Generating Fault-Tolerant Cluster States from Crystal Structures

Michael Newman[1], Leonardo Andreta de Castro[1,2], and Kenneth R. Brown[1]

[1]Departments of Electrical and Computer Engineering, Chemistry, and Physics, Duke University, Durham, NC, 27708, USA

[2]Present address: Q-CTRL Pty Ltd, Sydney, NSW, Australia

July 13, 2020

Measurement-based quantum computing (MBQC) is a promising alternative to traditional circuit-based quantum computing predicated on the construction and measurement of cluster states. Recent work has demonstrated that MBQC provides a more general framework for fault-tolerance that extends beyond foliated quantum error-correcting codes. We systematically expand on that paradigm, and use combinatorial tiling theory to study and construct new examples of fault-tolerant cluster states derived from crystal structures. Included among these is a robust self-dual cluster state requiring only degree-3 connectivity. We benchmark several of these cluster states in the presence of circuit-level noise, and find a variety of promising candidates whose performance depends on the specifics of the noise model. By eschewing the distinction between data and ancilla, this malleable framework lays a foundation for the development of creative and competitive fault-tolerance schemes beyond conventional error-correcting codes.

1   Introduction

Fault-tolerant quantum computation is possible, provided that error processes are sufficiently weak and uncorrelated [1–5]. However, quantum systems are inherently delicate, and so it is important maximize the robustness of a fault-tolerance strategy relative to the specific noise patterns a quantum processor experiences [6–16].

Typically, fault-tolerance schemes are founded on constructing and maintaining a quantum error-correcting code [17–22] on which abstracted logical circuits can be implemented with high fidelity. However, there is an alternative to circuit-based quantum computation (CBQC) that is built on the adaptive measurement of highly-entangled resource states [23, 24]. This alternative has been appropriately named measurement-based quantum computation (MBQC), and has blossomed within the framework of fault-tolerance [8, 25–27].

Most proposals for fault-tolerant MBQC are based on a topological approach, whereby quantum circuits are woven into resource states via adaptive measurement choices [25, 26]. The ambient space on which this topological pattern occurs is commonly referred to as the 3D cluster state vacuum, and represents the error-resilient stitching that protects the computation. The essential property of this state is that it can both propagate highly non-local correlations, which carry the logical information being manipulated, while supporting local correlations, which act as consistency checks to catch errors. This permits the sharing of long-range entanglement in the presence of noise, which is essential for any fault-tolerance scheme [27].

Early on, a fundamental connection was observed between 3D cluster states and surface codes [8], which was later extended to CSS codes [28, 29] and eventually all stabilizer codes [30]. This

Michael Newman: mgnewman@google.com

connection has been termed foliation, which according to its name, maps a stabilizer code to a cluster state representing the evolution of that code in time. In particular, each data qubit of the code is replaced by a time-like 1D cluster state, so that the foliated code maintains the error-correction properties of the base code in a naturally phenomenological error model.

Recently, pioneering work by Nickerson and Bombín studied cluster states that cannot be realized as the foliation of any quantum code [31]. The central insight is that fault-tolerant cluster states need not behave as time-like error-correcting codes to maintain their essential property: the coexistence of long-range logical correlations with local consistency checks. At the circuit level, this manifests as constantly measuring and reinitializing all of the qubits, so that each plays a role in both propagating logical information and catching errors. This stands in contrast to a quantum error-correcting code, in which the logical information is held statically in data qubits that remain unmeasured until the end of the computation, while only the ancilla qubits search for errors. Relaxing this division of labor opens up a new design space to explore, and [31] employed a splitting technique reminiscent of approaches in network theory [32, 33] to construct new cluster states that were resilient to both dephasing and erasure noise [34].

In this work, we use tools from combinatorial tiling theory [35] to study and construct a broad range of robust fault-tolerant cluster states. The central idea is to use a discrete analogue of orbifold covers to turn the geometric problem described in [31] into an algebraic problem on groups [36]. Although we focus on cluster states that are built from periodic crystal structures, we also discuss how these techniques can apply more generally. To illustrate the power of this algebraic approach, we construct a self-dual fault-tolerant cluster state that is optimally local, requiring only a 3-regular underlying graph state. See Figure 1 for a high-level description of the main idea.

(a) Choose a heuristic on dressed graphs. To each graph in the family, associate a group G.

(b) The graph can be used to define a desirable tiling of space. Place qubits on faces and edges.

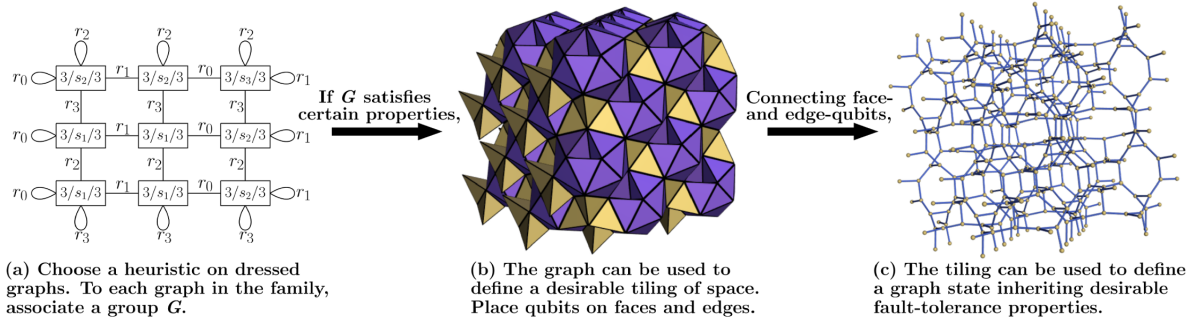(c) The tiling can be used to define a graph state inheriting desirable fault-tolerance properties.

Figure 1: Finding a degree-3 self-dual fault-tolerant cluster state, which necessarily cannot be realized as the foliation of any code. This is optimally local, as any degree-2 chain cannot fill space, and so cannot correct any type of error. More generally, a cluster state inherits properties affecting fault-tolerance from the chosen graph heuristic.

In addition, we benchmark these cluster states in the presence of circuit-level noise with varying relative parameters. We do so using a minimum-weight perfect matching decoder, which is less efficient than the union-find decoder [37, 38] utilized in [31], but which exhibits higher performance and relates to the zero-temperature phase transition of associated statistical mechanical models with quenched disorder [9, 39]. We find a variety of competitive cluster states whose robustness depends on the specifics of the noise parameters.

The paper is organized as follows. In Section 2, we discuss the general formulation of fault-tolerant cluster states in terms of chain complexes, and introduce their relevant circuit-level noise models. In Section 3, we review the foundations of combinatorial tiling theory, and use it to identify interesting fault-tolerant cluster states based on crystal structures. In Section 4, we provide numerics on examples drawn from an extensive classification of self-dual tilings as well as our numerical search to both benchmark promising candidates, and to probe the effects of circuit-level noise. Finally, we discuss avenues for future development in Section 5.

2 Fault-Tolerant Cluster States

2.1 CSS codes as chain complexes

To understand fault-tolerant cluster states, it is instructive to first review quantum CSS codes. In particular, there is a well-established interpretation of CSS codes as chain complexes [40–42]. Throughout, we implicitly work with coefficients in $\mathbb{Z}_2$, reflecting the use of qubits.

Definition 1. A *chain complex* $\mathcal{C}$ is a sequence of vector spaces $\{C_i\}$ and linear maps between them $\{\partial_i\}$ called boundaries:

$$0 \to C_\ell \xrightarrow{\partial_\ell} C_{\ell-1} \to \ldots \to C_1 \xrightarrow{\partial_1} C_0 \to 0$$

satisfying $\partial_{i-1} \circ \partial_i = 0$. Every such complex has an associated *cochain complex* $\mathcal{C}^*$ comprised of vector spaces $\{C^i := \text{Hom}(C_i, \mathbb{Z}_2)\}$ and linear maps between them $\{\partial^i(\cdot) := (\cdot) \circ \partial_i\}$ called coboundaries:

$$0 \leftarrow C^\ell \xleftarrow{\partial^\ell} C^{\ell-1} \leftarrow \ldots \leftarrow C^1 \xleftarrow{\partial^1} C^0 \leftarrow 0$$

satisfying $\partial^{i+1} \circ \partial^i = 0$. We sometimes refer to $\mathcal{C}^*$ as the *dual* of $\mathcal{C}$, and we call ℓ the *length* of the chain complex.

One important characteristic of a chain complex relevant to the associated quantum code is its homology. This measures the failure of each sequence of boundary maps to be exact.

Definition 2. The *homology groups* of the chain complex $\mathcal{C}$ are the quotient spaces $H_k(\mathcal{C}) := \ker(\partial_k)/\text{im}(\partial_{k+1})$. The *cohomology groups* of the chain complex $\mathcal{C}$ are the quotient spaces $H^k(\mathcal{C}) := \ker(\partial^{k+1})/\text{im}(\partial^k)$. Representatives of elements in these quotient spaces are referred to as *cycles* and *cocycles*, respectively.

It turns out that chain complexes and their homology provide a natural language for describing and building CSS codes.

Definition 3. An $[[n, k]]$ *CSS code* can be identified with a length-2 chain complex $\mathcal{C}$:

$$C_2 \xrightarrow{\partial_2} C_1 \xrightarrow{\partial_1} C_0$$

where $n = \dim(C_1)$ and $k = \dim(H_1(\mathcal{C}))$.

In particular, we have the following correspondence:

$$
\begin{array}{ccccc}
C_2 & \xrightarrow{\partial_2} & C_1 & \xrightarrow{\partial_1} & C_0, \\
\updownarrow & & \updownarrow & & \updownarrow \\
\{Z\text{-stabilizers}\} & & \{Z\text{-errors}\} & & \{X\text{-syndromes}\}
\end{array}
$$

$$
\begin{array}{ccccc}
C^2 & \xleftarrow{\partial^2} & C^1 & \xleftarrow{\partial^1} & C^0. \\
\updownarrow & & \updownarrow & & \updownarrow \\
\{Z\text{-syndromes}\} & & \{X\text{-errors}\} & & \{X\text{-stabilizers}\}
\end{array}
$$

The boundary condition $\partial_1 \circ \partial_2 = 0$ ensures the commutativity of the X- and Z-type stabilizers. By construction, the class of Z-type (X-type) logical operator representatives can be identified with nontrivial cycles (cocycles) in $H_1(\mathcal{C})$ ($H^1(\mathcal{C})$).

Logical qubits can be identified with pairs of X- and Z-type logical operators $\{\overline{X}_i, \overline{Z}_i\}$, and so it should be the case that $\dim(H_1(\mathcal{C})) = \dim(H^1(\mathcal{C}))$. Furthermore, these logical operators should satisfy canonical anti-commutativity relations $\overline{X}_i \overline{Z}_j = (-1)^{\delta_{ij}} \overline{Z}_j \overline{X}_i$. This correspondence manifests as a structure theorem on chain complexes [43].

Theorem 4 (Universal Coefficient Theorem). There is a well-defined isomorphism

$$\phi: \quad H^i(\mathcal{C}) \quad \longrightarrow \quad \mathrm{Hom}(H_i(\mathcal{C}), \mathbb{Z}_2)$$

given by $\phi([f])([x]) = f(x)$.

Thus, for any nontrivial cycle $x \in C_1$ representing a logical Z-type operator, there is a nontrivial cocycle $f \in C^1$ representing a logical X-type operator that intersects x with odd parity when identified with its dual in C_1. Furthermore, this element will intersect any equivalence class of cycles $[y] \neq [x]$ with even parity. Consequently, the factorization of the logical subspace into logical qubit subsystems is captured by the structure of the chain complex.

This perspective has been extremely profitable in several ways, including the construction of codes with improved parameters [41, 42, 44–47], the study of single-shot error-correction [48, 49], and the design of fault-tolerant gates [50, 51]. We detail two foundational examples of this correspondence in Figure 2.
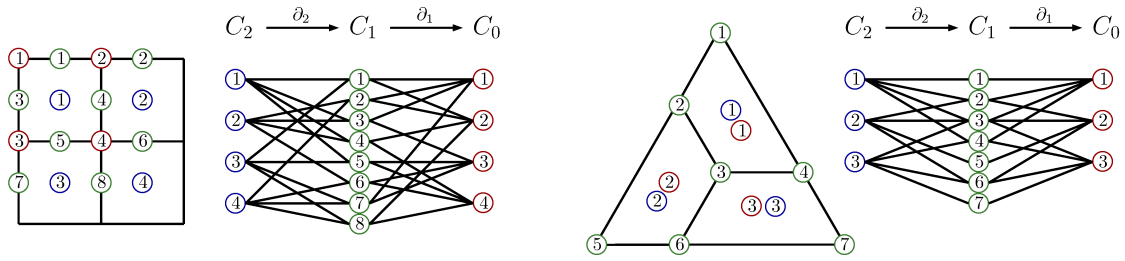

Figure 2: A toric code (left) and color code (right) regarded as chain complexes. For the toric code, the boundary maps between chain complexes are given by the geometric boundary maps in the periodic lattice. For the color code, the boundary maps between chain complexes are given by a pair of abstract Tanner graphs, which are not defined by geometric boundary maps.

In the language of chain complexes, error-correction in the code capacity model can be rephrased as the following problem:

(1)
$$\text{given } \left(\partial_1(\vec{z}), \partial^2(\vec{x})\right) \in C_0 \oplus C^2, \text{ determine } (\vec{c}_z, \vec{c}_x) \in C_1 \oplus C^1$$
$$\text{so that } ([\vec{z} + \vec{c}_z], [\vec{x} + \vec{c}_x]) = (0, 0) \in H_1(\mathcal{C}) \oplus H^1(\mathcal{C}).$$

Colloquially, this reads: given X- and Z-type syndromes, determine corrections that, taken together with the errors, apply the logical identity. For many error models, this problem can be treated independently in each coordinate without significant loss in performance [7, 52]. In the presence of depolarizing noise, it is then optimal to choose code families for which these two problems are symmetric, as the threshold will be determined by the worse of the two. These include families like the toric and color codes, while asymmetric families like Shor's code [18] will be bottlenecked by one type of error [53].

The chain complex formulation of error-correction can be extended to phenomenological noise as well. In particular, recent work has studied redundant syndromes checks in the context of length-4 chain complexes in order to characterize single-shot error-correction [48, 49]. It is then apt that fault-tolerant cluster states, which eschew the distinction between data and ancilla, meet in the middle between length-2 and length-4 chain complexes.

2.2 Fault-tolerant cluster states as chain complexes

Having described CSS codes in the chain complex framework, we turn to describing fault-tolerant cluster states. Despite their similarities, fault-tolerant cluster states afford an extra degree of freedom

that is useful in the construction of robust quantum processes [31]. As with most measurement-based schemes, our fault-tolerant cluster states will be built from a particular class of stabilizer states associated to graphs.

Definition 5. For a graph $G = (V, E)$, the associated *graph state* $|G\rangle$ is the unique state stabilized by the set of operators

$$\{X_v \prod_{(v,w) \in E} Z_w : v \in V\}.$$

Informally, *cluster states* refer to particular families of graph states that are used as a resource for MBQC.

Defining a complete fault-tolerant MBQC scheme involves several considerations, including details describing the implementation of a universal gate set. For existing proposals, this information requires the specification of additional structures, such as input/output surfaces and their relation to defects [25, 30].

In this work, we focus on the error-resilience of the vacuum. Consequently, we consider a distilled problem: the preservation of long-range correlations in the presence of errors. As was noted in [31], the fault-tolerance schemes we discuss can be readily adapted to universal fault-tolerant MBQC using the topological techniques in [25]. However, this simplified perspective allows for a concise description of error-resilient vacuums.

Definition 6. A *fault-tolerant cluster state* can be identified with a length-3 chain complex $\mathcal{C}$:

$$C_3 \xrightarrow{\partial_3} C_2 \xrightarrow{\partial_2} C_1 \xrightarrow{\partial_1} C_0.$$

The underlying graph state is given by the bipartite graph (C_2, C_1) with biadjacency matrix ∂_2. We refer to qubits indexed by C_2 as *dual*, and qubits indexed by C_1 as *primal*.

We emphasize that the underlying graph state is entirely determined by the map ∂_2, which encodes a single bipartite graph; the maps ∂_3 and ∂_1 serve to distinguish the correlations used for error-correction.

There is an intrinsic asymmetry in cluster states, indicated by the presence of X-type symmetries and absence of Z-type symmetries, introduced by the asymmetric definition of a graph state. This asymmetry also manifests in the construction and processing of a fault-tolerant cluster state. Following the prescription in [25], for a cluster state on n qubits with underlying graph state $G = (V, E)$, this proceeds in three steps.

(i) Prepare $|+\rangle^{\otimes n}$.

(ii) For each $(v, w) \in E$, apply $\text{CZ}_{v,w}$ in some specified order.

(iii) Measure each qubit in the X-basis.

In particular, we have the following correspondence:

$$
\begin{array}{ccccccccc}
C_3 & \xrightarrow{\partial_3} & C_2 & \xrightarrow{\partial_2} & C_1 & \xrightarrow{\partial_1} & C_0, \\
\updownarrow & & \updownarrow & & \updownarrow & & \updownarrow \\
\{\text{dual } X\text{-stabilizers}\} & & \{\text{dual } X\text{-symmetries}\} & & \{\text{primal } Z\text{-errors}\} & & \{\text{primal syndromes}\}
\end{array}
$$

$$
\begin{array}{ccccccccc}
C^3 & \xleftarrow{\partial^3} & C^2 & \xleftarrow{\partial^2} & C^1 & \xleftarrow{\partial^1} & C^0. \\
\updownarrow & & \updownarrow & & \updownarrow & & \updownarrow \\
\{\text{dual syndromes}\} & & \{\text{dual } Z\text{-errors}\} & & \{\text{primal } X\text{-symmetries}\} & & \{\text{primal } X\text{-stabilizers}\}
\end{array}
$$

For any $\vec{v} \in C_2$ indexing dual qubits, there is a corresponding symmetry of the form

$$\prod_{i:\vec{v}_i=1} X_i \prod_{j:\partial_2(\vec{v})_j=1} Z_j,$$

where the X_i are supported on dual qubits and the Z_j are supported on primal qubits. The reverse relation holds for $\vec{v} \in C^1$ in the cochain complex. Thus, when we identify C_2 (C^1) with dual (primal) X-symmetries, we mean symmetries whose restriction to dual (primal) qubits is of X-type.

Just as the structure of a length-2 chain complex can be interpreted naturally in terms of the properties of a CSS code, so too can the structure of a length-3 chain complex be interpreted naturally in terms of the properties of a fault-tolerant cluster state.[1] Note that this chain complex is also related to the ungauging complex introduced for quantum codes, and more specifically for 3D surface codes when the cluster state is built from a tiling [54]. In the table below, the left-hand side represents either an object associated to or property of the chain complex. The right-hand side represents how that object or property plays a role in the protected computation within the MBQC model.

Chain Complex	Fault-tolerant Cluster State
$\partial_2 \circ \partial_3 = 0$	Dual X-stabilizers are accessible via X-measurements.
$H_2(\mathcal{C})$	Indexes equivalence classes of dual logical symmetries.
$\partial_1 \circ \partial_2 = 0$	Primal Z-errors equivalent to dual X-errors are undetectable.
$H_1(\mathcal{C})$	Indexes equivalence classes of primal logical Z-errors.
$\partial^2 \circ \partial^1 = 0$	Primal X-stabilizers are accessible via X-measurements.
$H^1(\mathcal{C})$	Indexes equivalence classes of primal logical symmetries.
$\partial^3 \circ \partial^2 = 0$	Dual Z-errors equivalent to primal X-errors are undetectable.
$H^2(\mathcal{C})$	Indexes equivalence classes of dual logical Z-errors.

Equivalence classes of dual (primal) logical symmetries are defined up to dual (primal) stabilizer equivalence, while equivalence classes of undetectable dual (primal) Z-errors are defined up to primal (dual) X-error equivalence. Such errors are trivial, as they commute with our X-measurements. Theorem 4 ensures that there are canonical isomorphisms

$$H^2(\mathcal{C}) \leftrightarrow H_2(\mathcal{C})$$
$$H^1(\mathcal{C}) \leftrightarrow H_1(\mathcal{C})$$

so that for every logical symmetry, there is a corresponding undetectable error that negates it.

One subtlety is that, without specifying input/output surfaces, there is no explicit correspondence between primal and dual symmetries emulating a logical qubit [27, 30]. This correspondence is explicit for foliated codes, as measuring the bulk of the lattice generates an encoded long-range Bell pair between these surfaces [28]. We focus on fault-tolerant cluster states realized by periodic cellulations of 3-space, for which the notions of input/output surfaces, defects, and topologically protected gates can be introduced directly [25, 31]. However, fault-tolerant cluster state families need not be geometrically defined, as the chain complex structure need not stem from cellulations of a manifold. Foliations of more general quantum low-density parity check code families are examples of such states, and fault-tolerant cluster states can be more general still.

Having established the parallels between length-3 chain complexes and fault-tolerant cluster states, error-correction in the fault-tolerant MBQC model can be rephrased as the following problem:

(2)
$$\text{given } \left(\partial_1(\vec{z}_p), \partial^3(\vec{z}_d) \right) \in C_0 \oplus C^3, \text{ determine } (\vec{c}_p, \vec{c}_d) \in C_1 \oplus C^2$$
$$\text{so that } ([\vec{z}_p + \vec{c}_p], [\vec{z}_d + \vec{c}_d]) = (0,0) \in H_1(\mathcal{C}) \oplus H^2(\mathcal{C}).$$

Once again, this can be read colloquially as: given primal and dual syndromes, determine primal and dual corrections that, taken together with the errors, preserve the primal and dual logical symmetries. Mirroring CSS codes, this problem can be treated independently in each coordinate, and so it is often

---

[1]Note that primal logical symmetries and dual logical symmetries correspond, respectively, to dual correlation surfaces and primal correlation surfaces as described in [25]. For a geometrically defined cluster state, a primal (dual) logical symmetry can be visualized as a closed surface in the dual (primal) lattice.

optimal to choose a fault-tolerant cluster state for which these two problems are symmetric. In particular, both primal and dual symmetries are required to perform universal fault-tolerant computation [25].

Although the problems of error-correction in CSS codes and error-correction in fault-tolerant cluster states appear similar, comparing problems (1) and (2), there are key differences between them (see Figure 3). There is a mapping between these two problems that associates to any stabilizer code a fault-tolerant cluster state called the *foliation* of that code. The key insight made in [31] is that this mapping is too restrictive; there is an extra degree of freedom in problem (2) that can be exploited to construct more robust fault-tolerant protocols. To highlight this difference, we review foliation in the case of CSS codes, which has also been extended to all stabilizer codes [30].

(a) The 3D cluster state.

(b) The 3D surface code.
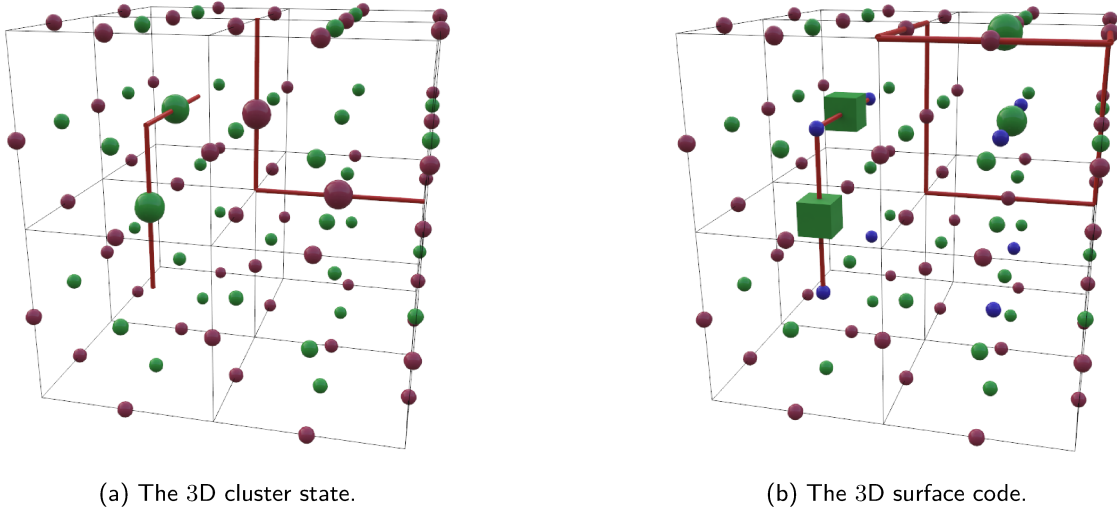
Figure 3: A comparison of error-correction in the superficially similar (a) 3D cluster state and (b) 3D surface code on a cubic lattice. Z-errors (X-errors) are denoted by enlarged circles (cubes), and their syndromes are represented as red lines. (a) In the 3D cluster state, only Z-errors are damaging, as X-errors commute with the X-measurements. Each qubit is measured, and syndromes are processed as correlations between outcomes. Z-errors on dual (primal) qubits are detected by correlations about cubes (vertices). (b) In the 3D surface code, there are code qubits (green), Z-ancilla (blue), and X-ancilla (red). X-errors (Z-errors) are caught by Z-ancilla (X-ancilla), producing point-like (string-like) syndromes. However, measurements themselves can fail, which is captured in (a) but not (b). In particular, vertices play a role in (a) but not (b), highlighting the difference between the underlying length-3 and length-2 chain complexes.

2.3 Foliated quantum codes

Foliated quantum codes are not codes per se, but rather fault-tolerant cluster states that are closely related to codes. In particular, foliation is a map that takes any CSS (or more generally stabilizer) code and constructs a corresponding fault-tolerant cluster state that emulates that code evolving in time for some number of time steps [28]. The most famous example is the 3D cluster state on a cubic lattice, which can be seen as a foliated 2D surface code.

Given a CSS code presented as a length-2 chain complex

$$C_2 \xrightarrow{\partial_2} C_1 \xrightarrow{\partial_1} C_0$$

and a number of time-steps T, its T-*time-step foliation* is the following fault-tolerant cluster state, presented as a length-3 chain complex:

$$C_2^{\oplus T} \xrightarrow{\delta_3} (C_2 \oplus C_1)^{\oplus T} \xrightarrow{\delta_2} (C_0 \oplus C_1)^{\oplus T} \xrightarrow{\delta_0} C_0^{\oplus T}.$$

Let id^V denote the identity on V, and for $f : V \to W$, let $f_{i,j} : V^{\oplus T} \to W^{\oplus T}$ denote the map that applies f from V_i to W_j and is zero everywhere else. Then the corresponding boundary maps are

$$\delta_3 = \mathrm{id}_{T,T}^{C_2} + \partial_{2_{T,T}} + \sum_{i=1}^{T-1} \mathrm{id}_{i,i}^{C_2} + \mathrm{id}_{i,i+1}^{C_2} + \partial_{2_{i,i}},$$

$$\delta_2 = \mathrm{id}_{T,T}^{C_1} + \partial_{2_{T,T}} + \partial_{1_{T,T}} + \sum_{i=1}^{T-1} \mathrm{id}_{i,i}^{C_1} + \mathrm{id}_{i,i+1}^{C_1} + \partial_{2_{i,i}} + \partial_{1_{i,i}},$$

$$\delta_1 = \mathrm{id}_{T,T}^{C_0} + \partial_{1_{T,T}} + \sum_{i=1}^{T-1} \mathrm{id}_{i,i}^{C_0} + \mathrm{id}_{i,i+1}^{C_0} + \partial_{1_{i,i}}.$$

Intuitively, the above prescription assembles 1D cluster states of length $2T$ for each data qubit. Then, Z-type and X-type ancilla qubits are introduced in alternating layers, and connected via CZ gates to the data qubits in their associated stabilizers. Commutativity of the stabilizers ensures that this chain complex is well-defined, and the corresponding fault-tolerant cluster state inherits the properties of the underlying CSS code [28, 30]. This picture recovers the 3D cluster state as a foliated surface code, depicted in Figure 4. We include the 2-time-step foliated $[[4, 2, 2]]$ code, along with its associated chain complexes, as another leading example in Figure 5.
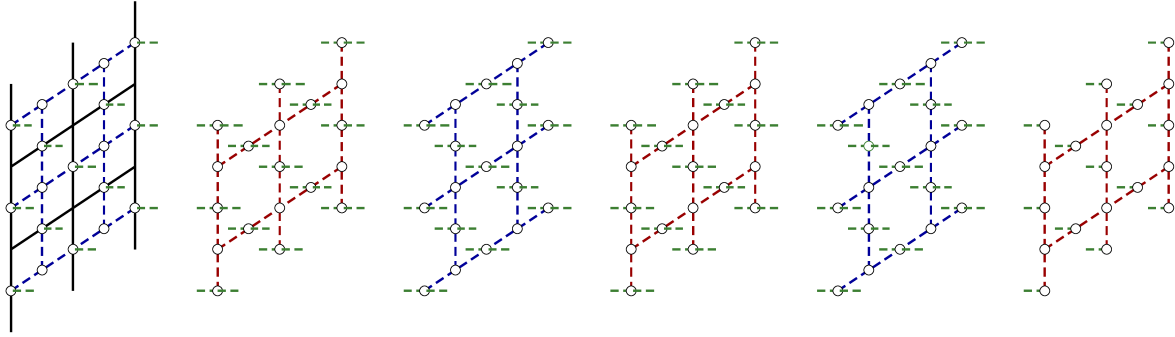

Figure 4: A 3D cluster state regarded as a foliated surface code. Dotted green lines (abbreviated for clarity) represent time-like 1D cluster states, dotted blue lines represent Z-type ancilla interactions, and dotted red lines represent X-type ancilla interactions. The time-like 1D cluster states can be regarded as Hadamard clocks, with Z- and X-type information available at alternating time steps, with time moving from left to right.

Foliated quantum codes are closely related to their corresponding CSS codes in a phenomenological model; indeed the space-time decoding processes for each can be nearly identical [7, 8]. Foliated codes also yield a prescription for building fault-tolerant cluster states: a priori, building length-3 chain complexes with desirable properties like high distance, high encoding rate, or locality can be a difficult task. Generally, foliation allows one to port existing constructions of quantum error-correcting codes to the MBQC framework, making it a powerful tool.

However, it is also restrictive. There are chain complexes corresponding to fault-tolerant cluster states that cannot be realized as the foliation of any quantum code [31]. The broadened framework of length-3 chain complexes permits a wide variety of robust cluster states that often outperform their foliated cousins. In some sense, this generalizes the notion of quantum error-correction in a phenomenological error model. Mirroring the construction of the toric code, one good prescription for building sparse chain complexes corresponding to robust cluster states is by cellulating manifolds. However, before building cluster states from cellulations, it will be helpful to first interpret the effects of different types of noise on such states. In what follows, we describe a circuit-level error model and identify which cluster states should be robust to which parameter regimes of noise.
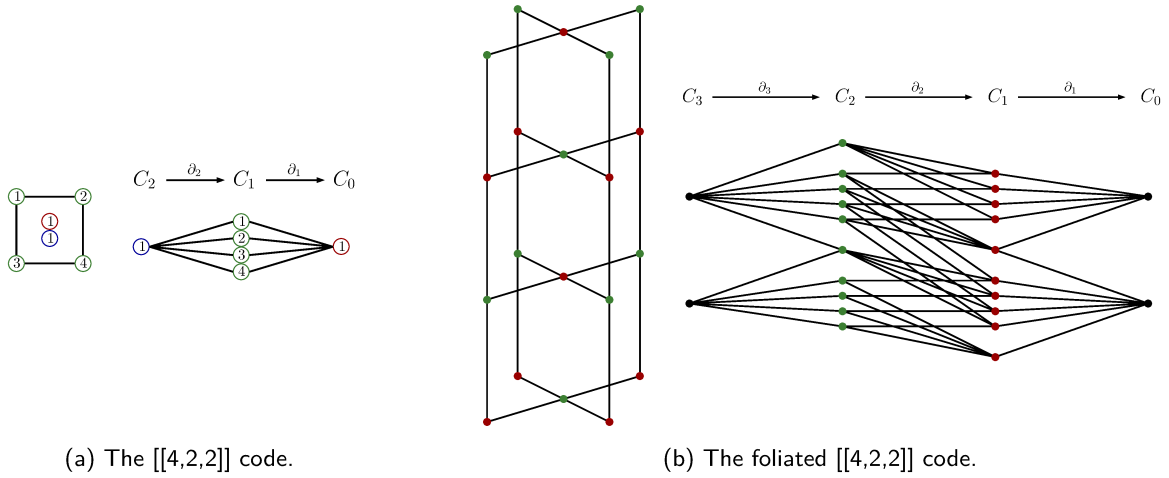
(a) The [[4,2,2]] code.

(b) The foliated [[4,2,2]] code.

Figure 5: (a) A standard representation of the $[[4, 2, 2]]$ code and its associated length-2 chain complex. (b) The underlying graph state of the 2-time-step foliated $[[4, 2, 2]]$ code and its associated length-3 chain complex. Dual (primal) qubits are colored green (red). For clarity, the four data qubits in a particular time-slice are grouped together. Each time-slice is further subdivided into two steps, which is encoded by the location of the group of four in either C_2 or C_1.

2.4   Circuit-based error model and its effects

We consider a simple circuit-level Pauli error model that captures the correlated errors that may occur due to gate failures. Decoding will be performed independently on primal and dual qubits, and so we describe errors that manifest as Z-type on primal qubits; errors on dual qubits can be treated analogously. We consider three types of errors.

(i)   Measurement errors occur with probability p_m. This can also account for preparation errors.

(ii) X-type errors occur on the dual qubit support of a CZ gate with probability p_X after the gate.

(iii) Z-type errors occur on the primal qubit support of a CZ gate with probability p_Z after the gate.

Although X-errors commute with our measurements, X-errors occurring on dual qubits during the construction of the cluster state will result in correlated Z-errors on primal qubits. Note that this is *not* the standard depolarizing model, in which each two-qubit gate would ordinarily be followed by a depolarizing channel on its support. For example, we do not include idling errors, and each two-qubit gate has two independent chances to fail, so that the probability of having no errors is $(1-p_X)(1-p_Z)$ rather than $(1-p)$. This atypical model is more damaging, and leads to lower thresholds than the standard error model. However, it allows us to sample errors directly as edges on the decoder graph, which simplifies numerical simulations over a wide range of complicated lattices significantly. In order to draw some comparison between this model and the standard model, we include benchmarks of the usual 3D cluster state as a point of reference [8].

For fault-tolerant cluster states constructed as cellulations, we build the underlying graph state face-by-face. In each face, the CZ gates are applied in consecutive order about the face so that the resulting correlated errors are string-like.[2] An augmented 1-skeleton of the cellulation will then serve as the decoder graph, see Figure 6. We elaborate on the unit cell of each lattice, along with the decoder graph, and the gate scheduling in Appendix A.

There are three broad design principles discussed in [31] relevant to the construction of fault-tolerant cluster states built as cellulations of manifolds:

---

[2]There is a subtlety here: the error-correction performance may depend on this ordering, and optimal orderings for primal and dual qubits may not be compatible.
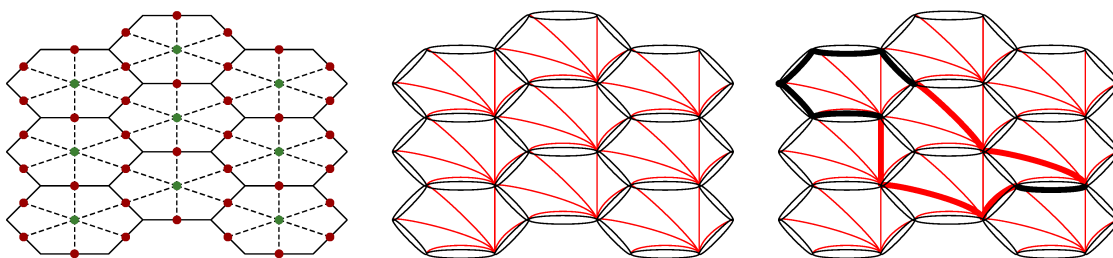

Figure 6: On the left, part of a cluster state, with dual (primal) qubits in green (red). The cell complex is defined by solid lines, while the underlying graph state is defined by dashed lines. The cluster state is constructed by applying CZ gates in each face beginning with the bottom edge and proceeding clockwise. In the center, each black edge represents a Z-error occurring on a primal qubit immediately following a CZ gate. Each red edge represents a correlated Z-error resulting from an X-error on a dual qubit immediately following a CZ gate. Measurement errors are omitted for clarity. On the right, a collection of highlighted gate errors resulting in a trivial cycle. Like the foliated repetition code, this 2D cluster has no checks on dual qubits. However, this partial protection extends to more general cluster phases defined on 2D lattices supporting one-form symmetries [55].

(i) Low average vertex degree: heuristically, lower degree vertices yield more information about the surrounding errors, resulting in more robust error-correction. In the limiting case, the repetition code in a code capacity model can be thought of as having a degree-2 decoder graph. This is particularly important when p_m dominates, as measurement errors are indifferent to the construction cost of the cluster state. Consequently, choosing the most robust error-correction is optimal.

(ii) Few edges in each face: a face with many edges may result in long-range correlated errors. This is particularly important when p_X dominates, as X-errors on dual qubits cause such errors.

(iii) Few faces incident to each edge: to first order in p_Z, the effective error rate each qubit experiences is given by the number of CZ gates it participates in. This is determined by the number of faces its corresponding edge belongs to, and so is particularly important when p_Z dominates.

As correlated errors on two-qubit gates tend to commute with the gates themselves [56–58], we generally expect $p_Z > p_X$ with p_m varying; however, we consider a variety of noise models. Additionally, for self-dual complexes, principles (ii) and (iii) become identical. A final heuristic worth following is increasing the symmetry of the underlying graph states, as homogeneous structures tend to perform better. Broadly, we expect that given different relative error strengths, different fault-tolerant cluster states appropriately balancing robust error-correction with locality become viable.

3 Combinatorial tiling theory

Building quantum codes from manifolds has been enormously profitable, resulting in some of the best known local codes [41, 42, 45, 46, 59–61]. However, we do not focus on the topological properties of a fault-tolerant cluster state, but rather the way in which its vacuum is stitched together. Correspondingly, the objects of interest are no longer manifolds but rather certain *orbifolds*,[3] and in particular, their underlying simplicial structures. These admissible triangulations are encoded into dressed graphs known as *extended Schläfli symbols*, and their study is broadly known as *combinatorial tiling theory* [35, 36, 62–71]. Put simply for our purposes, it is the algebraic study of ways to periodically tile space.

Following [31], we focus on building fault-tolerant cluster states from such tilings, forming a length-3 chain complex in the usual way. The underlying graph state is constructed by placing qubits on faces and edges, with graph state edges determined by ∂_2. The 2-skeleton of the tiling will determine

---

[3]A space locally modeled on Euclidean space modulo a linear finite group action, which may not be free.

the robustness of primal error-correction, with a decoder graph that is an augmented version of the 1-skeleton. The same holds for dual qubits in the dual tiling, and so we focus on self-dual complexes to realize symmetric thresholds.

3.1 Constructing tilings

We next review the foundations of combinatorial tiling theory, following the prescription of [36] towards the construction and recognition of tilings.

Definition 7. A *tiling* is a regular CW-complex[4] $\mathcal{C}$ realizing a simply connected d-dimensional manifold. Its *symmetry group* Γ is the group of all automorphisms that respect the induced CW-filtration.

Informally, a tiling is a partition of space, with each tile glued face-to-face in such a way that the entire space is filled. It determines a group of symmetries that respect the partition. These are reflections, rotations, and translations that map the tiles back to themselves in such a way that the new tiling is the same as the old. There is a notion of equivalence between tilings that depends on both their topological structure and their symmetry group. This will determine the level of granularity at which we will consider tilings, with two equivalent tilings sharing the same combinatorial structure.

Definition 8. Two tilings $\mathcal{C}_1, \mathcal{C}_2$ with respective symmetry group Γ_1, Γ_2 are *equivariantly equivalent* if there exists a homeomorphism $f : \mathcal{C}_1 \to \mathcal{C}_2$ respecting the filtration that induces an isomorphism $\Gamma_1 \to \Gamma_2$ given by $\gamma_1 \mapsto f \circ \gamma_1 \circ f^{-1}$.

To study tilings, it is useful to break them down into elemental combinatorial objects called flags. These will be defined in terms of objects of different dimensions. We refer to such objects C_i as cells, where i denotes the dimension of the cell. For example, C_0 is a point, C_1 an edge, and so on. The set of all cells of dimension $\leq i$ is called the i-skeleton, and when $i = d$, it represents the entire CW-complex.

Definition 9. A *flag* is an increasing sequence of cells $(C_0, C_1, \ldots, C_d)$ with $C_i \subseteq C_{i+1}$ and where each C_i belongs to the i-skeleton, but not the $(i-1)$-skeleton.

Geometrically, flags can be visualized as d-simplices obtained in the following way. For each i-cell in a complex, associate a vertex at its barycenter. Then, connect the vertices of the cells comprising a flag to form its corresponding i-simplex. For example, a square has 8 flags corresponding to triangles obtained from the barycentric subdivision of its faces and edges, while a cube has 48 flags corresponding to tetrahedra.

Let $\mathcal{F}$ denote the set of all flags. A central insight of [62] is to study two relevant group actions on $\mathcal{F}$. The first is induced by Γ, as each automorphism respects the filtration. The second is by the free Coxeter group.

Definition 10. The *free Coxeter group* generated by $d + 1$ reflections is the group

$$\mathcal{R} := \langle \{r_i\}_{i=0}^d : \{r_i^2 = 1\}_{i=0}^d \rangle.$$

For each flag $F = (C_i)_i$ and each index $j \in \{0, \ldots, d\}$, there is a unique flag $F' = (C_i')_i$ satisfying $C_i = C_i'$ for all $i \neq j$ and $C_j \neq C_j'$. This yields a well-defined action given by $r_j(F) := F'$. Geometrically, this action can be visualized as reflecting the d-simplex corresponding to F about the face opposite the vertex corresponding to C_j. This new d-simplex corresponds to F'.

The actions of Γ and $\mathcal{R}$ commute, and so the $\mathcal{R}$-action descends to an action on the set of flag orbits $\{\Gamma \cdot F : F \in \mathcal{F}\}$. Specifying the free Coxeter group action, in terms of its generators and relations, on each flag orbit carries all relevant information about a tiling. The objects representing this information are known as *extended Schläfli symbols* (or *Delaney symbols*, see [72]).

---

[4]Closure-finite, weak topology.

Definition 11. A d-dimensional *extended Schläfli symbol* is a set S equipped with a transitive $\mathcal{R}$-action along with, for each $s \in S$ and for all $0 \leq i, j \leq d$, positive natural numbers $m_{ij}(s)$ satisfying

(i) $m_{ii}(s) = 1$,

(ii) $(r_i r_j)^{m_{ij}(s)} \cdot s = s$,

(iii) $m_{ij}(s) = m_{ji}(s) = m_{ij}(s')$ for any $s' \in \langle r_i, r_j \rangle \cdot s$.

The symbol is further called *regular* if $m_{ij} = 2$ for all $|i - j| > 1$.

We will generally refer to these as symbols S, leaving implicit the extra information carried by the set. There is also a notion of maps between symbols called *coverings*, analogous to covering maps between orbifolds.

Definition 12. A *covering* of a d-dimensional symbol S by another d-dimensional symbol S' is an $\mathcal{R}$-equivariant map $f : S' \to S$ satisfying $m_{ij}(f(s')) = m_{ij}(s')$ for all $s' \in S'$. The covering is an *isomorphism* if f is bijective.

Informally, coverings are simply maps between symbols that preserve the structure of the symbol, namely the labeled adjacency relations and the vertex labels. We will often be concerned with self-dual cluster states, as our thresholds will be limited by error-correction on the worse of the two lattices. Fortunately, the corresponding notion of duality on symbols is quite simple.

Definition 13. The *dual* of a d-dimensional extended Schläfli symbol is the symbol obtained by interchanging $r_i \leftrightarrow r_{d-i}$ and $m_{ij} \leftrightarrow m_{d-i,d-j}$. A symbol is called *self-dual* if it is isomorphic to its dual.

Thus, for the purposes of building fault-tolerant cluster states, we will mostly be concerned with self-dual symbols. For a d-dimensional tiling $\mathcal{C}$ with symmetry group Γ, we can associate a d-dimensional symbol in the following way.

(i) Let $S := \{\Gamma \cdot F : F \in \mathcal{F}\}$ with corresponding transitive $\mathcal{R}$-action inherited from $\mathcal{F}$.

(ii) For each flag orbit $\Gamma \cdot F$, for any flag $F' \in \Gamma \cdot F$, and for any $0 \leq i, j \leq d$, let m_{ij} denote the order of the action about $r_i r_j$, namely $m_{ij} := \min\{k : (r_i r_j)^k F' = F'\}$. This is independent of the choice of representative F'.

Note that for a tiling, the ith cell of $r_i(F)$ depends only on the $(i - 1)$- and $(i + 1)$-cells of F. Consequently, any symbol associated to a tiling must be regular, as the corresponding r_i, r_j must commute.

To summarize, a symbol has a vertex for each unique flag, up to symmetries of the tiling. For example, the symbol for a regular tiling like that of the plane by squares has a single vertex. In contrast, the 4.8.8 tiling of the plane has three vertices corresponding to flags (triangles) within squares, flags that border two octagons, and flags that border both a square and an octagon; see Figure 7. Extended Schläfli symbols provide a sharp invariant for characterizing equivariant equivalence classes of tilings.

Theorem 14 ([62])**.** Two tilings are equivariantly equivalent if and only if their associated extended Schläfli symbols are isomorphic.

Symbols are often presented as dressed graphs [36]. The vertex set is given by S, with each vertex labeled by its corresponding $m_{i,j}$. For regular symbols associated to tilings, the order of the necessarily commuting reflections is typically omitted, so that each vertex is only labeled by $m_{i,i+1}$. The edge set is determined by the $\mathcal{R}$-action, with each edge labeled by the corresponding reflection generator. For tilings, these can be visualized by barycentrically subdividing the tiling into d-simplices representing flags, with the $\mathcal{R}$-action represented by reflections about facets; see Figure 7.

Just as we can create a symbol from a tiling, so too can we construct the underlying space $\mathbf{Top}(S)$ of an orbifold encoded by a symbol S. This is done by associating to each $s \in S$ a simplex with

---

(a) The 4.8.8 tiling.

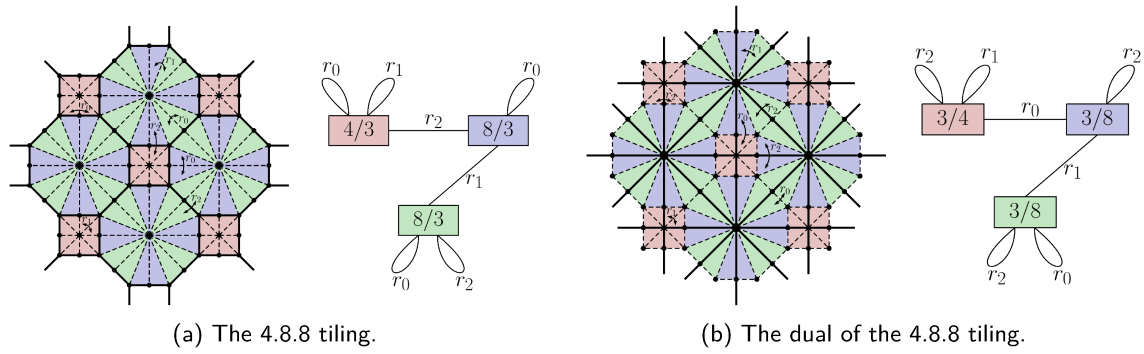(b) The dual of the 4.8.8 tiling.

Figure 7: The 4.8.8 tiling and its corresponding extended Schläfli symbol, represented as a dressed graph, along with their duals. The edges of the original lattice are solid, while edges introduced by the barycentric subdivision are dashed, with the added vertices shown. Up to symmetry, there are three unique flags colored red, blue, and green. Each triangle represents a flag, with its vertices corresponding to a face, edge, and vertex. The action of r_i corresponds to reflecting about the facet opposite the vertex representing an i-cell. Examples of each reflection generator appearing in the symbol are shown. Dissimilar to a regular tiling, not all of these reflections are themselves symmetries. For example, any edge corresponding to a reflection mapping between flags of different colors cannot be a symmetry of the tiling, as it maps between two symmetry inequivalent flags. Each flag orbit is labeled with m_{01}/m_{12}; in dimension 2, this corresponds to the number of edges in the face containing the flag, and the degree of the vertex incident to the flag, respectively. The nodes of the dressed graph are colored for clarity. As the 4.8.8 tiling is not self-dual, the two symbols are not isomorphic.

$(d + 1)$-facets, and then identifying the ith facets of two simplices if the corresponding elements of S are exchanged by r_i.[5] Coverings can also be realized as continuous maps that identify simplices.

To summarize, we can map from tilings to symbols, and from symbols to topological spaces, where the equivariant equivalence class of a tiling is determined by its associated symbol.[6] We would like to construct tilings from symbols directly in order to turn the former geometric problem into a (dressed) graph problem. To do so, we must determine which symbols can be used to realize tilings of Euclidean 3-space.

In order to construct tilings from symbols, we can use a covering space theory of symbols inherited from the covering space theory of orbifolds [36, 70]. This will further transform the problem on dressed graphs into an algebraic problem on their associated fundamental groups.

Definition 15. Let S be an extended Schläfli symbol with basepoint s. Up to isomorphism, there is a unique symbol $\overline{S}$ with basepoint $\overline{s}$ and a covering $u_S : \overline{S} \to S$ with $u_S(\overline{s}) = s$ such that for any other covering $f : S' \to S$ and any $s' \in f^{-1}(s)$, there is a unique $u_{S'} : \overline{S} \to S'$ through which u_S factors: $u_S = f \circ u_{S'}$ with $u_{S'}(\overline{s}) = s'$ [70]. We call $\overline{S}$ the *universal cover* of S.

Informally, a symbol is an encoding of a topological space with some symmetry 'packed away'. The larger the symbol, the more flags it has, the more triangles glued together it represents, and the less of its symmetry is packed away. To come back to the square tiling example, a 2-torus corresponds to a square, which can be constructed from a symbol with eight vertices. There is an implicit translational symmetry in the x- and y-directions. The square tiling is its universal cover, obtained from 'unfurling' these symmetries; this is constructed from an infinite symbol that has no symmetries packed away. Each of its triangles is explicitly defined by the symbol, rather than implicitly defined by a symmetry. These implicit symmetries of a symbol are described by the *fundamental group* of the symbol, and to obtain the fundamental group of a symbol, we must construct its universal cover.

---

[5]The information specifying the singular locus of the orbifold is lost in this mapping.

[6]The composition of these maps need not recover the tiling, but rather encodes an orbifold covered by the tiling.

An explicit presentation for the universal cover of S can be constructed according to the prescription in [36]. Namely, fix a basepoint $s \in S$ and identify $\overline{S}$ with cosets in $\mathcal{R}$ of the normal subgroup

$$U_s \coloneqq \langle r^{-1}(r_i r_j)^{m_{ij}(r \cdot s)} r : r \in \mathcal{R}, 0 \leq i, j \leq d \rangle,$$

where the $\mathcal{R}$-action is inherited from the group, and define $m_{ij}(r \cdot U) \coloneqq m_{ij}(r \cdot s)$. As $U_s \subseteq \mathrm{Stab}_{\mathcal{R}}(s)$, this defines a valid cover with covering map $r \cdot U \mapsto r \cdot s$, where the information about S is passed to U_s through the group action in the exponent of $r_i r_j$. In fact, opposite to the universal cover $\overline{S}$ of S, there also exists a *final* (or minimal) *cover* $\underline{S}$. Up to basepoints, this is defined as the unique symbol with covering $f_S : S \to \underline{S}$ such that, for any other covering $f : S \to S'$, there is a unique $f_{S'} : S' \to \underline{S}$ through which f_S factors: $f_S = f_{S'} \circ f$.

Given the universal cover of a symbol S, we can define the fundamental group of S as the group of deck transformations in the usual way.

Definition 16. Let S be an extended Schläfli symbol with basepoint s, and let $\overline{S}$ be its universal cover with covering map u_S. Then the *fundamental group* of S is given by the subgroup of automorphisms of $\overline{S}$ annihilated by u_S,

$$\pi_1(S, s) \coloneqq \{\phi \in \mathrm{Aut}(\overline{S}) : u_S \circ \phi = u_S\}.$$

Following the construction of the universal cover, the fundamental group can be constructed explicitly as $\pi_1(S, s) = \mathrm{Stab}_{\mathcal{R}}(s)/U_s$ with a finite presentation [73]. If this symbol is covered by a tiling, then the fundamental group corresponds to a subgroup of the symmetries of that tiling. For example, only reflections r_i that map between symmetry equivalent flags are themselves symmetries of the tiling. Correspondingly, these are exactly the reflections that belong to $\mathrm{Stab}_{\mathcal{R}}(s)$. More generally, we have the usual correspondence between subgroups $G \subseteq \pi_1(S, s)$ and covers of S, given by $G \leftrightarrow \overline{S}/G$ and pictured below.
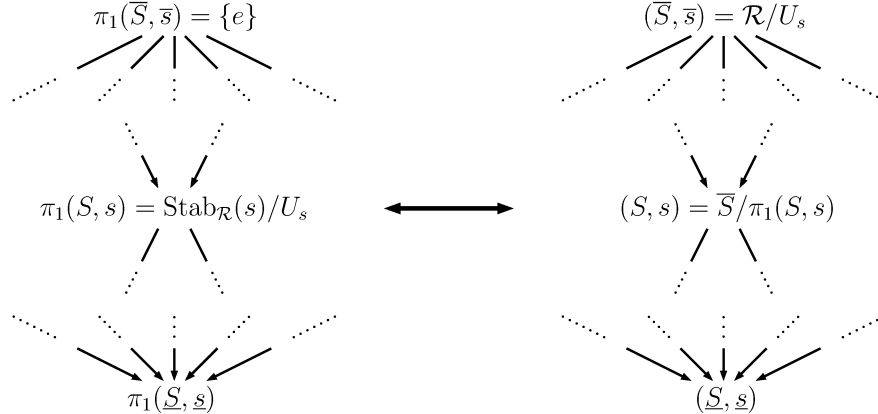
On the left, we have a sequence of inclusions of groups, and on the right, a corresponding sequence of coverings of symbols. From top to bottom, the sizes of the objects on the left increase, while the sizes of the objects on the right decrease. We can regard moving up the diagram as unfurling the symmetries of a symbol; the final cover has a maximal group of symmetries $\pi_1(\underline{S}, \underline{s})$ and the universal cover is a complete unfurling of those symmetries. In particular, the final cover is the symbol we associate to a tiling, while the universal cover is the symbol that we extract in order to construct a tiling.[7]

For now, we are interested in determining which symbols S correspond to (unfurled) periodic tilings of Euclidean 3-space. Such an S should be covered by $\mathbb{R}^3$ with deck transformations realizing Euclidean

---

[7] In particular, the symbols we associated to tiling were actually their final covers, corresponding to a realization of space by a minimal set of simplices, but expanded by a maximal set of symmetries.

isometries, up to homeomorphism. As symbols also carry the underlying orbifold structure, we also insist that the $\mathbb{R}^3$ cover carry a trivial orbifold structure free of local group actions. Correspondingly, the universal cover must satisfy

$$(r_i r_j)^k (\Gamma \cdot F) \neq (\Gamma \cdot F)$$

for all $k < m_{ij}(\Gamma \cdot F)$.[8] Symbols satisfying these two properties are called *flat*, and it suffices to test flatness on any symbol covered by S.

Theorem 17 ([36]). *If S and S' are 3-dimensional symbols with S covering S', then S is flat if and only if S' is flat.*

This implies that no matter which symbol in a sequence of coverings we choose, we can check for flatness directly on that symbol. This is important because, by design, we are looking for infinite universal covers (corresponding to a tiling). Fortunately, by Theorem 17, we can instead restrict our attention to finite symbols S corresponding to compact orbifolds that are covered by tilings. Any finite flat symbol must use an infinite translational symmetry group of finite index $\mathbb{Z}^d \cong H \subseteq \pi_1(S, s)$ to fill $\mathbb{R}^d$ [74, 75]. Thus, one can enumerate over finite index subgroups of $\pi_1(S, s)$ until identifying one isomorphic to $\mathbb{Z}^3$, and then construct the corresponding finite cover. If the symbol is free of local group actions, then one can check algorithmically whether its simplicial topological realization is homeomorphic to a 3-torus [76]. If so, then the symbol encodes a tiling by unfurling these final translational symmetries. For a complete description of how to perform this check efficiently, see [36]. In practice, we perform these tests using the open-source software Gavrog [77], but have included a summary of the internal sequence of checks below for completeness.

Checking symbols

1. Write down valid regular symbol according to Definition 11, chosen so that the corresponding graph state (if it exists) has desirable properties like locality, self-duality, etc.

2. Check if locally Euclidean [36]. If yes,

3. Construct orbifold graph and orbifold invariants [69] to determine space group. If it corresponds to one of the 219 classified space groups,

4. From fundamental group constructed according to Definition 16 and [73], enumerate over finite index subgroups. Test if the next group isomorphic to $\mathbb{Z}^3$. If yes,

5. Construct the corresponding (possibly toroidal) cover of the symbol as quotient of universal cover by that subgroup.

6. Perform symbol reductions [36] to simplify symbol. Once simplified,

7. Check if the reduced symbol belongs to a known list of simplified symbols. If yes, declare success; the tiling and its corresponding desired graph state exist. If no, continue from Step 4. If there are no more subgroups isomorphic to $\mathbb{Z}^3$, declare failure.

In summary, in order to construct a fault-tolerant cluster state with desirable properties like self-duality and locality, one begins by constructing valid symbols with those desired properties. There are several necessary conditions one can impose to narrow this search, such as demanding regularity and positive curvature on tiles, as they themselves must be spherical. Then, one can follow the above prescription to construct the associated fundamental group explicitly, and search for a subgroup of spanning translational symmetries. If found, the corresponding finite cover encodes a tiling realizing

---

[8]This property, called *semi-good* in [36], is not necessary for the construction of the chain complex; we only care about the underlying manifold structure. However, it is helpful in not double-counting symbols that may correspond to the same underlying tiling.

15

the desired cluster state, which can be built from translations of the unit cell represented by this symbol.

It is worth noting that, instead of building fault-tolerant cluster states based on tilings of 3-space, we could instead try and build CSS codes based on tilings of 2-space. In the 2D setting, the Euler characteristic provides a simple invariant for detecting the zero-curvature of a flat 2D symbol S, which must satisfy $\sum_s m_{01}(s) + m_{12}(s) = (\#s)/2$. However, this limits the average degree of a self-dual tiling to 4, which also determines the locality of the code, and so the toric code is essentially optimal with symmetric noise. The central insight in [31] is that this limitation is relaxed in 3D (where the Euler characteristic is zero) due to the increased length of the chain complex. However, fault-tolerant cluster states still incur a tradeoff between the graph state degree and decoder graph degree in a self-dual lattice, as we will see shortly.

3.2 Generating self-dual fault-tolerant cluster states with local underlying graph states

3.2.1 An optimally local fault-tolerant cluster state

Before detailing a large class of self-dual fault-tolerant cluster states, presented in terms of their corresponding symbols, we begin with a single fully-realized and optimally local example. Broadly, we have discussed two design principles from [31] for building a robust fault-tolerant cluster state: low average degree of the underlying 1-skeleton, contributing to robust error-correction, and underlying graph-state locality, contributing to a low effective error-rate and minimal correlated errors.

Mirroring the 2D case, these two parameters experience a tradeoff. For example, any spanning surface in a 3D tiling must itself obey the zero-curvature of a 2D tiling. As described in Section 2.4, we expect the performance of fault-tolerant cluster states to vary based on the relative strength of different noise sources. In Figures 8 and 9, we depict four highly symmetric self-dual fault-tolerant cluster states spanning this tradeoff. While the first three are well-known, the last one generates an optimally local (regular) graph state, and was found from our search. In particular, the underlying graph state requires only degree-3 connectivity.

The cubic, diamond, and triamond [31] lattices are carried by special self-dual tilings. Up to a maximal group of symmetries, they only have one type of volume, face, edge, and vertex, although only the cubic tiling has a single type of flag. It has been conjectured that, among a subclass of tilings called natural, these are the only self-dual tilings with this property [67].[9] However, the degree-3 self-dual fault-tolerant cluster state described in Figure 8 is also highly symmetric: both the graph state and the decoder graph are regular. In fact, up to a maximal group of symmetries, it has only two types of volumes, faces, edges, and vertices.[10] Although the decoder graph is relatively dense, the graph state locality can make it robust to particular noise processes.

### 3.2.2 Families of degree-3, -4, and -5 self-dual fault-tolerant cluster states

Next, we describe how simple heuristics, combined with the prescription in Section 3.1, can yield a wealth of robust self-dual fault-tolerant cluster state candidates. Although one should ideally choose a cluster state that is optimal for a particular noise model, we focus on only 3-, 4-, and 5-regular graph state families for simplicity. Very broadly, we would expect local families to perform better in the presence of stronger correlated errors, and denser families to perform better in the presence of stronger independent errors. In particular, this search yielded the aforementioned optimally local example, among many others.

---

[9]However, natural tilings play no special role for our purposes, and non-natural tilings with only one type of volume, face, edge, and vertex have been constructed [78]. In fact, this construction also yields a degree-3 graph state, and can be realized as a gluing of 3-cells in the tetrahedral-octahedral honeycomb.
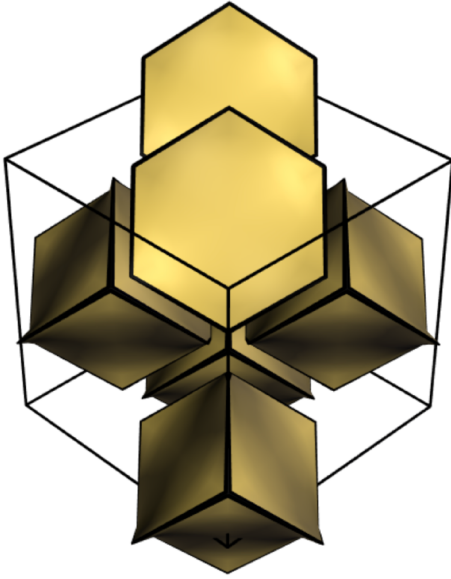
[10]This can be determined by removing one of the $d + 1$ reflection generators, and then counting distinct isomorphism classes of the connected components of the new $(d - 1)$-dimensional symbol.
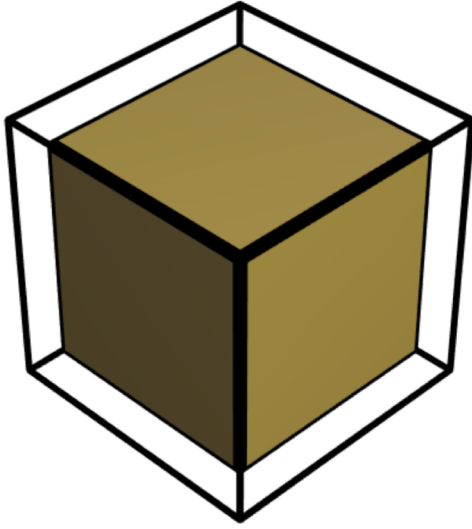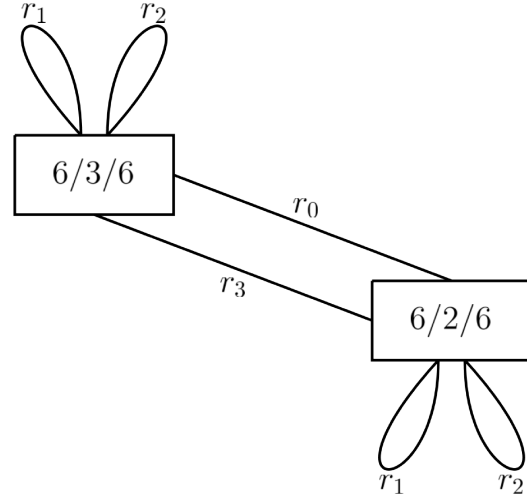
(a) 3-regular graph state, 6-regular decoder graph.
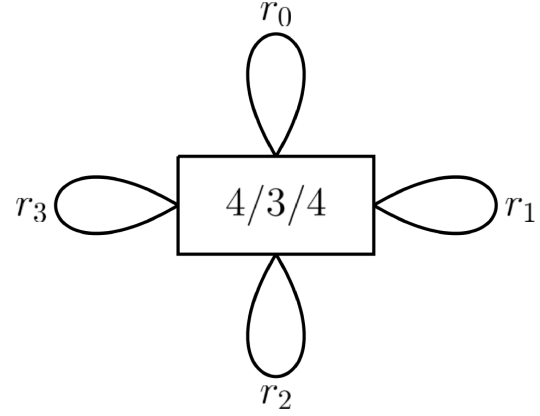
The duality isomorphism is the identity.

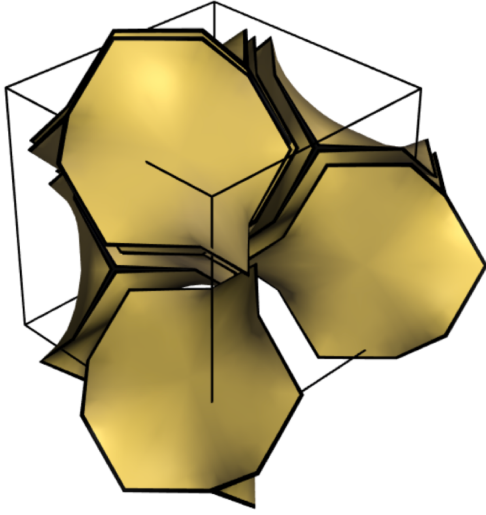(b) 6-regular graph state, 4-regular decoder graph.
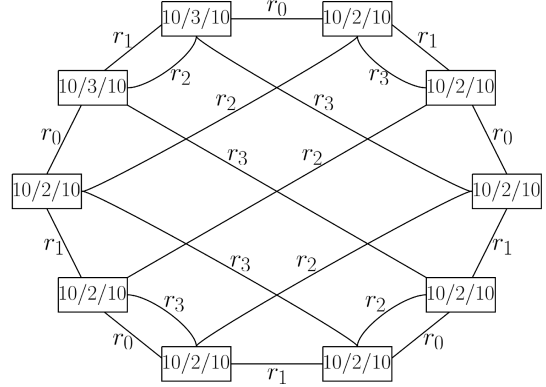
The duality isomorphism is the identity.

Figure 8: Two self-dual fault-tolerant cluster states that balance locality of the decoder graph and the graph state, built from the well-studied cubic and diamond lattices, and their corresponding final covers. The first represents the usual 3D cluster state originally introduced in [25], while the second was studied in [31] (figures rendered with Gavrog [77]).

To begin the search, we must write down a family of valid, regular, self-dual symbols corresponding to n-regular graph states. This can be tricky given the many conditions they must satisfy. However, here is one prescription for doing so.

(i) For a desired n, choose any $k|n$ or $k = 2n$.

(ii) Form a $k \times k$ matrix of nodes defining the vertex set of S.

(iii) Indexing the nodes from $(x, y) = (1, 1)$ in the bottom-left corner to (k, k) in the top-right corner,

(a) 10-regular graph state, 3-regular decoder graph.

The duality isomorphism swaps pairs in each corner and the pair opposite each other along the equator.
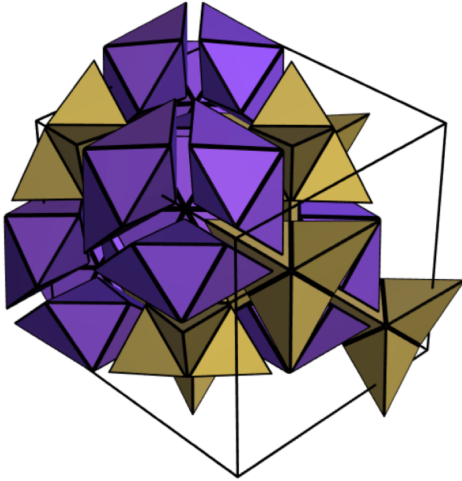
(b) 3-regular graph state, 12-regular decoder graph.
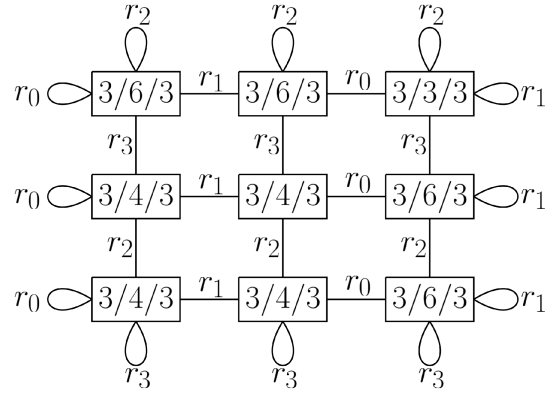
The duality isomorphism is given by reflecting about the line $y = x$.

Figure 9: Two self-dual fault-tolerant cluster states which optimize locality of the decoder graph or the graph state, respectively, and their final covers. The first represents the triamond cluster state studied in [31]. The second is new, and was generated by our search. It corresponds to a fault-tolerant degree-3 graph state.

insert the following edges.

(a) For every $i \neq k$, connect (i, j) and $(i+1, j)$ with an r_0-edge if i is even, or with an r_1-edge if i is odd.

(b) For every $j \neq k$, connect (i, j) and $(i, j+1)$ with an r_3-edge if j is even, or with an r_2-edge if j is odd.

(c) If k is odd, complete the symbol with self-loops along the boundary. If $k = 2n$, complete the symbol with periodic boundaries. If $k \neq 2n$ is even, complete the symbol with either self-loops along the boundary or with periodic boundaries.

(iv) Define $m_{01}(s) = m_{23}(s) = n$ for all $s \in S$.

(a) 3-regular graph state, 12-regular decoder graph.

(b) 4-regular graph state, 6-regular decoder graph.

(c) 6-regular graph state, 4-regular decoder graph.

(d) 10-regular graph state, 3-regular decoder graph.
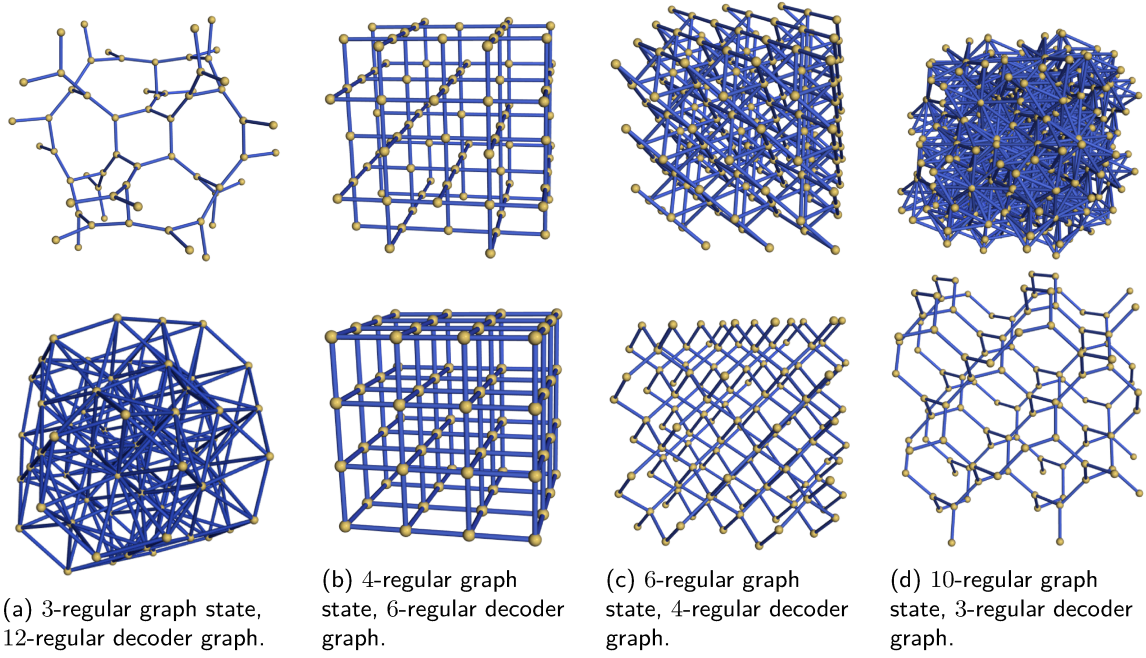
Figure 10: Visualizing the tradeoff between the degree of a graph state (top) and the degree of its corresponding decoder graph (bottom). We comment that locality often plays a role in mitigating errors that may not appear in a gate error model [79], with recent work constructing a 2D quantum code on a degree-3 graph for this purpose [80, 81].
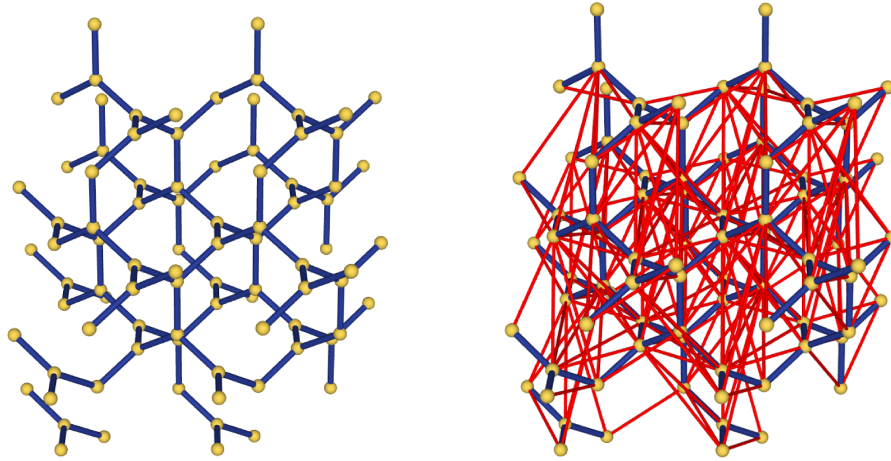
Figure 11: The decoder graph of the triamond lattice both with (right) and without (left) correlated errors. Red bonds correspond to X-type errors during the construction of the state. One must carefully balance the robustness of error-correction against the cost of construction (figures rendered in VESTA [82]).

(v) For every $\langle r_1, r_2 \rangle$-orbit lying either below or intersecting the diagonal, and for all s in the orbit, choose $m_{12}(s) \in \{2, 4, 6, 8, 12\}$ if the orbit size is greater than one, else choose $m_{12}(s) \in \{1, 2, 3, 4, 6\}$. This limitation is imposed by the crystallographic restrictions on rotational symmetries.

(vi) Define the remaining $m_{12}(s_{ij}) = m_{12}(s_{ji})$.

One example with $k = n = 3$ appears in Figure 8. These are valid regular symbols, as for $|i - j| > 1$, each $\langle r_i, r_j \rangle$-orbit forms a cycle of length $1, 2,$ or 4. If the choice of the $m_{12}(s)$ yields a tiling of 3-space, then the underlying graph state will be an (m_{01}, m_{23})-biregular graph. Thus, if this algorithm succeeds, it yields an n-regular graph state. By construction, the symbol is self-dual, with reflection about $y = x$ inducing the duality isomorphism. From this definition, S and its associated $\mathcal{R}$-action are fixed. The only free parameters are the $m_{12}(s)$, which lie in restricted sets, forming $5^{\binom{\lceil \frac{k}{2} \rceil + 1}{2}}$ potential configurations to check for flatness.[11]

Gavrog is an open-source software built explicitly for analyzing tilings [77, 83–86]. Using its internal methods (and more generally GAP [87]), which apply an optimized version of the construction in Section 3.1, we search over these symbols to identify self-dual fault-tolerant cluster states with n-regular underlying graph states. Among the $\gg 10^9$ candidates we searched over, the vast majority generated by $k = 10$, we found 475 unique symbols realizing self-dual cluster states. The search results are summarized in Table 1, which are complete for all but $(n, k) = (5, 10)$. Rendering these can be difficult, depending on the complexity of the symmetry group. However, to fully translate from algebraic representations to concrete tilings, we graph the two examples described by $(n, k) = (3, 3)$ in Figure 12.

| (n,k)	Periodic?	#(Candidates)	#(Cluster States)	Example (m_{12})-vectors
$(3, 1)$	N/A	5	0	$\varnothing$
$(3, 3)$	No	125	2	$(6\ 4\ 4\ 2); (4\ 6\ 6\ 3)$
$(3, 6)$	Yes	15625	21	$(2\ 4\ 4\ 4\ 6\ 6\ 4\ 6\ 6); (6\ 4\ 8\ 4\ 2\ 4\ 8\ 4\ 4); \ldots$
$(4, 1)$	N/A	5	1	(3)[12]
$(4, 2)$	Yes	5	0	$\varnothing$
$(4, 2)$	No	5	0	$\varnothing$
$(4, 4)$	Yes	125	0	$\varnothing$
$(4, 4)$	No	125	1	$(2\ 2\ 2\ 3)$[13]
$(4, 8)$	Yes	9765625	376	$(2\ 2\ 2\ 2\ 2\ 4\ 4\ 2\ 4\ 6\ 6\ 2\ 4\ 6\ 6);$ $(6\ 2\ 6\ 2\ 2\ 4\ 6\ 4\ 6\ 6\ 4\ 2\ 2\ 4\ 2\ 4), \ldots$
$(5, 1)$	N/A	5	0	$\varnothing$
$(5, 5)$	No	15625	4	$(6\ 2\ 4\ 2\ 2\ 2\ 4\ 2\ 2); (4\ 2\ 6\ 2\ 2\ 2\ 6\ 2\ 3); \ldots$
$(5, 10)$	Yes	$\approx 3 \times 10^{10}$	70	$(2\ 2\ 2\ 2\ 2\ 2\ 2\ 4\ 4\ 4\ 2\ 4\ 2\ 4\ 4\ 2\ 4\ 4\ 4\ 2\ 2\ 4\ 4\ 2\ 4);$ $(4\ 2\ 2\ 4\ 2\ 2\ 4\ 4\ 4\ 2\ 2\ 4\ 4\ 2\ 4\ 4\ 4\ 2\ 2\ 4\ 2\ 2\ 4\ 4\ 4), \ldots$

Table 1: A brute-force search for self-dual cluster states with n-regular underlying graph states. The nodes of the $k \times k$ matrix are ordered alphabetically from left-to-right, and then from bottom-to-top. The components of the m_{12}-vectors represent m_{12} for each orbit $\langle r_1, r_2 \rangle \cdot s$ with respect to this ordering, determining the symbol. When possible, a pair of random examples are chosen from each search.

In summary, we can search for fault-tolerant cluster states with desired properties by defining a heuristic on graphs representing symbols that satisfy those properties. Then, one can convert this to a computational problem on groups, and identify candidates whose geometric realization gives the desired cluster state. Finally, one can construct this cluster state from its graphical representation, and study robustness of the resulting state.

Although we stop at degree-5, we could continue this search to higher degree graph states. However, at higher degrees, one might also look for more symmetric tilings using other heuristics. Alternatively, one could search for low degree decoder graphs, rather than low degree graph states, although this constraint is not as straightforward to impose on a symbol.[14]

---

[11]Small changes to this prescription, like mixed boundary conditions or exchanging $r_0 \leftrightarrow r_1$ and $r_1 \leftrightarrow r_2$ also yield valid symbols. We make specific choices to limit our search.

[14]For smaller symbols, one can determine vertex degrees by solving systems of equations involving the Euler characteristic. The maximum m_{12} of the symbol also provides a simple lower-bound on the minimum degree.
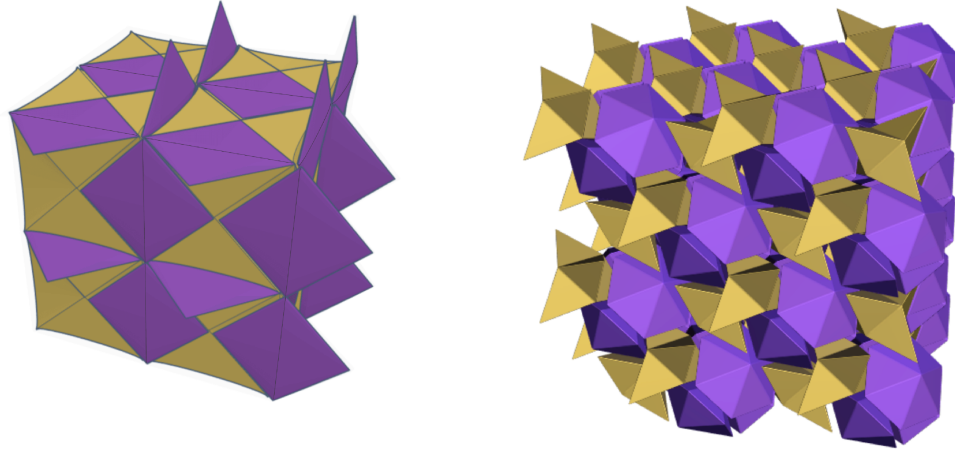
(a) The self-dual tiling corresponding to m_{12}-vector (6 4 4 2). It consists of interwoven (purple) 4-sided cones and 24-sided (yellow) cube-like tiles (drawn in Tinkercad [88]). In particular, the purple objects are four glued triangles with a thin interior.

(b) The self-dual tiling corresponding to m_{12}-vector (4 6 6 3). It consists of interwoven (purple) 12-sided hexagonal bipyramids and 12-sided star-convex polyhedra.

Figure 12: The two self-dual tilings found by searching $(n, k) = (3, 3)$ according to Section 3.2.2. To obtain the corresponding graph states, qubits are placed on faces and edges, with an edge between them if they are adjacent (an edge within a face) in the tiling. For example, the purple objects on the left have ten associated qubits: four face qubits and six edge qubits.

4 Benchmarking crystalline cluster states

Crystals are of fundamental importance in chemistry, and so there are also an abundance of crystals carried by self-dual tilings already in the literature. In addition to building fault-tolerant cluster states from symbols, we can search over existing databases for promising candidates. One example is the RCSR database, which has indexed 38 self-dual tilings [89]. Each lattice is labeled with a three-letter identifier, which we adopt to refer to each tiling.[15]

In [31], the cubic (pcu), diamond (dia), double-edged cubic, and triamond (srs) lattices were benchmarked with a union-find decoder [37] in weighted phenomenological and weighted erasure error models [34]. Although the union-find decoder is highly efficient, it is often suboptimal when compared with the less efficient minimum-weight perfect matching (MWPM) decoder [38, 90, 91]. In this section, we extend this analysis in three ways. First, we benchmark additional examples from the aforementioned set of self-dual fault-tolerant cluster states, along with the optimally local example constructed in Section 3.2. In total, we benchmarked three degree-3 lattices drawn from our search; the one included was the top-performer among the three. Second, we perform these simulations in a circuit-level noise model that captures the tradeoff between robust error-correction and cluster state preparation. Our simulations span four different relative parameter regimes in order to better characterize the effects of this tradeoff. Finally, we use the high-performance MWPM decoder to obtain a closer estimate of the maximum thresholds achievable using this approach.

In fact, we can say more using a well-established connection between thresholds for quantum error-correction and statistical mechanical models. For each of these crystal lattices with cell complex $\mathcal{C}$,

---

[15]From a practical perspective, we focus on benchmarking lattices with smaller unit cells (see Appendix A), as there can be a large gap in the computational expense of different lattice simulations.

we can associate a $\mathbb{Z}_2$-lattice gauge theory with quenched disorder mirroring the random-plaquette $\mathbb{Z}_2$-gauge model on the cubic lattice [7, 9, 10]. The corresponding Hamiltonian is given by assigning spins to edges and defining

$$H = -\sum_{f \in C_2} \tau_f \prod_{e \in f} S_e$$

where $\tau_f \in \{\pm 1\}$ is a quenched random variable and the local $\mathbb{Z}_2$-symmetry is obtained by flipping the spins on all edges incident to a vertex. The optimal phenomenological threshold for each lattice then corresponds to a phase transition of its corresponding model along the Nishimori line [7, 39, 92, 93].

The MWPM decoder selects, for each syndrome, an error class with the most likely individual error. While this is generally suboptimal, it can be viewed from the statistical mechanics perspective as the decoder that chooses a recovery string minimizing the free energy at zero temperature. Correspondingly, up to the relative weighting of edges, our dephasing thresholds can be interpreted as estimates for zero temperature phase transitions for lattice gauge theories with disorder defined on a variety of crystal lattices, whose behavior may be of independent interest. The circuit-level thresholds can also be interpreted similarly according to extensions of this model to correlated errors [39]. However, we emphasize that these should only be regarded as rough estimates, as degeneracy among minimum-weight strings for a syndrome can cause such Monte Carlo estimates to be imprecise [94].

To simulate our noise model on a particular lattice, we first embed the lattice on a 3-torus of linear-size L. We choose a translation-invariant ordering of CZ gates in each face, and then form an augmented 1-skeleton with edges corresponding to Z-error gate failures on the primal qubit support and X-error gate failures on the dual qubit support (see Appendix A). For this initial comparative benchmarking, we make this choice arbitrarily. On the one hand, this gate ordering is likely suboptimal. On the other, we do not benchmark the dual lattice, for which the gate ordering may be worse. The effects of this choice will grow with p_X, and we leave further optimization to future work.

For each edge e, let z_e (x_e) be the number of Z-type (X-type) gate failures that could excite e. Then the probability of exciting edge e with a Z-type gate failure is given by

$$P_{e,Z} := \sum_{k \text{ odd}}^{z_e} \binom{z_e}{k} p_Z^k (1-p_Z)^{(z_e-k)} = \frac{1}{2}(1 - (1-2p_Z)^{z_e}),$$

and the probability of exciting that edge due to an X-type gate failure $P_{e,X}$ is given analogously. The total probability of exciting an edge e is then given by

$$P_e := \sum_{\vec{v} \in \{0,1\}^3 : |\vec{v}| \text{ odd}} p_m^{v_1}(1-p_m)^{(1-v_1)} P_{e,Z}^{v_2}(1-P_{e,Z})^{(1-v_2)} P_{e,X}^{v_3}(1-P_{e,X})^{(1-v_3)}.$$

In each trial, we excite edges according to P_e and form the resulting syndrome. To choose a recovery string, we run MWPM [90, 91, 95, 96] with edges weighted according to $-\ln(P_e)$ [8, 15, 25, 97]. We declare failure if we have introduced a homologically nontrivial cycle after recovery. For each data point we ran 10^6 trials, adding up to approximately $40,000$ computational hours in total, and observed good agreement with previous benchmarks [9, 34].

We reemphasize that this a non-standard error model which is very severe. In particular, the threshold for the 3D cluster state in a standard depolarizing model is $\approx 0.75\%$ [8]. Consequently, these numbers should be used only comparatively: while our error model allows us to more efficiently simulate these lattices, we leave benchmarking in standard models to future work.

We simulate seven different lattices, referring to them by their three-letter RCSR identifiers.[16] We denote the degree-3 cluster state by *bst* for bipyramid-star with triangular faces, and because it is best with respect to locality. Our simulation sizes span $L = 4$ through $L = 12$, depending on the lattice. Although we do observe some finite size effects, it is important to note that several of the lattices have

---

[16]We omit the double-edge cubic lattice, which can be viewed as a concatenation of the 3D cluster state with a phase-flip repetition code, and was studied in [16, 31, 98].

many vertices in their fundamental cell, so that their distance is much greater than L. The additional three lattices were chosen from the RCSR database to validate our hypotheses on the effects of noise; we expect there are many more robust candidates. Renderings of these new lattices, along with a short summary of the self-dual tilings available in the database, are given in Appendix A. In addition, we run four noise models in which p_Z, p_X, and p_m are weighted differently. The resulting threshold estimates are summarized in Table 2, and the raw data is available in Appendix B.

	bst	pcu	cdq	hms	dia	ctn	srs
p_Z (Union-Find [31])	N/A	0.65%	N/A	N/A	0.87%	N/A	0.95%
p_Z	0.35%	0.76%	0.83%	0.94%	1.01%	1.02%	1.16%
$p_Z = 10p_X = 10p_m$	0.32%	0.66%	0.71%	0.80%	0.81%	0.79%	0.75%
$p_Z = p_X = p_m$	0.18%	0.32%	0.32%	0.33%	0.31%	0.31%	0.19%
$p_X = 10p_Z = 10p_m$	0.47%	0.65%	0.58%	0.56%	0.48%	0.51%	0.25%

Table 2: Threshold estimates for seven different lattices in four different noise regimes. The graph state (decoder graph) degree increases (decreases) from left to right. The relative impact of Z-type (X-type) errors decreases (increases) from top to bottom. We include the corresponding union-find thresholds from [31] for comparison. The total error-rate p is defined as the largest individual error-rate among p_Z, p_X, and p_m. The lower thresholds reflect the harsh noise model due to our definition of p_X, p_Z and the the double sampling of errors for each two-qubit gate described in Section 2.4. By comparison, the 3D cluster state has a standard depolarizing threshold of $\approx 0.75\%$ [8].

Supporting the discussion in Section 2.4, the performance of each cluster state generally reflects the tradeoff between the degree of the graph state and of the decoder graph with respect to the relative strength of X-errors and Z-errors. One can see the peak performer moving from right to left as correlated errors become more prevalent, and so we would expect the degree-3 cluster state to be a top performer in an entirely correlated error model. However, as cluster states are constructed entirely from CZ gates, we believe that the noise regime in which p_Z dominates but p_X and p_m remain significant warrants the most interest [16, 56, 57]. Indeed, this reflects the growing effort to take advantage of biased models [53, 99–105]. However, even when noise is symmetric, there are several candidates competitive with the surface code, and it appears that some lattices are generally more robust than others.

Unsurprisingly, the MWPM decoder performs better than the union-find decoder. While this improvement is noticeable across all three compared lattices, the difference in performance only ranges from $\approx 0.1\% - 0.2\%$. Mirroring the results in [37] on the cubic lattice, we interpret this as positive evidence that the union-find decoder may be a promising alternative to MWPM, given its essentially linear-time efficiency. However, it could be the case that this performance gap increases in the presence of correlated errors, for which the matching decoder can be tailored.

One outlier for symmetric and Z-biased noise is the degree-3 cluster state. Although the trend suggests that it may be a top-performer in a fully X-type error model, for more realistic models its thresholds are lower. Generally, achieving fault-tolerance with degree-3 connectivity or 3-local interactions comes with difficult assumptions, or reductions in threshold. However, the sacrifice is often worth the cost, as locality is more valuable than it might appear at the level of gate errors [80, 81, 106]. Consequently, we believe that the performance of the degree-3 cluster state, whose threshold is only a factor of 2 away from the top-performing surface code in our simplified error model, is very encouraging.

5 Conclusions

Although there are many similarities between them, fault-tolerant MBQC has certain freedoms that CBQC with error-correcting codes lacks. We can take advantage of these freedoms to build more robust fault-tolerance protocols beyond those based on error-correcting codes. At the circuit level, this freedom manifests as passing logical information between qubits which are regularly measured and

reinitialized, without a distinction between data and ancilla. This stands in stark contrast to the more restrictive CBCQ fault-tolerance model, where logical information is held statically within the code qubits until a final measurement.

Using combinatorial tiling theory, we gave a prescription for building fault-tolerant cluster states. Different types of errors have different effects, and one must balance locality of both the decoder graph and the graph state according to different relative error strengths and architectural considerations. We also expect many of these cluster states to be particularly resilient to measurement errors, as their relative error strength does not scale with the construction cost of the underlying graph state. Finally, we analyzed several examples of fault-tolerant cluster states in a variety of noise models emulating circuit-level errors, and benchmarked them with the high-performing MWPM decoder. These results draw closer to optimal thresholds for topological MBQC, and relate to statistical mechanical models defined on different crystal structures.

There are several immediate extensions worth exploring. It would be interesting to consider more varied noise models; for example, we expect local graph state constructions to perform well in the presence of correlated noise. Additionally, this framework is naturally robust to leakage, as no qubit is long-lived. In comparison, thresholds of quantum error-correcting codes require additional teleportation gadgets which lower thresholds [107]. Further understanding these cluster states' resilience to hybridized loss and Pauli models would also be of interest [31, 108]. While we have tested only a few examples to take a snapshot of their performance, a more thorough investigation of promising tilings and their robustness is warranted.

There are also several broader avenues worth exploring. Following [31], we have restricted ourselves to geometrically defined cluster states. While this immediately extends to a universal fault-tolerance scheme, it is not required by the underlying homological problem of preserving distinguished primal and dual symmetries. It would be interesting to study non-foliated codes that are not geometrically defined, although it may be difficult to build abstract sparse chain complexes with the desired properties.

To this end, in the prescription for building fault-tolerant cluster states from simplicial structures on orbifolds, we discarded those whose symmetries were not compatible with a Euclidean metric. However, for computational devices with all-to-all connectivity, there is no need to restrict ourselves to periodic tilings of Euclidean 3-space. Candidates with a $\mathbb{Z}^3$ subgroup in their fundamental group are particularly interesting, as this subgroup serves the role of growing the cluster state in the Euclidean case. One could additionally look at generalized notions of periodicity which have less restrictive symmetries.

There has also been significant work aimed at constructing high rate codes in hyperbolic geometries [59, 60]. Often, the encoding rate can be described as a function of the combinatorics of an associated regular tiling [109]. It would be interesting to try to algorithmically construct non-regular self-dual tilings with higher rates, although recognizing symbols corresponding to tilings with negative curvature would likely prove more difficult.

Finally, there is no fundamental barrier to extending this approach to build fault-tolerant cluster states from tilings of higher dimensional spaces. For example, a fault-tolerant cluster state built from a 5D cubic lattice will inherit string-like excitations as a foliation of the 4D toric code. This comes with a number of favorable properties, including high phenomenological thresholds and local decoders, stemming from the 4D toric code's thermodynamic stability [110–112]. However, it also requires higher degree graph states due to the 4D toric code's weight-6 stabilizers, which are damaging at the circuit-level [110]. Analogous to dimension-3, could we use a similar prescription in higher dimensions to construct low-degree fault-tolerant cluster states that exhibit string-like excitations? We could also take the opposite approach, and try minimizing the degree of the decoder graph by constructing a higher-dimensional analogue of the triamond lattice, similar to the hypercubic lattice. Understanding this tradeoff in higher dimensions, which allows for richer symmetry, would be of significant interest.

6 Acknowledgments

References

[1] Panos Aliferis, Daniel Gottesman, and John Preskill. Quantum accuracy threshold for concatenated distance-3 codes. *Quantum Information & Computation*, 6(2):97–165, 2006.

[2] Emanuel Knill, Raymond Laflamme, and W Zurek. Threshold accuracy for quantum computation. *arXiv preprint quant-ph/9610011*, 1996.

[3] Dorit Aharonov and Michael Ben-Or. Fault-tolerant quantum computation with constant error. In *Proceedings of the Twenty-ninth Annual ACM Symposium on Theory of Computing*, pages 176–188. ACM, 1997. DOI: https://doi.org/10.1145/258533.258579.

[4] Barbara M Terhal and Guido Burkard. Fault-tolerant quantum computation for local non-markovian noise. *Physical Review A*, 71(1):012336, 2005. DOI: https://doi.org/10.1103/physreva.71.012336.

[5] Panos Aliferis and Barbara M Terhal. Fault-tolerant quantum computation for local leakage faults. *Quantum Information & Computation*, 7(4):139–156, 2007.

[6] Emanuel Knill. Quantum computing with realistically noisy devices. *Nature*, 434(7029):39, 2005. DOI: https://doi.org/10.1038/nature03350.

[7] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Journal of Mathematical Physics*, 43(9):4452–4505, 2002. DOI: https://doi.org/10.1063/1.1499754.

[8] Robert Raussendorf and Jim Harrington. Fault-tolerant quantum computation with high threshold in two dimensions. *Physical Review Letters*, 98(19):190504, 2007. DOI: https://doi.org/10.1103/physrevlett.98.190504.

[9] Chenyang Wang, Jim Harrington, and John Preskill. Confinement-higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory. *Annals of Physics*, 303(1):31–58, 2003. DOI: https://doi.org/10.1016/s0003-4916(02)00019-2.

[10] Takuya Ohno, Gaku Arakawa, Ikuo Ichinose, and Tetsuo Matsui. Phase structure of the random-plaquette z2 gauge model: accuracy threshold for a toric quantum memory. *Nuclear Physics B*, 697(3):462–480, 2004. DOI: https://doi.org/10.1016/j.nuclphysb.2004.07.003.

[11] DS Wang, AG Fowler, AM Stephens, and LCL Hollenberg. Threshold error rates for the toric and planar codes. *Quantum Information & Computation*, 10(5):456–469, 2010.

[12] Austin G Fowler, Ashley M Stephens, and Peter Groszkowski. High-threshold universal quantum computation on the surface code. *Physical Review A*, 80(5):052312, 2009. DOI: https://doi.org/10.1103/physreva.80.052312.

[13] Austin G Fowler, Adam C Whiteside, and Lloyd CL Hollenberg. Towards practical classical processing for the surface code. *Physical Review Letters*, 108(18):180501, 2012. DOI: https://doi.org/10.1103/physrevlett.108.180501.

[14] Martin Suchara, Andrew W Cross, and Jay M Gambetta. Leakage suppression in the toric code. *Quantum Information & Computation*, 15(11-12):997–1016, 2015. DOI: https://doi.org/10.1109/isit.2015.7282629.

[15] David S Wang, Austin G Fowler, and Lloyd CL Hollenberg. Surface code quantum computing with error rates over 1%. *Physical Review A*, 83(2):020302, 2011. DOI: https://doi.org/10.1103/physreva.83.020302.

[16] Ashley M Stephens, William J Munro, and Kae Nemoto. High-threshold topological quantum error correction against biased noise. *Physical Review A*, 88(6):060301, 2013.

[17] Peter W Shor. Fault-tolerant quantum computation. In *Foundations of Computer Science, 1996. Proceedings., 37th Annual Symposium on*, pages 56–65. IEEE, 1996. DOI: https://doi.org/10.1109/sfcs.1996.548464.

[18] Peter W Shor. Scheme for reducing decoherence in quantum computer memory. *Physical Review A*, 52(4):R2493, 1995. DOI: https://doi.org/10.1103/physreva.52.r2493.

[19] A Robert Calderbank and Peter W Shor. Good quantum error-correcting codes exist. *Physical Review A*, 54(2):1098, 1996. DOI: https://doi.org/10.1103/PhysRevA.54.1098.

[20] Andrew M Steane. Error correcting codes in quantum theory. *Physical Review Letters*, 77(5): 793, 1996. DOI: https://doi.org/10.1103/physrevlett.77.793.

[21] Andrew M Steane. Active stabilization, quantum computation, and quantum state synthesis. *Physical Review Letters*, 78(11):2252, 1997. DOI: https://doi.org/10.1103/physrevlett.78.2252.

[22] Daniel Gottesman. Stabilizer codes and quantum error correction. *arXiv preprint quant-ph/9705052*, 1997.

[23] Robert Raussendorf, Daniel E Browne, and Hans J Briegel. Measurement-based quantum computation on cluster states. *Physical Review A*, 68(2):022312, 2003. DOI: https://doi.org/10.1103/physreva.68.022312.

[24] Hans J Briegel, David E Browne, Wolfgang Dür, Robert Raussendorf, and Maarten Van den Nest. Measurement-based quantum computation. *Nature Physics*, 5(1):19, 2009. DOI: https://doi.org/10.1038/nphys1157.

[25] Robert Raussendorf, Jim Harrington, and Kovid Goyal. Topological fault-tolerance in cluster state quantum computation. *New Journal of Physics*, 9(6):199, 2007. DOI: https://doi.org/10.1088/1367-2630/9/6/199.

[26] Robert Raussendorf, Jim Harrington, and Kovid Goyal. A fault-tolerant one-way quantum computer. *Annals of Physics*, 321(9):2242–2270, 2006. DOI: https://doi.org/10.1016/j.aop.2006.01.012.

[27] Robert Raussendorf, Sergey Bravyi, and Jim Harrington. Long-range quantum entanglement in noisy cluster states. *Physical Review A*, 71(6):062313, 2005. DOI: https://doi.org/10.1103/physreva.71.062313.

[28] A Bolt, G Duclos-Cianci, D Poulin, and TM Stace. Foliated quantum error-correcting codes. *Physical Review Letters*, 117(7):070501, 2016. DOI: https://doi.org/10.1103/physrevlett.117.070501.

[29] A Bolt, D Poulin, and TM Stace. Decoding schemes for foliated sparse quantum error-correcting codes. *Physical Review A*, 98(6):062302, 2018. DOI: https://doi.org/10.1103/physreva.98.062302.

[30] Benjamin Brown and Sam Roberts. Universal fault-tolerant measurement-based quantum computation. *arXiv preprint arXiv:1811.11780*, 2018.

[31] Naomi Nickerson and Héctor Bombín. Measurement based fault tolerance beyond foliation. *arXiv preprint arXiv:1810.09621*, 2018.

[32] John Nguyen, John Pezaris, Gill Pratt, and Steve Ward. Three-dimensional network topologies. In *International Workshop on Parallel Computer Routing and Communication*, pages 101–115. Springer, 1994. DOI: https://doi.org/10.1007/3-540-58429-3̇31.

[33] Behrooz Parhami and Ding-Ming Kwai. A unified formulation of honeycomb and diamond networks. *IEEE Transactions on Parallel and Distributed Systems*, 12(1):74–80, 2001. DOI: https://doi.org/10.1109/71.899940.

[34] Sean D Barrett and Thomas M Stace. Fault tolerant quantum computation with very high threshold for loss errors. *Physical Review Letters*, 105(20):200502, 2010. DOI: https://doi.org/10.1103/physrevlett.105.200502.

[35] Andreas WM Dress. Presentations of discrete groups, acting on simply connected manifolds, in terms of parametrized systems of coxeter matricesa systematic approach. *Advances in Mathematics*, 63(2):196–212, 1987. DOI: https://doi.org/10.1016/0001-8708(87)90053-3.

[36] Olaf Delgado-Friedrichs. Recognition of flat orbifolds and the classification of tilings in R3. *Discrete & Computational Geometry*, 26(4):549–571, 2001. DOI: https://doi.org/10.1007/s00454-001-0022-2.

[37] Nicolas Delfosse and Naomi H Nickerson. Almost-linear time decoding algorithm for topological codes. *arXiv preprint arXiv:1709.06218*, 2017.

[38] Shilin Huang, Michael Newman, and Kenneth R Brown. Fault-tolerant weighted union-find decoding on the toric code. *arXiv preprint arXiv:2004.04693*, 2020.

[39] Christopher T Chubb and Steven T Flammia. Statistical mechanical models for quantum codes with correlated noise. *arXiv preprint arXiv:1809.10704*, 2018.

[40] Hector Bombín and Miguel A Martin-Delgado. Homological error correction: Classical and quantum codes. *Journal of Mathematical Physics*, 48(5):052105, 2007. DOI: https://doi.org/10.1063/1.2731356.

[41] A Yu Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, 2003. DOI: https://doi.org/10.1016/s0003-4916(02)00018-0.

[42] Michael H Freedman, David A Meyer, and Feng Luo. Z2-systolic freedom and quantum codes. In *Mathematics of Quantum Computation*, pages 287–320. CRC Press, 2002. DOI: https://doi.org/10.1201/9781420035377.ch12.

[43] Allen Hatcher. *Algebraic Topology*. Cambridge Univ. Press, Cambridge, 2000. URL https://cds.cern.ch/record/478079.

[44] Sergey Bravyi and Matthew B Hastings. Homological product codes. In *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, pages 273–282. ACM, 2014.

[45] Matthew B Hastings. Quantum codes from high-dimensional manifolds. *arXiv preprint arXiv:1608.05089*, 2016.

[46] Vivien Londe and Anthony Leverrier. Golden codes: quantum LDPC codes built from regular tessellations of hyperbolic 4-manifolds. *Quantum Information & Computation*, 19(5-6):361–391, 2019.

[47] Benjamin Audoux and Alain Couvreur. On tensor products of CSS codes. *Annales de lInstitut Henri Poincaré D*, 2019. DOI: https://doi.org/10.4171/aihpd/71.

[48] Earl Campbell. A theory of single-shot error correction for adversarial noise. *Quantum Science and Technology*, 4:025006, 2019. DOI: https://doi.org/10.1088/2058-9565/aafc8f.

[49] Héctor Bombín. Single-shot fault-tolerant quantum error correction. *Physical Review X*, 5(3): 031043, 2015. DOI: https://doi.org/10.1103/physrevx.5.031043.

[50] Tomas Jochym-O'Connor. Fault-tolerant gates via homological product codes. *Quantum*, 3: Art–No, 2019. DOI: https://doi.org/10.22331/q-2019-02-04-120.

[51] Anirudh Krishna and David Poulin. Fault-tolerant gates on hypergraph product codes. *arXiv preprint arXiv:1909.07424*, 2019.

[52] Héctor Bombín, Ruben S Andrist, Masayuki Ohzeki, Helmut G Katzgraber, and Miguel A Martín-Delgado. Strong resilience of topological codes to depolarization. *Physical Review X*, 2 (2):021004, 2012. DOI: https://doi.org/10.1103/physrevx.2.021004.

[53] Muyuan Li, Daniel Miller, Michael Newman, Yukai Wu, and Kenneth R Brown. 2D compass codes. *Physical Review X*, 9(2):021041, 2019. DOI: https://doi.org/10.1103/physrevx.9.021041.

[54] Aleksander Kubica and Beni Yoshida. Ungauging quantum error-correcting codes. *arXiv preprint arXiv:1805.01836*, 2018.

[55] Austin K Daniel, Rafael N Alexander, and Akimasa Miyake. Computational universality of symmetry-protected topologically ordered cluster phases on 2d archimedean lattices. *Quantum*, 4:228, 2020. DOI: https://doi.org/10.22331/q-2020-02-10-228.

[56] Panos Aliferis, Frederico Brito, David P DiVincenzo, John Preskill, Matthias Steffen, and Barbara M Terhal. Fault-tolerant computing with biased-noise superconducting qubits: a

case study. *New Journal of Physics*, 11(1):013061, 2009. DOI: https://doi.org/10.1088/1367-2630/11/1/013061.

[57] Panos Aliferis and John Preskill. Fault-tolerant quantum computation against biased noise. *Physical Review A*, 78(5):052331, 2008. DOI: https://doi.org/10.1103/physreva.78.052331.

[58] Colin J Trout, Muyuan Li, Mauricio Gutiérrez, Yukai Wu, Sheng-Tao Wang, Luming Duan, and Kenneth R Brown. Simulating the performance of a distance-3 surface code in a linear ion trap. *New Journal of Physics*, 20(4):043038, 2018. DOI: https://doi.org/10.1088/1367-2630/aab341.

[59] Nikolas P Breuckmann, Christophe Vuillot, Earl Campbell, Anirudh Krishna, and Barbara M Terhal. Hyperbolic and semi-hyperbolic surface codes for quantum storage. *Quantum Science and Technology*, 2(3):035007, 2017. DOI: https://doi.org/10.1088/2058-9565/aa7d3b.

[60] Nikolas P Breuckmann and Barbara M Terhal. Constructions and noise threshold of hyperbolic surface codes. *IEEE transactions on Information Theory*, 62(6):3731–3744, 2016. DOI: https://doi.org/10.1109/tit.2016.2555700.

[61] Jonathan Conrad, Christopher Chamberland, Nikolas P Breuckmann, and Barbara M Terhal. The small stellated dodecahedron code and friends. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2123):20170323, 2018. DOI: https://doi.org/10.1098/rsta.2017.0323.

[62] Andreas WM Dress. Regular polytopes and equivariant tessellations from a combinatorial point of view. In *Algebraic Topology Göttingen 1984*, pages 56–72. Springer, 1985. DOI: https://doi.org/10.1007/bfb0074423.

[63] Andreas WM Dress and Daniel Huson. On tilings of the plane. *Geometriae Dedicata*, 24(3): 295–310, 1987. DOI: https://doi.org/10.1007/bf00181602.

[64] Branko Grünbaum and Geoffrey Colin Shephard. *Tilings and Patterns*. Freeman, 1987. DOI: https://doi.org/10.2307/2323457.

[65] Daniel H Huson. The generation and classification of tile-k-transitive tilings of the euclidean plane, the sphere and the hyperbolic plane. *Geometriae Dedicata*, 47(3):269–296, 1993. DOI: https://doi.org/10.1007/bf01263661.

[66] Ludwig Balke and Daniel H Huson. Two-dimensional groups, orbifolds and tilings. *Geometriae Dedicata*, 60(1):89–106, 1996. DOI: https://doi.org/10.1007/bf00150869.

[67] Olaf Delgado-Friedrichs, Michael O'Keeffe, and Omar M. Yaghi. Three-periodic nets and tilings: regular and quasiregular nets. *Acta Crystallographica Section A*, 59(1):22–27, 2003.

[68] Olaf Delgado-Friedrichs, Andreas WM Dress, Daniel H Huson, Jacek Klinowski, and Alan L Mackay. Systematic enumeration of crystalline networks. *Nature*, 400(6745):644, 1999. DOI: https://doi.org/10.1038/23210.

[69] Olaf Delgado Friedrichs and Daniel H Huson. Orbifold triangulations and crystallographic groups. *Periodica Mathematica Hungarica*, 34(1-2):29–55, 1997.

[70] William P Thurston, Steve Kerckhoff, WJ Floyd, and John Willard Milnor. *The Geometry and Topology of Three-Manifolds*. Unpublished, 1997.

[71] ST Hyde, O Delgado-Friedrichs, SJ Ramsden, and Vanessa Robins. Towards enumeration of crystalline frameworks: the 2D hyperbolic approach. *Solid State Sciences*, 8(7):740–752, 2006. DOI: https://doi.org/10.1016/j.solidstatesciences.2006.04.001.

[72] VA Blatov, M O'Keeffe, and DM Proserpio. Vertex-, face-, point-, Schläfli-, and Delaney-symbols in nets, polyhedra and tilings: recommended terminology. *CrystEngComm*, 12(1):44–48, 2010. DOI: https://doi.org/10.1039/b910671e.

[73] Ludwig Balke and Alba Valverde. Chamber systems, coloured graphs and orbifolds. *Contributions to Algebra and Geometry*, 37(1):17–29, 1996.

[74] Ludwig Bieberbach. Über die bewegungsgruppen der euklidischen räume. *Mathematische Annalen*, 70(3):297–336, 1911. DOI: https://doi.org/10.1007/bf01564500.

[75] Ludwig Bieberbach. Über die bewegungsgruppen der euklidischen räume (zweite abhandlung.) die gruppen mit einem endlichen fundamentalbereich. *Mathematische Annalen*, 72(3):400–412, 1912. DOI: https://doi.org/10.1007/bf01456724.

[76] Geoffery Hemion. *The Classification of Knots and 3-Dimensional Spaces*. Oxford University Press, 1992.

[77] Olaf Delgado-Friedrichs. Gavrog. https://github.com/odf/gavrog.

[78] Olaf Delgado-Friedrichs, Michael O'Keeffe, and Omar M. Yaghi. Three-periodic nets and tilings: semiregular nets. *Acta Crystallographica Section A*, 59(6):515–525, 2003. DOI: https://doi.org/10.1107/s0108767302018494.

[79] Natalie C Brown, Michael Newman, and Kenneth R Brown. Handling leakage with subsystem codes. *New Journal of Physics*, 21(1):073055, 2019. DOI: https://doi.org/10.1088/1367-2630/ab3372.

[80] Christopher Chamberland, Guanyu Zhu, Theodore J Yoder, Jared B Hertzberg, and Andrew W Cross. Topological and subsystem codes on low-degree graphs with flag qubits. *Physical Review X*, 10(1):011022, 2020. DOI: https://doi.org/10.1103/physrevx.10.011022.

[81] Christopher Chamberland, Aleksander Kubica, Ted Yoder, and Guanyu Zhu. Triangular color codes on trivalent graphs with flag qubits. *New Journal of Physics*, 22(2):023019, 2020. DOI: https://doi.org/10.1088/1367-2630/ab68fd.

[82] Koichi Momma and Fujio Izumi. Vesta 3 for three-dimensional visualization of crystal, volumetric and morphology data. *Journal of Applied Crystallography*, 44(6):1272–1276, 2011. DOI: https://doi.org/10.1107/s0021889811038970.

[83] Olaf Delgado-Friedrichs and Michael O'Keeffe. Identification of and symmetry computation for crystal nets. *Acta Crystallographica Section A: Foundations of Crystallography*, 59(4):351–360, 2003. DOI: https://doi.org/10.1107/s0108767303012017.

[84] Olaf Delgado-Friedrichs. Barycentric drawings of periodic graphs. In *International Symposium on Graph Drawing*, pages 178–189. Springer, 2003. DOI: https://doi.org/10.1007/978-3-540-24595-7˙17.

[85] Olaf Delgado-Friedrichs and Michael O'Keeffe. Crystal nets as graphs: Terminology and definitions. *Journal of Solid State Chemistry*, 178(8):2480–2485, 2005. DOI: https://doi.org/10.1016/j.jssc.2005.06.011.

[86] Olaf Delgado-Friedrichs. Equilibrium placement of periodic graphs and convexity of plane tilings. *Discrete & Computational Geometry*, 33(1):67–81, 2005. DOI: https://doi.org/10.1007/s00454-004-1147-x.

[87] GAP. *GAP – Groups, Algorithms, and Programming, Version 4.10.2*. The GAP Group, 2019. URL https://www.gap-system.org.

[88] Tinkercad. URL http://www.tinkercad.com.

[89] Michael O'Keeffe, Maxim A Peskov, Stuart J Ramsden, and Omar M Yaghi. The reticular chemistry structure resource (RCSR) database of, and symbols for, crystal nets. *Accounts of Chemical Research*, 41(12):1782–1789, 2008. DOI: https://doi.org/10.1021/ar800124u.

[90] Jack Edmonds. Paths, trees, and flowers. In *Classic Papers in Combinatorics*, pages 361–379. Springer, 2009. DOI: https://doi.org/10.1007/978-0-8176-4842-8˙26.

[91] Vladimir Kolmogorov. Blossom V: a new implementation of a minimum cost perfect matching algorithm. *Mathematical Programming Computation*, 1(1):43–67, 2009. DOI: https://doi.org/10.1007/s12532-009-0002-8.

[92] Hidetoshi Nishimori. Geometry-induced phase transition in the $\pm$J ising model. *Journal of the Physical Society of Japan*, 55(10):3305–3307, 1986. DOI: https://doi.org/10.1143/jpsj.55.3305.

[93] Aleksander Kubica, Michael E Beverland, Fernando Brandão, John Preskill, and Krysta M Svore. Three-dimensional color code thresholds via statistical-mechanical mapping. *Physical Review Letters*, 120(18):180501, 2018. DOI: https://doi.org/10.1103/physrevlett.120.180501.

[94] Thomas M Stace and Sean D Barrett. Error correction and degeneracy in surface codes suffering loss. *Physical Review A*, 81(2):022317, 2010. DOI: https://doi.org/10.1103/physreva.81.022317.

[95] Pierre L'Eculier. Good parameter sets for combined multiple recursive random number generators. *Operations Research*, 47:159–164, 1999. DOI: https://doi.org/10.1287/opre.47.1.159.

[96] Pierre L'Eculier, R. Simard, E. J. Chen, and W. D. Kelton. An objected-oriented random-number package with many long streams and substreams. *Operations Research*, 50:1073–1075, 2002. DOI: https://doi.org/10.1287/opre.50.6.1073.358.

[97] Thomas M Stace, Sean D Barrett, and Andrew C Doherty. Thresholds for topological codes in the presence of loss. *Physical Review Letters*, 102(20):200501, 2009. DOI: https://doi.org/10.1103/physrevlett.102.200501.

[98] Terry Rudolph. Why I am optimistic about the silicon-photonic route to quantum computing. *APL Photonics*, 2(3):030901, 2017. DOI: https://doi.org/10.1063/1.4976737.

[99] David K Tuckett, Stephen D Bartlett, and Steven T Flammia. Ultrahigh error threshold for surface codes with biased noise. *Physical Review Letters*, 120(5):050505, 2018. DOI: https://doi.org/10.1103/physrevlett.120.050505.

[100] David K Tuckett, Andrew S Darmawan, Christopher T Chubb, Sergey Bravyi, Stephen D Bartlett, and Steven T Flammia. Tailoring surface codes for highly biased noise. *Physical Review X*, 9(4):041031, 2019. DOI: https://doi.org/10.1103/physrevx.9.041031.

[101] David K Tuckett, Stephen D Bartlett, Steven T Flammia, and Benjamin J Brown. Fault-tolerant thresholds for the surface code in excess of 5% under biased noise. *Physical Review Letters*, 124 (13):130501, 2020. DOI: https://doi.org/10.1103/physrevlett.124.130501.

[102] Xiaosi Xu, Qi Zhao, Xiao Yuan, and Simon C Benjamin. High-threshold code for modular hardware with asymmetric noise. *Physical Review Applied*, 12(6):064006, 2019. DOI: https://doi.org/10.1103/physrevapplied.12.064006.

[103] Shruti Puri, Lucas St-Jean, Jonathan A Gross, Alexander Grimm, NE Frattini, Pavithran S Iyer, Anirudh Krishna, Steven Touzard, Liang Jiang, Alexandre Blais, et al. Bias-preserving gates with stabilized cat qubits. *arXiv preprint arXiv:1905.00450*, 2019.

[104] Paul Webster, Stephen D Bartlett, and David Poulin. Reducing the overhead for quantum computation when noise is biased. *Physical Review A*, 92(6):062309, 2015. DOI: https://doi.org/10.1103/physreva.92.062309.

[105] Peter Brooks and John Preskill. Fault-tolerant quantum computation with asymmetric Bacon-Shor codes. *Physical Review A*, 87(3):032310, 2013. DOI: https://doi.org/10.1103/physreva.87.032310.

[106] Sergey Bravyi, Guillaume Duclos-Cianci, David Poulin, and Martin Suchara. Subsystem surface codes with three-qubit check operators. *Quantum Information & Computation*, 13(11-12):963–985, 2013.

[107] Martin Suchara, Andrew W Cross, and Jay M Gambetta. Leakage suppression in the toric code. In *Information Theory (ISIT), 2015 IEEE International Symposium on*, pages 1119–1123. IEEE, 2015. DOI: https://doi.org/10.1109/isit.2015.7282629.

[108] Michael Varnava, Daniel E Browne, and Terry Rudolph. Loss tolerant linear optical quantum memory by measurement-based quantum computing. *New Journal of Physics*, 9(6):203, 2007. DOI: https://doi.org/10.1088/1367-2630/9/6/203.

[109] Nikolas P Breuckmann and Vivien Londe. Single-shot decoding of linear rate LDPC quantum codes with high performance. *arXiv preprint arXiv:2001.03568*, 2020.

[110] Kasper Duivenvoorden, Nikolas P Breuckmann, and Barbara M Terhal. Renormalization group decoder for a four-dimensional toric code. *IEEE Transactions on Information Theory*, 65(4): 2545–2562, 2018. DOI: https://doi.org/10.1109/tit.2018.2879937.

[111] Nikolas P Breuckmann, Kasper Duivenvoorden, Dominik Michels, and Barbara M Terhal. Local decoders for the 2D and 4D toric code. *Quantum Information & Computation*, 17(3-4):181–208, 2017.

[112] Aleksander Kubica and John Preskill. Cellular-automaton decoders with provable thresholds for topological codes. *Physical Review Letters*, 123(2):020501, 2019. DOI: https://doi.org/10.1103/physrevlett.123.020501.

A Additional tiling information

Here, we include renderings of the tilings used in our benchmarking simulations. For completeness, we also include a summary of self-dual tilings appearing in the RCSR database. This information can be found at http://rcsr.anu.edu.au/. The code used to construct and perform matching on these lattices, along with additional lattice benchmarks, can be found at https://github.com/leoadec/crystal-thresholds. In particular, unit cells and gate orderings for each lattice can be found in the lib/doc subdirectory. See Figure 14 for a description.

(a) **cdq** (b) **hms** (c) **ctn**
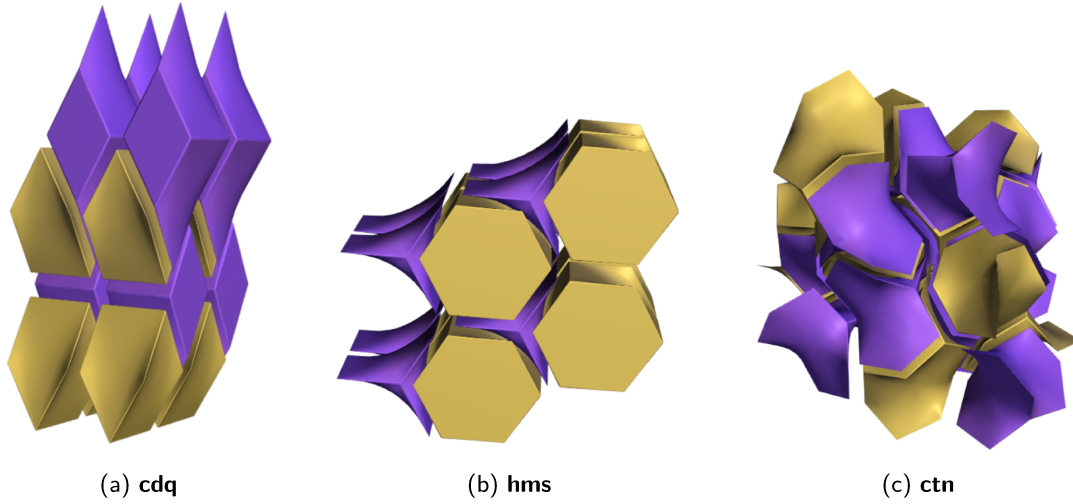
Figure 13: The benchmarked lattices from the RCSR database, ordered from left to right by increasing (decreasing) graph state (decoder graph) degree. For a summary of their parameters, see Table 3.

	bbr	cbs	cdq	cds	ctn	dia	est	ete	fsf	ftw	gsi	hms	hst
Avg. Decoder Graph Degree	4	6	5	4	3.43	4	5	3	5	6	4	4	3.33
Avg. Graph State Degree	7.5	4	4.8	7	8	6	5.375	10.33	5	4	7.5	6	9.6

	lcy	mab	mcf	mco	mgc	pcu	pte	pyr	qtz	rtw	sda	smt	srs
Avg. Decoder Graph Degree	6	6	4	3.33	8	6	4.67	4	6	4.66	6	6	3
Avg. Graph State Degree	5	5.33	6	8	4	4	5.14	6	4.66	5.42	4.33	4.66	10

	sto	svn	swl	sxd	tfa	tfc	ths	tph	ttv	unj	vck	vtx	bst
Avg. Decoder Graph Degree	3.33	7	7	6	3.33	3.33	3	6.86	4.29	4	7	5.33	12
Avg. Graph State Degree	8.8	4.29	3.71	4.33	8	8.8	10.66	4	4.8	6.5	3.71	4.5	3

Table 3: A list of the self-dual 3D nets in the RCSR database, along with their relevant average degrees. We have also included the degree-3 bst lattice, which does not appear in the RCSR database to the best of our knowledge.
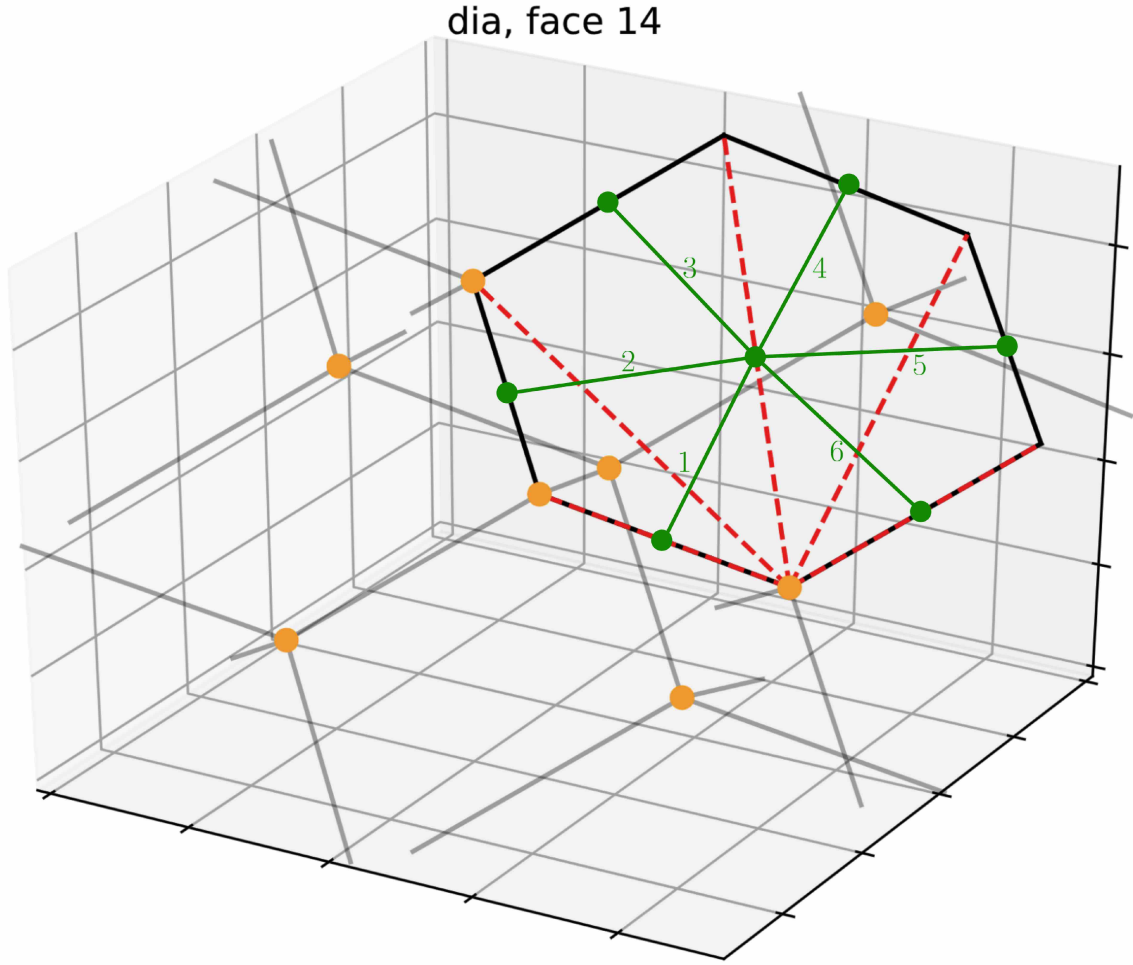
Figure 14: A guiding example for reading the gate orderings in lib/doc taken from the 14th face of the decoder graph of the diamond (dia) lattice. Here, the green is overlayed on the actual decoder graph shown in lib/doc. Qubits and CZ gates (green) are placed on edges and faces. The red dotted lines indicate correlated errors due to X-type errors. The gate ordering always begins with a CZ gate supported on a qubit causing a weight one error, and ends with a CZ gate supported on a qubit causing a weight one error. The ordering proceeds either clockwise or counter-clockwise (the choice of which, in our error model, will not have an effect); in this case, it is labeled clockwise. The unit cell vertices and edges of the tiling are depicted in yellow and black and grey, respectively.

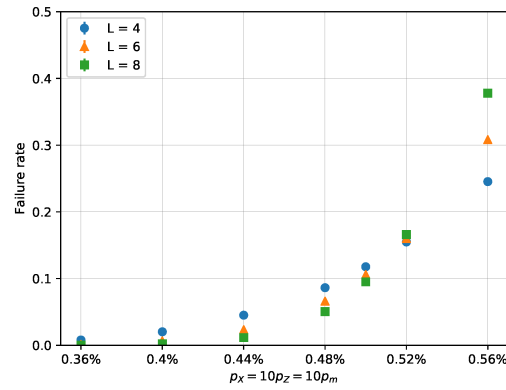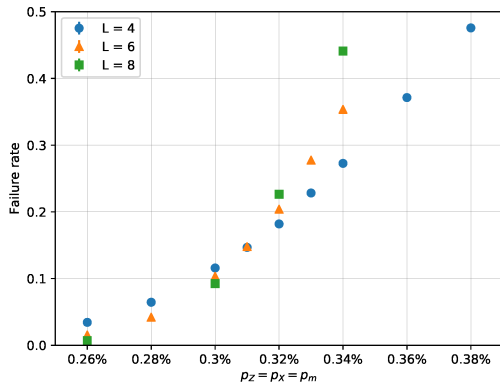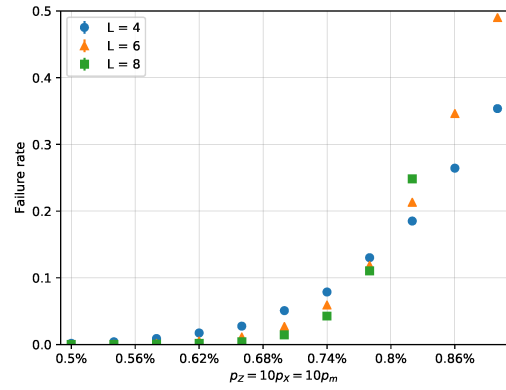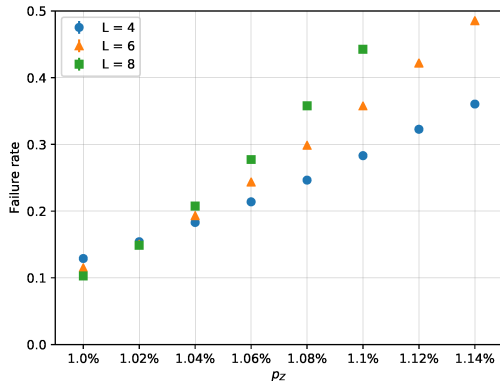# B Numerical results

BST Lattice
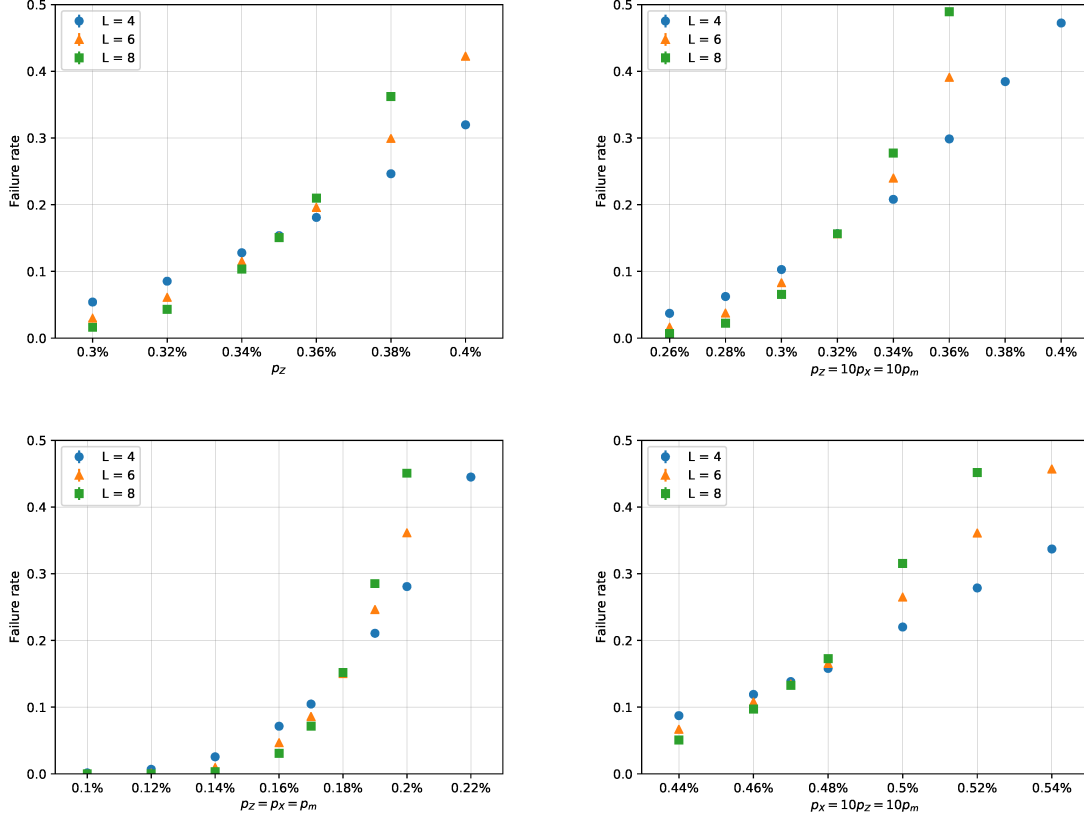
PCU Lattice
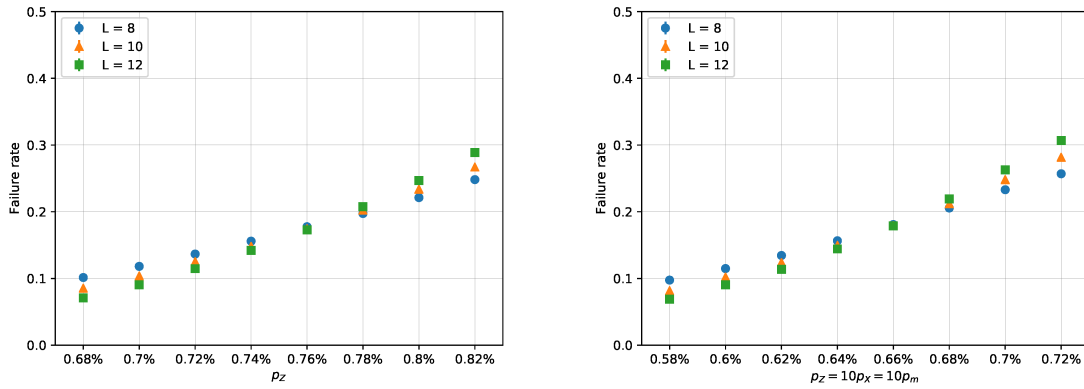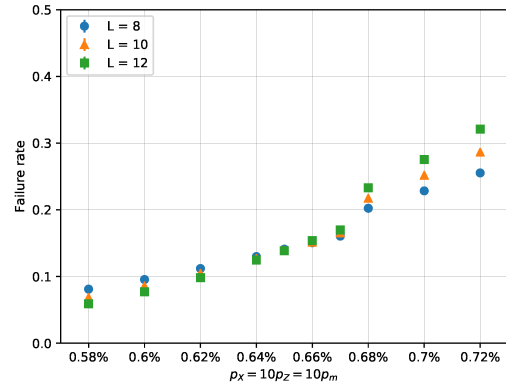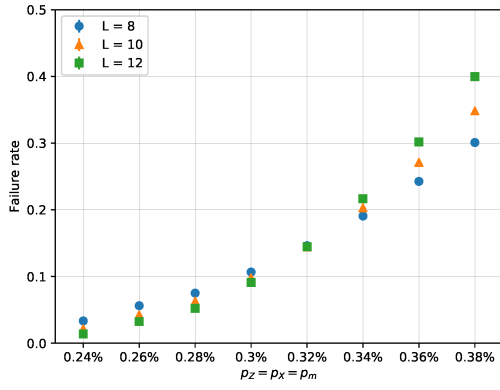
CDQ Lattice
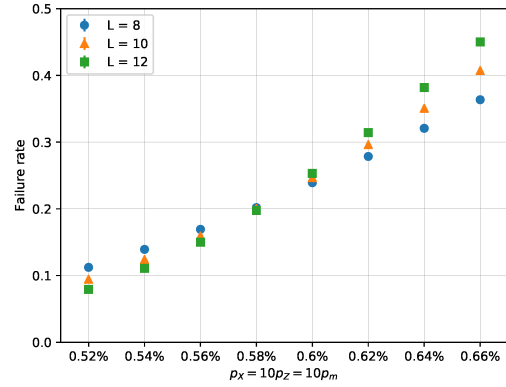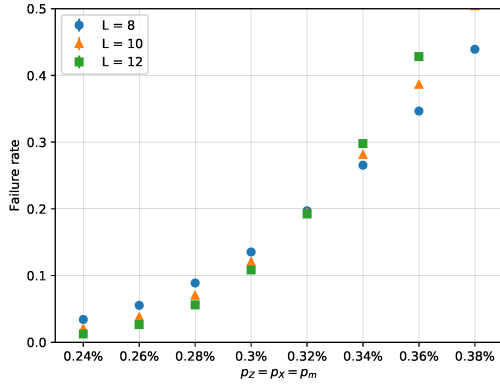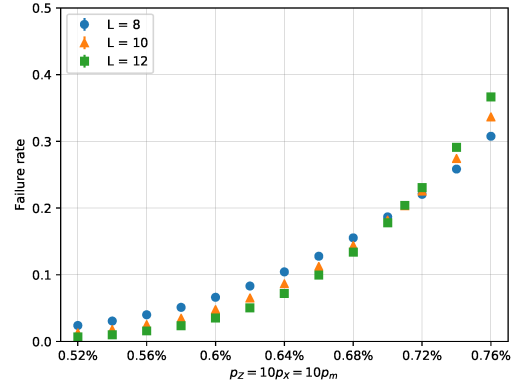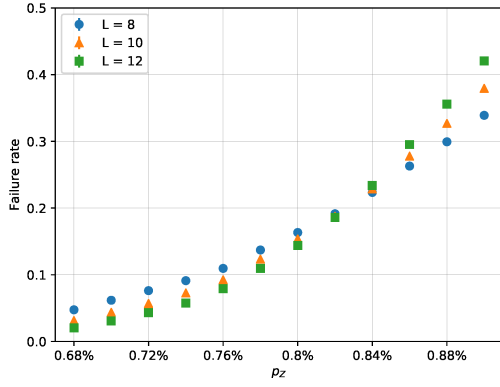
HMS Lattice
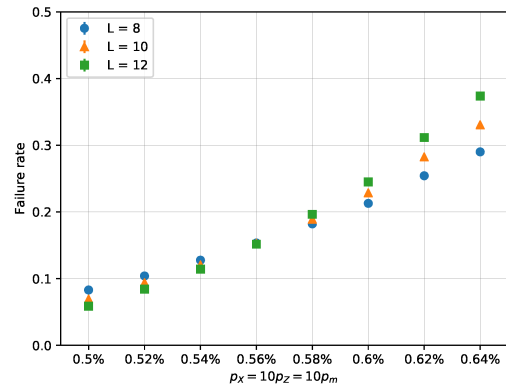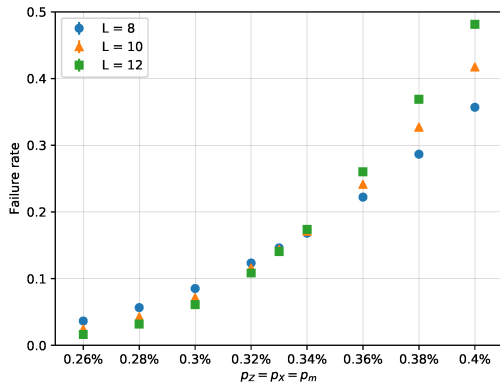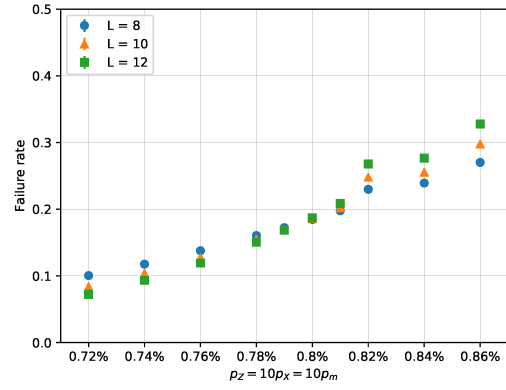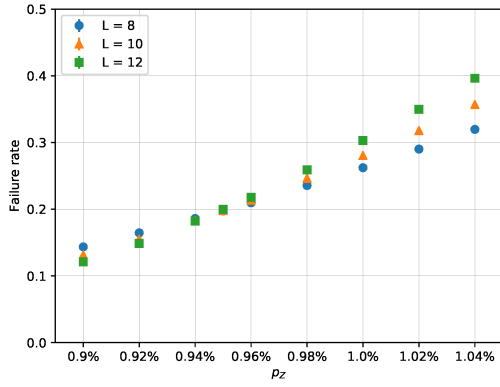
DIA Lattice

CTN Lattice

SRS Lattice