will be denoted $\bar{E}$. Let $\{D_i\}_{i=1}^{4k} \subset N(\mathcal{S})$ be a particular set of representatives for $N(\mathcal{S})/\mathcal{G}$, taking in particular $D_1 = \mathbf{1}$. The D_i with $i > 1$ will represent the ways in which error correction can fail. For example, if there is one encoded qubit a choice is $\{\mathbf{1}, \hat{X}_1, \hat{Z}_1, \hat{X}_1\hat{Z}_1\}$. To extend the equivalence of operators to channels, we choose the minimal equivalence relation such that $\{E_i\} \sim \{E_i'\}$ whenever $E_i \sim E_i'$ for all i.

Assume an error model in which Pauli errors $E \in \mathcal{P}_n$ occur with a given probability $p(E)$. That is, $\mathcal{E} = \{p(E)^{\frac{1}{2}}E\}_{E \in \mathcal{P}_n}$. Errors with different phases will not be distinguished when discussing $p(E)$ because phases are irrelevant. Up to gauge operations the error channel is $\mathcal{E} \sim \{p(\bar{E})^{\frac{1}{2}}E\}_{\bar{E} \in \mathcal{P}_n/\mathcal{G}}$, where E denotes any chosen element of $\bar{E}$ and $p(\bar{E}) = \sum_{G \in \mathcal{G}/\langle i\mathbf{1}\rangle} p(EG)$ is the probability for a given class of errors to happen. This makes it already apparent that class probabilities $p(\bar{E})$ are more important than individual error probabilities $p(E)$.

Error recovery starts with the measurement of the stabilizer generators S_j. This yields the error syndrome s, which limits the errors E that possibly happened to those with $s_E = s$. These possible errors are arranged into different classes, which may be labeled by choosing any possible error E and taking as representatives the elements $\{E D_i\}$. Then the conditional probability for the class of errors $\bar{E}$ to have happened given the syndrome outcome s is

$$p(\bar{E}|s) = \frac{p(\bar{E})}{\sum_i p(\bar{E}\bar{D}_i)}. \tag{1}$$

Suppose that these conditional probabilities can be computed, which may be potentially difficult due to the combinatorics. Then the class $\bar{E} = \bar{E}_s$ that maximizes $p(\bar{E}|s)$ is known and the optimal recovery operation is $\mathcal{R} = \{E_s P_s\}_s$, where P_s is the projector onto the subspace C_s. The combined effect of errors and recovery is $\mathcal{R} \circ \mathcal{E} \sim \{p_i^{\frac{1}{2}}D_i\}_i$ for some probabilities p_i that only depend on the error distribution $p(\bar{E})$. This gives a

success probability for the error recovery

$$p_0 = \sum_s p(\bar{E}_s). \tag{2}$$

An inadequate feature of expression (2) is that it depends on E_s. Consider an alternative error correction procedure where the class of errors with the maximum probability $\bar{E}_s$ is not always chosen. Instead, an operator from a class $\bar{E}$ is applied with probability $p(\bar{E}|s)$, so that $\mathcal{R}' = \{p(\bar{E}|s)^{\frac{1}{2}} E P_s\}_{s,\bar{E}}$. The success probability for this randomized correction procedure is

$$p_0' = \sum_{\bar{E}} p(\bar{E}) p(\bar{E}|s_E) = \sum_E p(E) p(\bar{E}|s_E). \tag{3}$$

This procedure is, at best, as successful as the original one, giving the bound $p_0' \leqslant p_0$. Notice that $p_0 = 1$ if and only if $p_0' = 1$. It follows that $p_0 = 1$ if and only if for any $D \in N(\mathcal{S}) - \mathcal{G}$

$$\sum_E p(E) p(\bar{E}\bar{D}|s_E) = 0. \tag{4}$$

This was the condition used in Ref. [13] to characterize successful recovery.

III. TOPOLOGICAL STABILIZER CODES

This section gathers together several aspects of topological stabilizer codes to provide a reference for Sec. IV. The goal is to put the subsystem codes introduced there in a broader context, making apparent the similarities and differences with previously known local and topological codes.

A. Local codes

In the context of fault-tolerant quantum computing it is advantageous to be able to perform the syndrome measurements in a simple way. It may be desirable, for example, that the number of qubits that form the support of the operators to be measured is small. Similarly, it may be convenient that the qubits that form the code only belong to the support of a small number of such operators. This idea can be formalized to give rise to the notion of local families of codes [12]. In particular, a family of stabilizer subspace codes $\{C_i\}$ is local when (i) it contains codes of arbitrary distance and (ii) there exist two positive integers μ, ν such that for each C_i there exists a family of generators of the stabilizer S_j such that (ii.a) $|S_j| \leqslant \mu$ and (ii.b) the number of S_j-s with support on any given qubit is smaller than ν.

The drawback of such an abstract notion of locality is that it is not related in any way to a particular geometry. Many physical settings have qubits disposed in 1, 2, or 3 spatial dimensions and only allow the direct interaction of nearby qubits. To reflect this fact and without loss of generality regarding the nature of the lattices, the previous definition may be modified as follows [27]. First, the qubits of each code C_i are considered to be disposed in the vertices of some finite, periodic, cubic lattice of a given dimension D. Second, instead of (ii) there must exist a positive number d, independent of C_i, such that the support of all the S_j is contained in some cube containing d^D vertices. A family of codes that is local in D dimensions is always local in the previous sense.

A similar notion of local stabilizer subsystem codes may be defined by substituting the stabilizer with the gauge group. Notice that a family of subsystem codes might have local gauge generators but nonlocal stabilizer generators, as the Bacon-Shor codes exemplify.

B. Topological codes

Topological stabilizer codes are constructed from lattices on a given manifold, in such a way that the generators of the stabilizer are local with respect to the lattice [12,14,15,28]. Typically the set of qubits and the set of generators are directly related to sets of geometric objects such as the vertices, links, or faces of the lattice. To distinguish truly topological codes from merely local ones we propose the following criterium. *In a topological stabilizer code, any operator $O \in N(\mathcal{S})$ with support in a subset of a region composed of disconnected pieces, each of them simply connected, has trivial action on logical qubits.* Stated this way, it is a rather vague criterium since no formal definition of region or connectedness is given. However, it will be enough for our purposes by adopting a reasonable interpretation when needed. Figure 1 shows an example of a region in a torus that cannot be the support of an undetectable error.

An enumeration of the common properties of known topological codes will be useful. First, the support of undetectable errors is topologically nontrivial in some well-defined way. Indeed, both in surface codes [12,13,28] and color codes [14,15] undetectable errors are related to homologically nontrivial cycles. Second and closely related, the number of encoded qubits k depends only upon the manifold in which the lattice is defined and not the lattice itself. For example, for 2D surface and color codes the number of logical qubits is, respectively, $k = 2 - \chi$ and $k = 4 - 2\chi$, with χ the Euler characteristic of the two-manifold. Finally, an important property of topological codes is that their nature makes it possible to define them
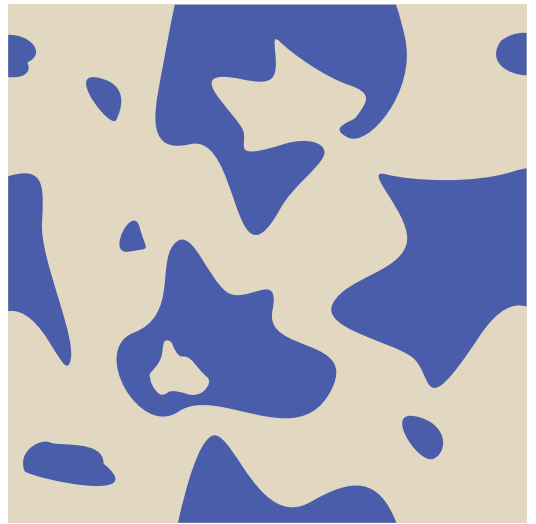

FIG. 1. (Color online) In this figure the geometry is that of a torus, with opposite sides of the square identified. The dark region can be contained in a disconnected collection of simply connected regions and thus cannot be the support of a nondetectable error in a topological code.

in many different lattices, typically rather arbitrary ones as long as some code-dependent constraints are satisfied. In other words, topological codes display a huge flexibility. This is to be expected from constructions that only see the topology, not the geometry, of the manifolds to which they are related.

Notice that the locality of the stabilizer generators was emphasized with no mention to gauge generators. The reason for this is that, up to now, no genuinely subsystem topological codes were known. This article introduces a family of such codes. They have both local gauge and local stabilizer generators. Such locality properties should be expected from any topological subsystem code.

The family of Bacon-Shor codes [16] provides an example of nontopological local gauge codes. These codes certainly do not satisfy the previous criterium for any interpretation of connectedness that agrees with their 2D lattice geometry. Moreover, their geometry is completely rigid in the sense that there is no clear way to generalize them to other lattices and manifolds.

It is interesting to observe that topological codes do not offer good d/n ratios, which go to zero as larger codes are considered. For example, in 2D surface or color codes $d = O(\sqrt{n})$ holds (which is optimal among 2D local codes [27]). But, as remarked in Ref. [12], this is a misleading point because topological codes can correct most errors of order $O(n)$.

Finally, classical topological codes also exist [28]. Unlike quantum ones they can be obtained from mere graphs, that is, one-dimensional (1D) objects.

C. Topological quantum memories

In Ref. [13] an interesting approach to the problem of indefinite preservation of quantum information was presented that makes use of topological codes. Since it will underly several discussions, a brief summary is in order. The main idea is that information is encoded in a surface code and preserved by keeping track of errors. To this end, round after round of syndrome extractions must be performed. There are thus two sources of errors since not only will the code suffer from storage errors, but the stabilizer measurements themselves are also faulty. When the error rate is below a certain threshold the storage time can be made as long as desired by making the code larger, a feature that is only available in topological codes (for other codes one will have to use concatenation). Interestingly, this error threshold can be connected to a phase transition in a classical statistical model, a random three-dimensional (3D) gauge model [13,21].

D. Code deformation

The flexibility of topological codes implies that they can be defined in many lattices. This feature makes the introduction of code deformations natural [13,20,29], which are briefly described next.

When two codes are very similar, in the sense that they differ only by a few qubits, it is possible to transform one into another by manipulating these few qubits. This is specially natural for local codes that only differ locally. In particular, such local code deformations will not compromise the protection of the encoded information. These ideas were first explored in

Ref. [13] where the geometry of a surface code is transformed step by step to compensate a change in the lattice geometry provoked by a transversal gate. In Ref. [13] code deformations are also used to initialize the code with an arbitrary state. This is done by "growing" the code from a single qubit encoding the desired state. Notice that, in this case, encoded information is not protected in the early stages of the code deformation when the code is still small.

In general [20], two main kinds of deformations may be distinguished: those that change the number of encoded qubits and those that do not. The former can be used to initialize and measure encoded qubits and the latter to perform operations on encoded qubits. In the case of topological codes, code deformations amount to changes in the geometry of a lattice, which may ultimately be understood as changes in the geometry of a manifold. When the topology of the manifold changes, initialization or measurement of encoded qubits will happen in a well-defined way [20]. When the manifold undergoes a continuous transformation that maps it to itself a unitary operation is performed on the encoded qubits [20]. This unitary operation only depends on the isotopy class of the mapping.

Code deformation can be naturally integrated with the successive rounds of stabilizer measurements mentioned in the previous section. In particular, as long as the deformations are local, one can perform them simply by changing the stabilizers to be measured at each stage of error detection [20].

E. String operators

In this and subsequent sections we only consider 2D topological codes with qubits as their basic elements because the topological subsystem codes introduced in Sec. IV fall into this category. For the same reason, subsystem code language, as opposed to subspace, will be used.

In known 2D topological codes the logical Pauli operators $\hat{X}_1, \ldots, \hat{Z}_k \in \mathbf{N}(\mathcal{G})$ can be chosen to be string operators [12, 14]. These are operators O_s with support along a set of qubits s that resemble a closed string. There are several types of strings, labeled as $\{l_i\}$. Two strings s and s' of the same type that enclose a given region, as in a and b in Fig. 2, give equivalent operators $O_s O_s' \in \mathcal{S}'$. In other words, only the homology of the strings is relevant. In particular, boundary strings, those that enclose a region as in c in Fig. 2, produce operators in $\mathcal{S}'$. Moreover, $\mathcal{S}'$ is generated by boundary strings of some minimal regions. When two strings s and s' cross once, like a and d in Fig. 2, O_s and $O_{s'}$ commute or not depending only on the labels of the strings. Finally, two strings s and s' with a common homology class can be combined in a single string s'' of a suitable type in the sense that $O_s O_{s'} O_{s''} \in \mathcal{S}'$. For example, d and e in Fig. 2 can be combined in a string f of a suitable type.

All these properties can be captured in a group $L' \simeq \mathcal{P}_t$, for some t that depends on the code. For example, $t = 1$ for surface codes and $t = 2$ for color codes. The group $L := L'/\langle i\mathbf{1}\rangle$ has, as its elements, the string types $\{l_i\}$ and the product corresponds to string combination. The commutation rules are recovered from L': crossing strings commute or not depending on whether their labels commute in L'. There are $2t$ labels $\bar{l}_1, \ldots, \bar{l}_t$ that generate L. In a given manifold $2 - \chi$ nontrivial cycles that generate the homology group can be chosen. Each
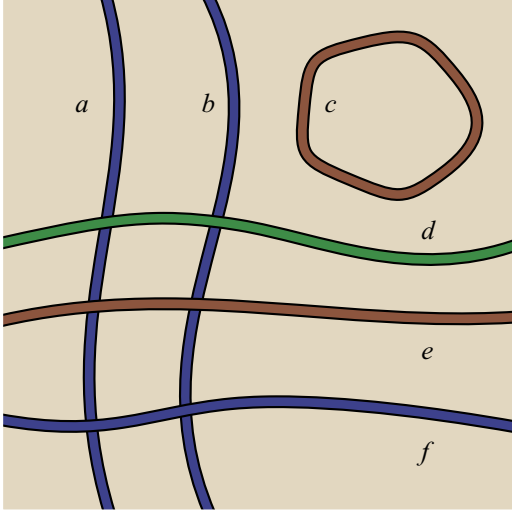
FIG. 2. (Color online) In this figure the geometry is that of a torus, with opposite sides of the square identified. The colored curves represent the support of string operators, with color standing for string labels. Strings a and b enclose a region and thus are homologically equivalent, producing equivalent operators. String c encloses a region and thus is homologically trivial, producing a stabilizer element. Strings d, e, and f are homologically equivalent but have different labels, producing inequivalent operators.

of them gives rise to $2t$ string operators with labels l_i, and the total $2t(2 - \chi)$ string operators generate $N(\mathcal{G})/\mathcal{S}'$, so that the number of encoded qubits is $k = t(2 - \chi)$.

Instead of strings, in general one can consider string-nets, where the strings meet at branching points [14]. The allowed branchings are those in which the product of all the labels involved is trivial. String-nets do not play a significant role in closed manifolds, but can be essential when the manifold has boundaries [14].

Let us check the criterium for topological codes of Sec. III B using the string operator structure. To this end a notion of connectedness is needed, but this can be obtained from the local generators of $\mathcal{S}'$: two regions or sets of qubits are disconnected from each other if no local generator has support on both of them at the same time. Let Q_O denote the support of an operator O. Then if $O \in N(\mathcal{S})$ and $Q_O = Q_1 \sqcup Q_2$ with Q_1 and Q_2 disconnected it follows that $O = O_1 O_2$ and $Q_1 = Q_{O_1}$, $Q_2 = Q_{O_2}$ for some $O_1, O_2 \in N(\mathcal{S})$. As for simple connectedness, it is easier to introduce a wider notion of the "trivial" region. A set of qubits Q forms a trivial region when there exist string operators O_i that generate $N(\mathcal{G})/\mathcal{S}'$ and such that $Q_{O_i} \cap Q = \emptyset$. If $O \in N(\mathcal{S})$ is such that Q_O is a trivial region then $[O, O_i] = 0$ for the corresponding string generators O_i and thus $O \in \mathcal{G}$. The criterium is satisfied within this language: if $O \in N(\mathcal{S})$ and $Q_O = Q_1 \sqcup \ldots \sqcup Q_t$ with the Q_i pairwise disconnected and each of the Q_i a trivial region, then $O \in \mathcal{G}$.

F. Anyons

Topological codes can be described in terms of string operators because they describe ground states of topologically ordered quantum models, that is, systems with emergent

Abelian anyons [12]. Anyons are localized quasiparticles with unusual statistics. String operators represent quasiparticle processes and their commutation rules are directly related to the topological interactions of the anyons. Moreover, when an open-ended string operator is applied to the ground state, a pair of anyons is created on the ends of the string. The labels of the created anyons are those of the string so that string labels are also quasiparticle labels.

From the perspective of the code quasiparticles, they correspond to error syndromes, signaling a chain of errors along the string [13]. Thus, keeping track of errors in a topological code, recall Sec. III C, amounts to keeping track of the word lines of these quasiparticles. Error correction will succeed if the word lines are correctly guessed up to homology [13].

G. Boundaries

From a practical perspective, codes that are local in a closed two-manifold like a torus are not very convenient. Instead, one prefers to have planar codes. Thus, a way to create nontrivial topologies in the plane is needed and this is exactly what is gained by introducing boundaries.

In a given code different types of boundaries are possible. To start with, one can always consider random, structureless boundaries. The introduction of such boundaries will typically produce many local encoded qubits along the boundary. However, these qubits are unprotected and thus essentially useless.

More interestingly, boundaries with well-defined properties and nonlocal encoded qubits are also possible [13,14]. The defining property of such boundaries is that strings s with labels from a certain subset $M \subset L$ are allowed to end in them (see Fig. 3) in the sense that O_s belongs to the normalizer $N(\mathcal{S})$. In other words, the introduction of the boundary changes the notion of a closed string by allowing on the boundary loose ends of strings of suitable types. The notion of boundary strings also changes. Two strings s and s' of the same type that, together with the boundaries in which they can end, form
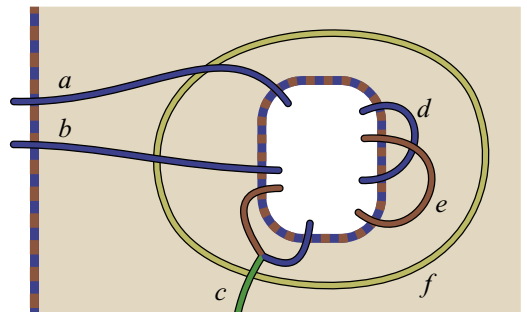

FIG. 3. (Color online) This figure illustrates boundaries on 2D topological codes, which are displayed as dashed thick lines. Strings a and b, together with the boundaries, enclose a region and thus produce equivalent operators. String c can end in the boundary because it can be decomposed in two strings that can end on it. Strings d and e enclose regions and thus produce stabilizer operators. String f produces a stabilizer element or an undetectable error, depending on whether its label is allowed in the boundary.

the the boundary of a given region (such as a and b in Fig. 3) produce equivalent operators so that $O_s O'_s \in \mathcal{S}'$.

Notice that M should be a subgroup of L because if strings with labels $l, l' \in M$ can end in the boundary then so can strings with the label ll' by splitting before reaching the boundary, such as string c in Fig. 3. Also, any two labels $l, l' \in M$ must commute in L'. In another case, the stabilizer will contain anticommuting elements, which is not possible. This is illustrated by strings d and e of Fig. 3, which must produce commuting operators. Finally, L' should be maximal in the sense that, for any $l \notin M$, there exist some $l' \in M$ such that l and l' anticommute in L'. In another case, according to the rules stated previously, an l-string s that surrounds an M hole, as in f in Fig. 3, produces an operator O_s that has to belong to $N(\mathcal{S}) - \mathcal{S}'$ because it is not a boundary, but for which there is no other string s' such that $\{O_s, O'_s\} = 0$. This is a contradiction. It is, in fact, possible to relax this last maximality condition, but at the cost of getting a boundary between two topological codes rather than a boundary between a code and the "vacuum."

Remarkably, boundaries in topological codes are directly related to anyon condensation in the corresponding topologically ordered models [30]. It will become apparent in Sec. IV E that this has important consequences because only bosons can condense and this forbids certain types of boundaries.

IV. A FAMILY OF TOPOLOGICAL SUBSYSTEM CODES

The subsystem codes introduced in this section have their origin in a spin-1/2 quantum model that shows topological order [31]. The Hamiltonian of the model is a sum of two-local Pauli terms that here will become the gauge generators. The Pauli symmetries of this Hamiltonian were already described in Ref. [31] and thus, to some extent, the codes were already implicitly considered in that work. Here we explicitly work out all the details from the code perspective. In addition, diverse aspects that are important in practice are explored, such as the possibility of introducing boundaries and the computational power of code deformations.

A. Lattice and gauge group

The family of codes C_Λ of interest is parametrized by tripartite triangulations Λ of closed two-manifolds, not necessarily orientable. That is, each code C_Λ is obtained from a 2D lattice Λ such that (i) all faces $f \in F$ are triangular and (ii) the set of vertices V can be separated in three disjoint sets, in such a way that no edge $e \in E$ connects two vertices from the same set. Figure 4(a) shows an example. Alternatively, Λ is the dual of a two-colex [32]. Following the notation used in previous work, the three sets of vertices are colored as red, green, and blue. The faces of Λ will be simply called triangles.

The first step in the construction of C_Λ is to derive a new lattice $\bar{\Lambda}$ from Λ, as exemplified in Fig. 4(b). In going from Λ to $\bar{\Lambda}$, the triangles of Λ separate from each other giving rise to new faces. In particular, each of the edges and vertices of Λ contributes a face to $\bar{\Lambda}$. The edges of Λ are divided into three subsets, $\bar{E} = \bar{E}_X \sqcup \bar{E}_Y \sqcup \bar{E}_Z$. In Fig. 4(b), X edges are dotted, Y edges are dashed, and Z edges are solid. The Z edges form the triangles of $\bar{\Lambda}$. Each edge in Λ contributes an X edge and
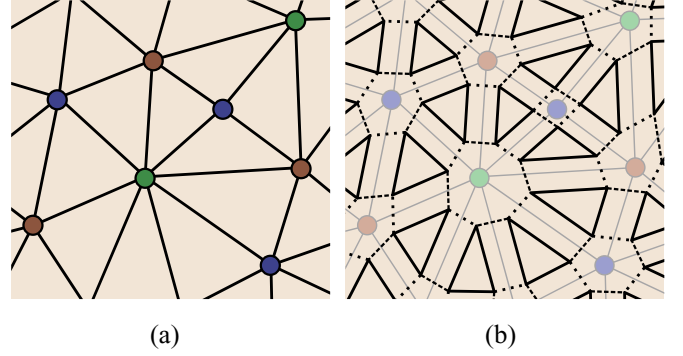

FIG. 4. (Color online) (a) Part of a 2D lattice Λ with triangular faces and three-colorable vertices. (b) The lattice $\bar{\Lambda}$ derived from Λ. It is obtained by separating the triangles of Λ and adding one face per edge and vertex of Λ. Its edges are classified in three types. Solid edges are Z edges, dashed edges are Y edges, and dotted edges are X edges. There is one qubit per vertex and the generators of the gauge group are related to edges. They are two-local operators of the form XX, YY, or ZZ, depending on the edge type.

a Y edge in such a way that no two X edges or two Y edges meet. There are thus two ways to choose the sets of X and Y edges.

The definition of C_Λ is now at hand. First, physical qubits correspond to the vertices of $\bar{\Lambda}$. Second, the gauge group is $\mathcal{G}_\Lambda := \langle i\mathbf{1}\rangle\langle G_e\rangle_{e\in E}$, with generators G_e related to the edges e of $\bar{\Lambda}$. These take the form $G_e := \sigma_v \sigma_{v'}$ for $e \in \bar{E}_\sigma$, $\sigma = X, Y, Z$, where v, v' are the vertices connected by e. Thus, the generators are two-local. This is an improvement with respect to previously known topological stabilizer codes, which have generators of weight of at least 4.

B. String operators

This section describes $N(\mathcal{G}_\Lambda)$ and its center $\mathcal{S}'_\Lambda = \mathcal{G}_\Lambda \cap N(\mathcal{G}_\Lambda)$ in terms of the string operator framework of Sec. III E, which is valid for these codes. In particular, it turns out that $L' \simeq \mathcal{P}_1$, as in surface codes. In Sec. IV E it will be apparent, however, that there exist differences between the nature of the strings of C_Λ codes and those in surface codes. These are not captured by L', which indeed does not contain all the information about the corresponding topological order. The details of the statements made in this section can be found in the Appendix.

We first seek a graphical representation of $N(\mathcal{G}_\Lambda)$. Take any subgraph γ of the graph of $\bar{\Lambda}$, such as the one in Fig. 5(b) that has at each of its vertices one of the configurations of Fig. 5(a). This graph γ produces a Pauli operator $O_\gamma = \bigotimes_v \sigma_v$ with $\sigma_v = \mathbf{1}, X, Y, Z$ according to the correspondence of Fig. 5(a). Observe that such operators O_γ belong to $N(\mathcal{G}_\Lambda)$. Up to a phase, the correspondence between the elements of $N(\mathcal{G}_\Lambda)$ and graphs is one to one.

These graphs either contain all the edges of a triangle or none of them. Thus, each graph γ determines a subset of triangles T_γ of the original lattice Λ. In Figs. 5 through 7 this subset appears shaded. Notice that the number of triangles of T_γ meeting at each vertex is even. In fact, any subset of
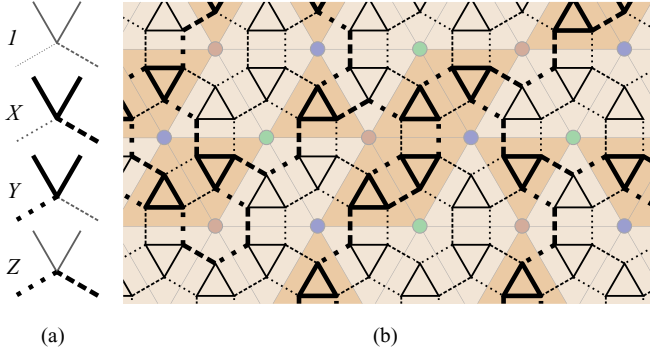
FIG. 5. (Color online) (a) The four possible configurations at a given vertex for allowed subgraphs of $\bar{\Lambda}$. A different Pauli operator corresponds to each of them. (b) A subgraph γ (thick lines) of a lattice $\bar{\Lambda}$ (thin lines), obtained from a regular triangular lattice Λ (lightest lines).

triangles that meets this property can be realized as T_γ for some γ.

String operators are obtained from string-like graphs such as the one in Fig. 6. Notice in the figure how triangles can be paired in a specific way. These pairs of triangles always connect vertices of the same color from the original lattice Λ. This allows us to classify strings accordingly with the labels r, g, and b. It is a simple exercise to check that crossing strings operators commute if they have the same color and anticommute in the other case, in accordance with L'. String-nets can be formed by introducing branching points where three strings of different colors meet.

The group $\mathcal{S}'_\Lambda$ is generated by small string operators related to vertices v of the original lattice Λ. In particular, let us set $S_v^c = O_\gamma$, $c = $ r, g, b with γ the c-colored string going around v, as shown in Fig. 7. Then $\mathcal{S}'_\Lambda = \langle i\mathbf{1} \rangle \langle S_v^c \rangle_{v,c}$. These generators are only subject to the relations

$$\prod_c S_v^c \propto \mathbf{1}, \quad \prod_v S_v^c \propto \mathbf{1}, \tag{5}$$

where the first product runs over the three colors and the second over the vertices of Λ. As a consequence, the rank of $\mathcal{S}_\Lambda$ is $s = 2|V| - 2$. Since the number of encoded qubits is $k = 2 - \chi$, it follows that the number of gauge qubits is $r = n - k - s = |3F| - 2|V| + \chi = 2|F| - \chi$, showing that gauge qubits "see" the global structure of the manifold.
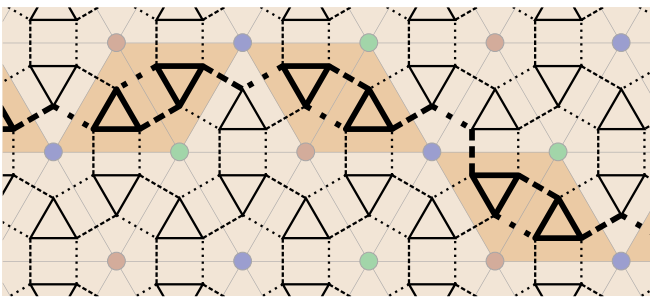

FIG. 6. (Color online) A string operator as a subgraph γ of $\bar{\Lambda}$, displayed in thick lines. Its triangles come in pairs, each of them connecting vertices of the same color in Λ.
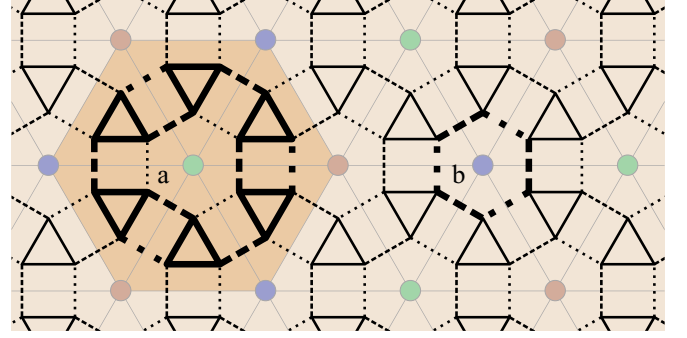

FIG. 7. (Color online) Two examples of string operators S_v^c related to vertices v in Λ. The string b shares color with the vertex that encloses, whereas for a the two colors differ producing a more involved operator.

What can string operators tell us about the code distance d? Given an operator $O_\gamma \in N(\mathcal{G})$, consider the subset E' of the edges of γ with elements of all the X and Y edges of γ and one of the three Z edges that correspond to each triangle in T_γ. Then $G = \prod_{e \in L} G_e \in \mathcal{G}_\Lambda$ and it is easy to check that $|O_\gamma G| = |T_\gamma|$. Therefore $d \leqslant d_T$, with d_T the minimal length, in terms of the number of triangles, among nontrivial closed strings. A lower bound for d is given in the next section.

C. Homology of errors

This section offers a homological description of error correction for C_Λ. The main idea is that the error syndrome can be identified with the boundary of errors, considered as paths on the surface. Then error correction succeeds if this path can be guessed up to homology. It is worth noting that the notation and results in this section will not be used again.

To fix notation we recall first some basic notions. Let Δ denote the additive group of $\mathbf{Z}_2$ one-chains in Λ. Its elements are sets of edges $\delta \subset E$ and addition is given by $\delta + \delta' = (\delta \cup \delta') - (\delta \cap \delta')$. The boundary $\partial\delta$ of $\delta \in \Delta$ is the set of vertices in which an odd number of edges from δ meet. The elements $\delta \in \Delta$ with $\partial\delta = 0$ are called cycles and form a subgroup $Z \subset \Delta$. Boundaries form a subgroup $B \subset Z$, generated by elements of the form $\delta = \{e_1, e_2, e_3\}$ with e_i the three edges of a given triangle. The first $\mathbf{Z}_2$ homology group of Λ is $H_1 := Z/B \simeq \mathbf{Z}_2^h$ with $h = 2 - \chi$ the number of independent nontrivial cycles of the closed surface formed by Λ. Two chains $\delta, \delta' \in \Delta$ are said to be equivalent up to homology, $\delta \sim \delta'$ if $\delta + \delta' \in B$.

Consider a morphism $f_r : \mathcal{P}_n \longrightarrow \Delta$ defined by $f_r(i) = \emptyset$ and the following action on single qubit operators $X_{\bar{v}}, Y_{\bar{v}}$, where $\bar{v} \in \bar{V}$. $X_{\bar{v}}$ anticommutes exactly with two operators of the form S_v^r, $v \in V$. The corresponding two vertices are connected by an edge $e \in E$ and $f_r(X_{\bar{v}}) = \{e\}$. $f_r(Y_{\bar{v}})$ is defined analogously. It is easy to check that $f_r[\mathcal{G}_\Lambda] = B$ and that for any $O \in \mathcal{P}_n$, the set $\partial f_r(O)$ contains those vertices $v \in V$ such that $\{O, S_v^r\} = 0$. Moreover, if γ is a string then $f_r(O_\gamma) \in Z$ and if γ is red $f_r(O_\gamma) \in B$. Indeed, if $\{\gamma_i\}$ is the set of red strings, then f_r gives an isomorphism $N(\mathcal{S}_\Lambda)/(\mathcal{G}_\Lambda \langle O_{\gamma_i} \rangle_i) \simeq H_1$.

Consider, in addition, an analogous morphism f_b with blue color playing the same role as red in f_r. Then for any

$O \in \mathcal{P}_n$, we have $O \in \mathcal{G}_\Lambda$ if and only if $f_c(O) \in B$ for $c = $ r, b. Similarly, $O \in N(\mathcal{S}_\Lambda)$ if and only if $f_c(O) \in Z$ for $c = $ r, b. This shows that error correction will succeed as long as errors can be guessed up to homology. In detail, suppose that the code suffers a Pauli error O. The error syndrome can be expressed in terms of the two sets $\partial f_r(O)$, $\partial f_b(O) \subset V$. Suppose that an attempt is made to correct the errors by applying some $O' \in \mathcal{P}_n$ such that $\partial f_c(O) = \partial f_c(O')$ for $c = $ r, b. Then error correction succeeds if and only if $O'O \in \mathcal{G}$, that is, if and only if $f_c(O) \sim f_c(O')$ for $c = $ r, b.

Although error correction can be expressed in these homological terms, this is really not the most natural thing to do because it involves an arbitrary choice of two of the three available colors. In this regard, notice that not any set of edges $\delta \in \Delta$ can be obtained from an operator $O \in \mathcal{P}_n$ as $\delta = f_r(O)$ and that the cardinalities of $f_r(O)$ and $f_b(O)$ by no means are enough to compute $|O|$. This makes a direct translation of the ideas used in Ref. [13] unfeasible for error correction in surface codes.

To give a lower bound for the distance d of the code, the definition of the mappings f_c, $c = $ r, b must be modified. We set $f_c'(\bigotimes_{\bar{v}} \sigma_{\bar{v}}) := \sum_{\bar{v}} f_c'(\sigma_{\bar{v}})$, where $\sigma_{\bar{v}} = \mathbf{1}_{\bar{v}}, X_{\bar{v}}, Y_{\bar{v}}, Z_{\bar{v}}$, and we fix $f_c'(\mathbf{1}_{\bar{v}}) := \emptyset$, $f_c'(X_{\bar{v}}) := f_c(X_{\bar{v}})$, $f_c'(Y_{\bar{v}}) := f_c(Y_{\bar{v}})$ and $f_c'(Z_{\bar{v}})$ is defined in analogy with $f_c(X_{\bar{v}})$. The new mappings f_c' are not group morphisms, but they do keep the good properties of the f_c mappings listed previously. And they satisfy $|O| \geqslant |f_c'(O)|$, which immediately leads to the bound $d \geqslant d_L$ with d_L the minimal length, in terms of the number of edges, among nontrivial closed loops in Λ.

D. Syndrome extraction

As indicated in Sec. III C, in a topological quantum memory one has to keep track of errors by performing round after round of syndrome extraction. This raises the question of how fast and simply the stabilizer generators of a code C_Λ can be measured. The faster the measurements the less errors will be produced in the meantime and the simpler they are the less faulty gates they will involve. Of course, what fast and simple really mean will depend on particular implementations; in other words, in the basic operations at our disposal.

To keep the discussion general, take gauge generator measurements to be the basic components of the syndrome extraction. At each time step the measurement of any subset of generators $\{G_i\}$ is allowed as long as each physical qubit only appears in one of the G_i. Then, in any code C_Λ it is possible to cyclically measure all the stabilizer generators by performing six rounds of measurements. The time step at which each generator is to be measured is indicated in Fig. 8. Notice that Z edges are measured at even times and X and Y edges at odd times. From time steps 1 through 3 the eigenvalue for operators S_v^c at blue vertices are obtained, from steps 3 through 5 those for red vertices, and from steps 5, 6, and 1 (this last one in the subsequent cycle) those for green vertices. It is not clear whether this number of time steps is optimal since, in principle, four or five can be enough. As a comparison, the number of steps needed for the Bacon-Shor codes is four. In this sense the six steps are not bad, taking into account that the codes C_Λ do not benefit from the the separation of gauge and stabilizer generators into X type and Z type as the Bacon-Shor codes do.
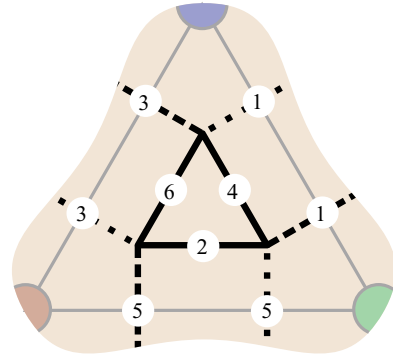

FIG. 8. (Color online) The proposed ordering for the measurements of the edge operators. It does not depend on the particular geometry of the lattice Λ because it is dictated by the coloring of its vertices.

E. The problem of boundaries

This section shows why it is not possible to introduce boundaries with the properties discussed in Sec. III G. This has important practical consequences since there is no other known way to introduce a nontrivial topology in a completely planar code. Notice, however, that we can always flatten a manifold to get a "planar" code at the price of doubling the density of physical qubits in the surface. Also, the absence of boundaries makes the use of code deformations less practical, although they are still possible, as shown in Sec. IV F. In any case, this leaves open the question of whether other kinds of interesting boundaries can be introduced.

The existence of boundaries leads to the following contradiction. According to the properties listed in Sec. III G, there are three potential kinds of boundaries, one per color. Each of them only allows strings of its color to end on it. Clearly either all the boundaries can be constructed or none of them. Thus, suppose that the three of them are allowed and consider a geometry like the one in Fig. 9, with three holes, one of each color. Take a string-net γ that connects the three holes, as in the figure, and deform it to another string-net γ'. It follows from the properties of string operators and boundaries that $O_\gamma O_{\gamma'} \in \mathcal{S}'$, but also that $\{O_\gamma, O_{\gamma'}\} = 0$ since they cross at a single point where they have different colors. This is not possible.
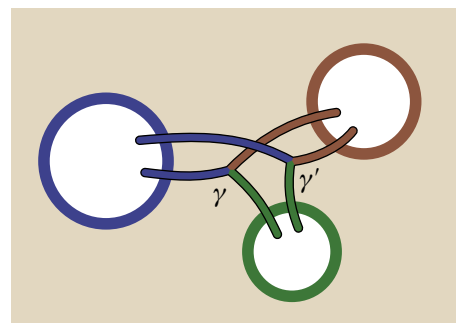

FIG. 9. (Color online) A hypothetical geometry for a C_Λ code with three holes of different colors. According to the properties of boundaries, the string operators a and b are equivalent up to stabilizer elements. This is a contradiction because, due to the way they cross, they anticommute.

In Sec. IV B it was noted that the string label group L' is the same in surface codes and the subsystem codes. Since, according to Sec. III G, the set of allowed boundaries is dictated by L', it can be expected that surface codes will not have boundaries. However, this is not the case: Two kinds of boundaries can be constructed in surface codes [13]. The point is that there is a key difference between the two families of codes: In surface codes the three types of strings are not equivalent in any sense so that the previous reasoning is not valid.

At a deeper level, this difference between the codes has its origin in the difference between the corresponding topological orders. Indeed, L' does not encode all the information about the properties of anyons. In surface codes two of the quasiparticle types are bosons and the third a fermion [12], whereas in the subsystem codes all three are fermions [31]. The connection between anyon condensation and boundaries is thus crucial: Nice boundaries cannot be introduced in these topological subsystem codes because all the string operators are related to fermions, which cannot condense.

F. Code deformation

This section explores the potential of code deformations in the topological subsystem codes C_Λ. We show how initializations and measurements of individual logical qubits in the X and Z basis are possible through certain topology-changing processes on the manifold. We also show that controlled-NOT and Hadamard gates can be, in principle, implemented through continuous deformations of the manifold, but not in a practical way.

To begin with, a manifold and a set of logical operators $\hat{X}_1, \hat{Z}_1, \cdots \hat{X}_k, \hat{Z}_k$ must be selected. We choose an h torus, that is, a sphere with h holes. Codes C_Λ on such a manifold provide $2h$ logical qubits, but only h of them will be used with the choice of logical operators indicated in Fig. 10(a). The rest of the logical qubits are considered gauge qubits.

When a new handle is introduced, a logical qubit is created and initialized in a definite way [see Fig. 10(b)]. In the figure the two surfaces are supposed to be already connected so that a handle is really created. There are two ways to introduce a new handle in a surface such as that of Fig. 10(a), depending on
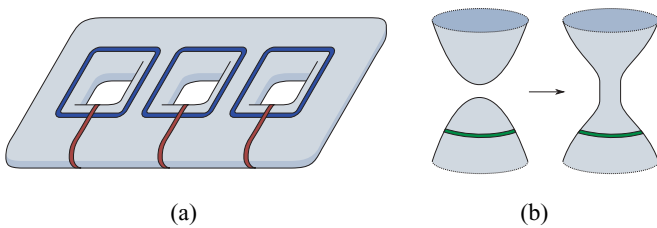
whether the process of Fig. 10(b) occurs "inside" or "outside" the surface. In the former case the new qubit is initialized in a $\hat{Z}$ eigenstate and in the latter in a $\hat{X}$ eigenstate. Whether the initialization occurs in the Z or X basis depends on which of the two string operators of the new qubit was a boundary initially. This operator has its eigenvalue fixed before the deformation occurs and during the process it is topologically protected at all times [20]. The particular sign of the eigenstate depends on the arbitrary sign choices for the logical operators and S_Λ. If the initialization process is reverted it yields a measurement in the corresponding basis [20].

It is always possible to detach a qubit or a torus from the rest of the code. This does not involve any measurement because the strings running along the cutting line are boundaries [20]. Similarly, there is no problem in attaching a torus to the code to add a logical qubit. But once a logical qubit is isolated it can undergo code deformations independently. Consider a mapping that exchanges the two principal cycles of the torus and shifts the lattice a bit, if necessary, to adjust the color correspondence, for example, by rotating the torus. Such a mapping can exchange $\hat{X}$ and $\hat{Z}$ operators, which amounts to a Hadamard gate. There exists an important drawback though. This deformation cannot be realized in 3D without producing self-intersections of the surface. Still, it is conceptually interesting that the Hadamard gate can be obtained from purely geometric code deformations because this is not possible in surface or color codes where X and Z-type operators correspond to different types of string operators and a transversal Hadamard gate must be added to the picture [13]. Because color is just a matter of location in the lattice, strings of different colors are equivalent up to lattice translations. This is, in essence, what makes the geometric implementation of the Hadamard gate possible.

A controlled phase gate $\mathbf{1} - \frac{1}{2}(1 - Z_i)(1 - Z_j)$ on a pair of logical qubits i, j can be implemented through a "continuous" deformation of the code. The process is indicated in Fig. 11. It follows from the way in which the logical operators evolve [20] that the complete process amounts to a controlled phase gate, up to some signs in the final logical operators that depend on the choice of S_Λ. A controlled-NOT gate can then be obtained by composing this gate with Hadamard gates. But again, such a code deformation requires the overlapping of the surface of the code with itself, see Fig. 11(b).

(a)   (b)

FIG. 10. (Color online) (a) A sphere with h holes can encode $2h$ qubits, but we choose to encode just h. The logical $\hat{X}_i$, $\hat{Z}_i$ operators correspond to the strings in the figure. Red strings give X's and blue strings Z's. (b) When the topology of the surface changes as indicated here two qubits are introduced in the code. They are initialized in a fixed way. In particular, the string operator in the figure is a boundary before the deformation takes place and thus has a fixed value. This value is not changed by the deformation because it occurs in a different part of the code.

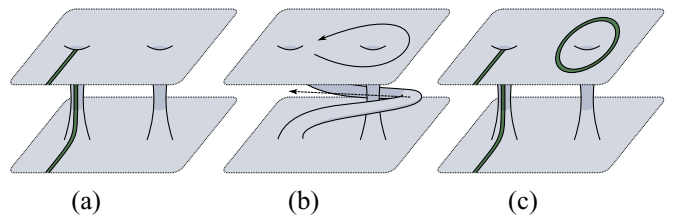(a)   (b)   (c)

FIG. 11. (Color online) The deformation that produces a controlled phase gate. (a) The code before the deformation takes place and a particular string. (b) The deformation moves one of the "holes" in the top part around the other, as indicated by the solid line with an arrow. To recover the original shape, as indicated by the dashed line, the two "tubes" have to overlap unavoidably. (c) After the deformation, the string operator was mapped to the product of these two string operators.

V. STATISTICAL PHYSICS OF ERROR CORRECTION

In Ref. [13] an interesting connection between error correction thresholds for surface codes and phase transitions in 2D random bond Ising models was developed. Similar mappings exist also for color codes [22], in this case to 2D random three-body Ising models. In both cases, the CSS structure of these codes is an important ingredient in the constructions: They are subspace codes with $\mathcal{S} = \mathcal{S}_X \mathcal{S}_Z$ in such a way that $\mathcal{S}_\sigma$ is generated by products of σ operators, $\sigma = X, Z$. To take full advantage of this the noise channel for each qubit must be a composition of a bit-flip channel $\mathcal{E}_{\mathrm{bf}}(p) := \{(1 - p)^{\frac{1}{2}}\mathbf{1}, p^{\frac{1}{2}}X\}$ and a phase-flip channel $\mathcal{E}_{\mathrm{pf}}(p) := \{(1 - p)^{\frac{1}{2}}\mathbf{1}, p^{\frac{1}{2}}Z\}$.

There are two main obstacles to constructing a similar mapping for the codes C_Λ. The first is that they are subsystem codes rather than subspace codes. The second is that the gauge group cannot be separated in an X and a Z part. As we show in the following both can be overcome.

A. Mapping to a statistical model

Rather than directly considering the codes C_Λ, this section deals with the general mapping from any given stabilizer subsystem code to a suitable classical statistical model. For simplicity, each qubit in the code is supposed to be subject to a depolarizing channel $\mathcal{E}_{\mathrm{dep}}(p) := \{(1 - p)^{\frac{1}{2}}\mathbf{1}, (p/3)^{\frac{1}{2}}X, (p/3)^{\frac{1}{2}}Y, (p/3)^{\frac{1}{2}}Z\}$, with p the error probability, but more general channels are possible within the same framework.

To build the classical Hamiltonian model, the first step is the choice of a set of generators $\{G_i\}_{i=1}^l$ of $\mathcal{G}/\langle i\mathbf{1}\rangle$. These generators can be captured in a collection of numbers $g_{ij}^\sigma = 0, 1$ defined by

$$G_i \sigma_j = (-1)^{g_{ij}^\sigma} \sigma_j G_i, \tag{6}$$

with $\sigma = X, Y, Z$, $i = 1, \ldots, l$, and $j = 1, \ldots, n$. Attach a classical Ising spin $s_i = \pm 1$ to each of the generators G_i. The family of Hamiltonians of interest is

$$H_\tau(s) := -J \sum_{\sigma=X,Y,Z} \sum_{j=1}^n \tau_j^\sigma \prod_{i=1}^l s_i^{g_{ij}^\sigma}, \tag{7}$$

with parameters $\tau_j^\sigma = \pm 1$ such that $\tau_j^X \tau_j^Y \tau_j^Z = 1$. The coupling $J > 0$ is introduced to follow conventions. Notice that codes with local gauge generators give rise to local Hamiltonian models. Since $g_{ij}^X + g_{ij}^Y + g_{ij}^Z = 0 \bmod 2$ the Hamiltonian (7) can be rewritten as

$$H_\tau(s) = n - \sum_j \left(1 + \tau_j^X \prod_i s_i^{g_{ij}^X}\right)\left(1 + \tau_j^Y \prod_i s_i^{g_{ij}^Y}\right). \tag{8}$$

The partition function for these Hamiltonians is

$$Z(K, \tau) = \sum_s e^{-\beta H_\tau(s)}, \tag{9}$$

with $K := \beta J$ and β the inverse temperature.

The goal is to express the class probabilities $p(\bar{E})$, $E \in \mathcal{P}_n$ in terms of the partition function (9) for a suitable τ. Let $\tau = \tau_E$

be such that

$$E \propto \bigotimes_j X^{\frac{1-\tau_j^Y}{2}} Y^{\frac{1-\tau_j^X}{2}}. \tag{10}$$

Similarly, for each $G \in \mathcal{G}$ choose any $s = s_G$ such that

$$G \propto G_1^{\frac{1-s_1}{2}} \cdots G_l^{\frac{1-s_l}{2}}. \tag{11}$$

We write $s'' = s's$ if $s_j'' = s_j' s_j$. Then $s_G s_{G'} = s_{GG'}$ and for any spin configuration s, $E \in \mathcal{P}_n$ and $G \in \mathcal{G}$ it can be checked that

$$H_{\tau_{EG}}(s) = H_{\tau_E}(s_G s). \tag{12}$$

In the depolarizing channel the probability for a Pauli error E is $p(E) = (p/3)^{|E|}(1 - p)^{n-|E|}$. It may be written as

$$p(E) = c_p^{-n} e^{-\beta_p H_{\tau_E}(s_1)}, \tag{13}$$

where $c_p := e^{3K_p} + 3e^{-K_p}$ and $\beta_p := K_p/J$ with

$$e^{-4K_p} := \frac{p}{3(1 - p)}. \tag{14}$$

The desired connection follows from Eqs. (12) and (13), which give

$$p(\bar{E}) = \sum_{G \in \mathcal{G}} p(EG) = \frac{1}{2^w c_p^n} Z(K_p, \tau_E), \tag{15}$$

where w is the number of redundant generators of $\mathcal{G}$, that is, $w = l - l'$ with l' the rank of $\mathcal{G}/\langle i\mathbf{1}\rangle$.

B. CSS-like codes

To connect the results of the previous section with the work in Refs. [13,21,22] CSS codes must be considered. These are codes with $\mathcal{G} = \langle i\mathbf{1}\rangle \mathcal{G}_X \mathcal{G}_Z$ for some $\mathcal{G}_\sigma$ generated by products of σ operators, $\sigma = X, Z$. And, instead of a depolarizing channel, the noisy channel must take the form $\mathcal{E} = \mathcal{E}_{\mathrm{bf}}(p) \circ \mathcal{E}_{\mathrm{pf}}(p')$. This allows us to treat the X and Z errors independently [13]. Here we consider the case of bit-flip errors; phase-flip errors are analogous.

The construction is similar to the one in the previous section. It starts with the choice of generators $\mathcal{G}_X = \langle G_i \rangle_{i=1}^l$. The relevant Hamiltonians read

$$H_\tau' := -J \sum_{j=1}^n \tau_j \prod_{i=1}^l s_i^{g_{ij}}, \tag{16}$$

where $\tau_j := \tau_j^Z$ and $g_{ij} := g_{ij}^Z$. The probability of an error E that is a product of X operators is

$$p(\tilde{E}) := \sum_{G \in \mathcal{G}_X} p(EG) = \frac{1}{2^w (2 \cosh K_p')^n} Z(K_p', \tau_E), \tag{17}$$

where w is the number of redundant generators of $\mathcal{G}_X$ and

$$e^{-2K_p'} := \frac{p}{1 - p}, \tag{18}$$

defines the Nishimori temperature [26].

The Bacon-Shor codes provide an example of gauge CSS-like codes. With the above procedure they yield models that amount to several copies of the 1D Ising model.

C. Symmetries

Interestingly, the redundancy of the generators of $\mathcal{G}$ is directly connected to the symmetries of the Hamiltonian (7). Suppose that the generators are subject to a constraint of the form

$$\prod_{i \in I} G_i \propto \mathbf{1}, \tag{19}$$

for some set of indices I. Then Eq. (12) gives

$$H_\tau(s) = H_\tau \left(s \prod_{i \in I} s_{G_i} \right). \tag{20}$$

In other words, making the most natural choice for the s_{G_i} it follows that the Hamiltonian is invariant under the transformation

$$s_i \longrightarrow s_i' = \begin{cases} -s_i, & i \in I, \\ s_i, & i \notin I. \end{cases} \tag{21}$$

Thus, global constraints lead to global symmetries and local constraints to local symmetries.

As a particular example, consider surface codes, which are mapped to Ising models [13]. In these codes the product of all X-type stabilizers equals the identity, producing a symmetry that is simply the global $\mathbf{Z}_2$ symmetry of the Ising model.

D. Error correction and free energy

We now put Eq. (15) to use in the error correction framework of Sec. II E. Recall that, after the syndrome has been measured, one has to find the most probable class of errors among several candidates $\bar{E}_i := \bar{E} \bar{D}_i$. This amounts to comparing the probabilities $p(\bar{E}_i)$ or, alternatively, the quantities $Z(K_p, \tau_{E_i})$. To do this, it is enough to know the free energy differences [13]

$$\Delta_i(K_p, \tau_E) := \beta F(K_p, \tau_{E_i}) - \beta F(K_p, \tau_E), \tag{22}$$

where $F(K, \tau) = -T \ln Z(K, \tau)$ is the free energy of a given interaction configuration τ. For example, in the Ising models that appear for 2D surface and color codes these are domain-wall free energies.

In practice, the computation of Eq. (22) may be difficult. In this regard, it was suggested in Ref. [13] in the context of surface codes that, in the absence of glassy behavior, the computation of Eq. (22) should be manageable and in Ref. [21] a possible approach was sketched.

E. Error threshold and phase transition

In surfaces codes there exists an error probability p_c, the error threshold such that the asymptotic value of the success probability p_0, in the limit of large code instances, is one for $p < p_c$ and $1/4^k$ for $p > p_c$ [21]. This is directly connected to an order-disorder phase transition in a model with random interactions. An analogous transition is observed for the random model that corresponds to color codes [22,33,34]. It is then natural to expect a similar connection in other topological codes, as we describe next.

Consider a random statistical model with Hamiltonian (7) in which the parameter τ is a quenched random variable. That is, τ is random but not subject to thermal fluctuations. The probability distribution $p(\tau)$ is such that the signs of τ_i^σ and $\tau_j^{\sigma'}$ are independent if $i \neq j$. For each i, the case $\tau_i^X = \tau_i^Y = 1$ has probability $1 - p$ and the other cases have probability $p/3$ each. In other words, if $\tau = \tau_E$ then $p(\tau) = p(E)$ with $p(E)$ given by the depolarizing channel $\mathcal{E}_{\text{dep}}(p)$.

In thermal equilibrium the model has two parameters, the temperature T and the probability p. For mapping only a particular line in the p-T plane is relevant, the Nishimori line [26], given by the condition $K = K_p$ that has its origin in Eq. (13). The error correction success probability in Eq. (3) can be written in terms of this statistical model as follows

$$p_0' = \left[\left(1 + \sum_{i=2}^{4k} e^{-\Delta_i(K_p, \tau)} \right)^{-1} \right]_{K_p}, \tag{23}$$

where $[\cdot]_{K_p} := \sum_\tau p(\tau) \cdot$ denotes the average over the quenched variables.

Suppose that the code has a threshold probability p_c below which $p_0' \to 1$ in the limit of large codes. Then [35], in the random model the average of the free energy difference (22) diverges with the system size, $[\Delta_i(K, \tau)]_{K_p} \to \infty$, for $p < p_c$ along the Nishimori line. This is exemplified [13] by surface codes and the corresponding random 2D Ising models, where $[\Delta_i(K, \tau)]_{K_p}$ is the domain-wall free energy. It diverges with the system size below $p = p_c$ and attains some finite limit over the threshold, signaling an order-disorder phase transition at p_c. A similar behavior can be expected for other topological codes. For 2D color codes this was shown in Ref. [22].

F. The Hamiltonian model for C_Λ codes

The previous mapping can be immediately applied to the subsystem codes C_Λ. Choose as generators of the gauge group the edge operators O_e so that there is an Ising spin s_e at each edge e. The Hamiltonian takes the form

$$H_\tau^\Lambda(s) := -J \sum_{j=1}^n \tau_j^X s_2 s_3 s_4 + \tau_j^Y s_1 s_3 s_4 + \tau_j^Z s_1 s_2, \tag{24}$$

where the sum runs over vertices and for each of them the Ising spins s_1, s_2, s_3, and s_4 correspond, respectively, to the X, Y, and two Z edges meeting at the vertex.

The Hamiltonian (24) has a local symmetry at each triangle. In particular, flipping the three Ising spins of the triangle leaves H_τ^Λ invariant. This is so because the product of the three edge operators in the triangle equals the identity. There exists also a global $\mathbf{Z}_2 \times \mathbf{Z}_2$ symmetry that follows from the global constraints in Eq. (5). The local constraints in Eq. (5) do not provide any symmetry as they are trivial in terms of the gauge generators.

G. Faulty measurements

The mapping considered up to now is only suitable if perfectly accurate quantum computations are allowed in error correction. This section generalizes it to include errors in the measurements of the stabilizer generators.

Following Ref. [13], take as a goal the "indefinite" preservation of the content of a quantum memory. Time is divided in discrete steps. At each time step, the memory suffers errors and at the same time the stabilizer generators are imperfectly

measured. Then, if from the history of measurements one can correctly infer the actual history of errors up to a suitable equivalence, the memory is safe.

The results in Refs. [13,21] show that, for surface codes, there exists a noise threshold below which long time storage is possible for sufficiently large codes. The same behavior can be expected for other topological codes, but the construction of a suitable random statistical model for each code is required first. Here we generalize the construction of Ref. [13] to subsystem codes and depolarizing channels.

1. Depolarizing channel

Consider first the case of a depolarizing channel $\mathcal{E}_{\text{dep}}(p)$ occurring for each physical qubit between each round of measurements. We adopt the convention that, at a given time t, first errors occur and then faulty measurements are performed.

Recall that in the mapping of error correction to a statistical model errors were mapped to interactions through the τ_j^σ [see Eq. (10)]. The new elements here are time and faulty measurements. Since errors can occur at different time steps t, a time label must be added to the τ_j^σ's to get the collection of signs $\tau = (\tau_{jt}^\sigma)$, subject as before to the constraints $\tau_{jt}^X \tau_{jt}^Y \tau_{jt}^Z = 1$. To represent errors in the measurements of stabilizers, first a set $\{S_k\}_{k=1}^m$ of generators of $\mathcal{S}$ to be measured at each time step t must be chosen. Attach to them a collection of signs $\kappa_{kt} = \pm 1$. The correct (wrong) measurement of the kth generator at time t corresponds to $\kappa_{kt} = 1$ ($\kappa_{kt} = -1$). In the statistical model the τ and κ are quenched variables. τ follows the same distribution as before, dependent on the probability p and each κ_{kt} is independent and takes the value -1 with probability q. For this to make sense under the mapping, errors in the measurements must occur independently with a fixed probability q. This will not be true in most settings. Still, it is a useful assumption because knowing the correlations between errors can only improve error correction. In analogy with the g_{ij}^σ defined earlier, the stabilizer generators are captured in a collection of numbers $h_{kj}^\sigma = 0, 1$ defined by

$$S_k \sigma_j = (-1)^{h_{kj}^\sigma} \sigma_j S_k, \qquad (25)$$

with $\sigma = X, Y, Z$, $k = 1, \ldots, m$, and $j = 1, \ldots, n$.

Recall also that, in the original mapping, gauge generators G_i were mapped to Ising spins s_i. The reason for this was that gauge generators play the role of basic equivalences between errors. Now for errors that occur at the same time t this kind of equivalence happens again, represented by spins s_{it}. In addition there is an equivalence between errors that involves errors at different time and measurement errors. If at times t and $t + 1$ a given error occurs and the measurements at time t of the stabilizers that will detect the error fail, then these errors go unnoticed altogether but produce no harm. Thus, two collections of errors that differ only by such an event should be considered equivalent. Therefore, Ising spins that represent this equivalence are necessary. This can be achieved by attaching two Ising spins s_{jt}^X, s_{jt}^Y to the tth time step and jth

qubit. The Hamiltonians are

$$
H_{\tau,\kappa}(s) := -J \sum_\sigma \sum_j \sum_t \tau_{jt}^\sigma s_{j(t-1)}^\sigma s_{jt}^\sigma \prod_i s_{it}^{g_{ij}^\sigma}
$$
$$
- K \sum_k \sum_t \kappa_{kt} \prod_j \prod_\sigma \left(s_{jt}^\sigma\right)^{h_{kj}^\sigma}, \qquad (26)
$$

where $s_{jt}^Z := s_{jt}^X s_{jt}^Y$ and the range of values of the different indices should be clear from the context. To recover the probability of a given set of errors from the partition function the relations

$$
e^{-4\beta J} = \frac{p}{3(1-p)}, \qquad e^{-2\beta K} = \frac{q}{1-q}, \qquad (27)
$$

must hold.

For each time step t, the Hamiltonians (26) keep the symmetries (21). In addition, there is a symmetry for each gauge generator $G_{i'}$ and time t'. Namely,

$$
s_{jt}^\sigma \longrightarrow s_{jt}'^\sigma = \begin{cases} (-1)^{g_{i'j}^\sigma} s_{jt}^\sigma, & t = t', \\ s_{jt}^\sigma, & t \neq t', \end{cases}
$$
$$
s_{it} \longrightarrow s_{it}' = \begin{cases} -s_{it}, & i = i', t = t', t' + 1, \\ s_{it}, & \text{otherwise.} \end{cases} \qquad (28)
$$

Therefore, local gauge generators give rise to a (random) gauge model.

2. Bit-flip channel

Finally, consider the simpler case of a bit-flip channel $\mathcal{E}_{\text{bf}}(p)$ in a CSS-like code. As noted earlier, the case of a phase-flip channel is analogous and if both channels happen consecutively they can be treated independently.

The construction is an extension of the one in Sec. V B. The τ_j's and s_i are, respectively, replaced by the signs τ_{jt} and the Ising spins s_{it}. Given a choice of generators $\mathcal{S}_Z = \langle S_k \rangle$ to be measured at each time step, there is a corresponding collection of signs κ_{kj}. The generators take the form $S_k = \pm \bigotimes_j Z_j^{h_{kj}}$ for some $h_{kj} = 0, 1$. There is also an Ising spin $\hat{s}_{jt}$ for each physical qubit j and time step t. The Hamiltonians read

$$
H'_{\tau,\kappa}(s) := -J \sum_j \sum_t \tau_{jt} \hat{s}_{j(t-1)} \hat{s}_{jt} \prod_i s_{it}^{g_{ij}}
$$
$$
- K \sum_k \sum_t \kappa_{kt} \prod_j \hat{s}_{jt}^{h_{kj}}. \qquad (29)
$$

Instead of Eq. (27), the right conditions are now

$$
e^{-2\beta J} = \frac{p}{1-p}, \qquad e^{-2\beta K} = \frac{q}{1-q}. \qquad (30)
$$

The analog of Eq. (28) is

$$
\hat{s}_{jt} \longrightarrow \hat{s}_{jt}' = \begin{cases} (-1)^{g_{i'j}} s_{jt}, & t = t', \\ \hat{s}_{jt}, & t \neq t', \end{cases}
$$
$$
s_{it} \longrightarrow s_{it}' = \begin{cases} -s_{it}, & i = i', t = t', t' + 1, \\ s_{it}, & \text{otherwise.} \end{cases} \qquad (31)
$$

VI. CONCLUSION

Topological codes are intrinsically local and gauge or subsystem codes can have interesting locality properties. In

this article we introduce a family of topological subsystem codes, thus putting together the two concepts. The gauge group of the code is generated by two-local operators, which compares well with surface or color codes that have at least four-local and six-local generators, respectively. In particular, the measurement of these two-local operators is enough to recover the error syndrome.

We argue that these codes do not allow the introduction of boundaries with nice properties, which motivates further research. There are probably interesting topological codes still to be discovered. For example, one can look for subsystem codes with nice boundaries or with interesting transversality properties as those found in color codes.

We also explore a general connection between error correction in subsystem codes and statistical physics. The connection is especially meaningful in the case of topological codes where the error threshold maps to a phase transition in the corresponding statistical model. There is a lot of work to do in this direction, for example, in the computation, probably numerically, of the error threshold of the topological subsystem codes presented here.

ACKNOWLEDGMENTS

APPENDIX: STRUCTURE OF $N(\mathcal{G})$

This Appendix is a complement to Sec. IV B and uses the same notation.

A color code C_Λ^c [14] can be obtained from a lattice Λ with the properties enumerated in Sec. IV A. The construction is the following. First, there is one qubit per triangle so that the relevant Pauli group is $\mathcal{P}_{|F|}$. Given a collection of triangles $T = \{\tau_i\}$, set $X_T := \bigotimes_i X_{\tau_i}$, $Z_T := \bigotimes_i Z_{\tau_i}$. If each vertex $v \in V$ is identified with the set of triangles meeting at v, the stabilizer for the color code is $\mathcal{S}_\Lambda^c := \mathcal{S}_X \mathcal{S}_Z$ with $\mathcal{S}_X := \langle X_v \rangle_v$, $\mathcal{S}_Z := \langle Z_v \rangle_v$. Let $\{T_i\}$ be the collection of those sets of triangles that have an even number of triangles meeting at each vertex. Then $N(\mathcal{S}_\Lambda^c) = \langle i\mathbf{1} \rangle N_X N_Z$ with $N_X := \langle X_{T_i} \rangle_i$, $N_Z := \langle Z_{T_i} \rangle_i$.

Next, consider the morphism $f : N(\mathcal{G}_\Lambda) \longrightarrow N_X$ such that $f(O) = X_{T_\gamma}$ for any subgraph γ of $\bar{\Lambda}$ and $O \in N(\mathcal{G}_\Lambda)$ such that $O \propto O_\gamma$. The kernel of f is formed by those operators that only involve Z's, not X's or Y's. That is, $\ker f = \langle i\mathbf{1} \rangle \langle O_v^{c_v} \rangle_v \subset \mathcal{S}'$, where c_v is the color of v in Λ. Since $f(O_v^c) = X_v$ for $c \neq c_v$, $f[\mathcal{S}_\Lambda'] = \mathcal{S}_X$ and thus $\mathcal{S}_\Lambda' / \ker f \simeq \mathcal{S}_X$. This implies that there are no other constraints for the generators of $\mathcal{S}_\Lambda'$ apart from the ones in Eq. (5) because exactly two of the generators $\{X_v\}$ of $\mathcal{S}_X$ are unnecessary [14]. Finally, it is easy to check that for any string operator $X_T \in N_X$, as described in Ref. [14], there exists a string-like graph γ such that $f(O_\gamma) = X_T$, so that f is onto and $N(\mathcal{G}_\Lambda)/\mathcal{S}_\Lambda' \simeq N_X/\mathcal{S}_X$. Then the properties of the string operators in C_Λ are consequences of those for string operators in C_Λ^c. This is, in particular, true regarding the generating set and the composition rules, but not for commutation rules, which have to be worked out separately.

[1] P. W. Shor, Phys. Rev. A **52**, R2493 (1995).

[2] A. M. Steane, Phys. Rev. Lett. **77**, 793 (1996).

[3] E. Knill and R. Laflamme, Phys. Rev. A **55**, 900 (1997).

[4] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wootters, Phys. Rev. A **54**, 3824 (1996).

[5] P. W. Shor, in *Proceedings of the 37th Symposium on Foundations of Computing* (IEEE Computer Society Press, Los Alamitos, CA, 1996).

[6] E. Knill, R. Laflamme, and W. Zurek, Science **279**, 342 (1998).

[7] D. Aharonov and M. Ben-Or, in *Proceedings of the 29th annual ACM symposium on Theory of computing* (ACM Press, New York, 1997).

[8] D. Gottesman, Phys. Rev. A **57**, 127 (1998).

[9] J. Preskill, in *Introduction to quantum computation and information* (World Scientific, NJ, 1998).

[10] D. Gottesman, Phys. Rev. A **54**, 1862 (1996); e-print arXiv:0904.2557.

[11] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, Phys. Rev. Lett. **78**, 405 (1997).

[12] A. Yu. Kitaev, Ann. Phys. (NY) **303**, 2 (2003).

[13] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, J. Math. Phys. (NY) **43**, 4452 (2002).

[14] H. Bombin and M. A. Martin-Delgado, Phys. Rev. Lett. **97**, 180501 (2006).

[15] H. Bombin and M. A. Martin-Delgado, Phys. Rev. Lett. **98**, 160502 (2007).

[16] D. Bacon, Phys. Rev. A **73**, 012340 (2006).

[17] D. Poulin, Phys. Rev. Lett. **95**, 230504 (2005).

[18] R. Alicki, M. Fannes, and M. Horodecki, J. Phys. A: Math. Theor. **42**, 065303 (2009).

[19] R. Alicki, M. Horodecki, P. Horodecki, and R. Horodecki, e-print arXiv:0811.0033.

[20] H. Bombin and M. A. Martin-Delgado, J. Phys. A: Math. Theor. **42**, 095302 (2009).

[21] C. Wang, J. Harrington, and J. Preskill, Ann. Phys. (NY) **303**, 31 (2003).

[22] H. G. Katzgraber, H. Bombin, and M. A. Martin-Delgado, Phys. Rev. Lett. **103**, 090501 (2009).

[23] D. W. Kribs, R. Laflamme, D. Poulin, and M. Lesosky, Quantum Inf. Comput. **6**, 383 (2006).

[24] A. R. Calderbank and P. W. Shor, Phys. Rev. A **54**, 1098 (1996).

[25] A. Steane, Proc. R. Soc. London, Ser. A **452**, 2551 (1996).

[26] H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing* (Oxford University Press, New York, 2001).

[27] S. Bravyi and B. Terhal, New J. Phys. **11**, 043029 (2009).

[28] H. Bombin and M. A. Martin-Delgado, J. Math. Phys. (N.Y.) **48**, 052105 (2007).

[29] R. Raussendorf, J. Harrington, and K. Goyal, New J. Phys. **9**, 199 (2007).

[30] H. Bombin and M. A. Martin-Delgado, Phys. Rev. B **78**, 165128 (2008).

[31] H. Bombin, M. Kargarian, and M. A. Martin-Delgado, Phys. Rev. B **80**, 075111 (2009).

[32] H. Bombin and M. A. Martin-Delgado, Phys. Rev. B **75**, 075103 (2007).

[33] M. Ohzeki, Phys. Rev. E (to be published).

[34] A. Landahl, J. T. Anderson, and P. Rice (private communication).

[35] Let us show that if p_0' in Ref. [23] is 1 in a certain limit, then $[\Delta_i(K_p, \tau)]_{K_p} \to \infty$ for $i \neq 1$. Given $r > 0$, let T be the set of those τ's with $\Delta_i(K_p, \tau) < r$ for some $i \neq 1$ and set $p_r := \sum_T p(\tau)$. Then $p_0' \leqslant (1 - p_r) + p_r(1 + e^{-r})^{-1} = 1 - p_r(1 + e^r)^{-1}$. Thus for any r we have $p_r \to 0$.