

Quantum error correction with fractal topological codes

Arpit Dua¹, Tomas Jochym-O'Connor^{2,3}, and Guanyu Zhu^{2,3}

¹Department of Physics and Institute for Quantum Information and Matter, California Institute of Technology, Pasadena, CA 91125 USA

²IBM Quantum, IBM T.J. Watson Research Center, Yorktown Heights, NY 10598 USA

³IBM Almaden Research Center, San Jose, CA 95120 USA

Recently, a class of fractal surface codes (FSCs), has been constructed on fractal lattices with Hausdorff dimension $2 + \epsilon$, which admits a fault-tolerant non-Clifford CCZ gate [1]. We investigate the performance of such FSCs as fault-tolerant quantum memories. We prove that there exist decoding strategies with non-zero thresholds for bit-flip and phase-flip errors in the FSCs with Hausdorff dimension $2 + \epsilon$. For the bit-flip errors, we adapt the sweep decoder, developed for string-like syndromes in the regular 3D surface code, to the FSCs by designing suitable modifications on the boundaries of the holes in the fractal lattice. Our adaptation of the sweep decoder for the FSCs maintains its self-correcting and single-shot nature. For the phase-flip errors, we employ the minimum-weight-perfect-matching (MWPM) decoder for the point-like syndromes. We report a sustainable fault-tolerant threshold ($\sim 1.7\%$) under phenomenological noise for the sweep decoder and the code capacity threshold (lower bounded by 2.95%) for the MWPM decoder for a particular FSC with Hausdorff dimension $D_H \approx 2.966$. The latter can be mapped to a lower bound of the critical point of a confinement-Higgs transition on the fractal lattice, which is tunable via the Hausdorff dimension.

Topological stabilizer codes are a highly promising class of codes for scalable architectures of fault-tolerant quantum memories and quantum computation [2–6]. This can be attributed to the geometrically local stabilizer terms and high fault-tolerant thresholds. However, the power of topological stabilizer codes is restricted. For instance, the

Arpit Dua: adua@caltech.edu

Tomas Jochym-O'Connor: tjoc@ibm.com

Guanyu Zhu: guanyu.zhu@ibm.com

Bravyi-König bound provides an upper bound on the set of transversally implementable logical gates for topological stabilizer codes on D -dimensional lattices to be the D -th level of the Clifford hierarchy [7]. Hence, in two-dimensional (2D) topological stabilizer codes, only the Clifford group can be directly implemented fault-tolerantly [8]; in order to have a universal logical gate set, magic-state distillation is required, leading to additional space-time overhead [6, 9, 10]. One potential solution is resorting to the 2D non-Abelian Turaev-Viro codes, which evades the simple stabilizer formalism [11–17]. An alternate proposal is the three-dimensional (3D) topological stabilizer code for which the fault-tolerant gates can reach the third level of the Clifford hierarchy. In particular, there exists a fault-tolerant single-shot implementation of the non-Clifford CCZ gate in the 3D surface code [18, 19], analogous to the implementation of the T gate in the 3D color code [20, 21]. Another advantage of the 3D surface code is that the bit-flip sector of syndromes is string-like for which there exists a cellular automaton-based decoding strategy with a local update rule, called the sweep decoder [22, 23]. This shows that the system is self-correcting for bit-flip errors [24], implying the single-shot error correction property [21]. This remedies the overhead associated with non-local classical communication of the usual decoder [25] in realistic scenarios and also allows hardware-efficient autonomous error correction with dissipation engineering [26] for bit-flip errors.

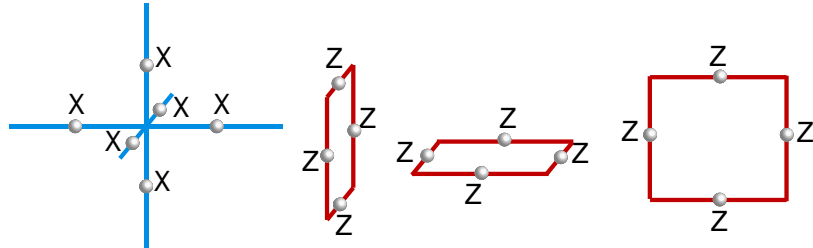
In recent work, the authors have proved that topological order can exist on fractal lattices embedded in D spatial dimensions. In particular, a family of topological stabilizer codes on fractal lattices, embedded in D dimensions, with Hausdorff dimensions $D_H = D - 1 + \epsilon$, where ϵ could be an arbitrarily small nonzero number, is constructed [1]. We refer to this family of codes on fractal lattices as the fractal surface codes (FSCs) [1]. FSCs in 3D, which we simply refer to as FSCs from here on, are constructed by punching homotopically trivial holes with smooth boundaries in the 3D surface code such that the resulting lattice is a fractal. The code distance $d = L$ is preserved under such code puncturing, where L is the linear system size. Moreover, the non-Clifford CCZ gate implementation of the 3D surface code is still possible up to some modifications at the hole boundaries [1]. Surprisingly, due to this possibility, the space-overhead associated with the fault-tolerant universal gate set is reduced to $\mathcal{O}(d^{2+\epsilon})$ compared to $\mathcal{O}(d^3)$ for the 3D surface code. The FSCs still require 3D connectivity of qubits, which can be realized in architectures such as 3D integrated superconducting qubits [27–29] and photonic qubits [30, 31].

In this work, we study the performance of the FSCs as fault-tolerant quantum memories. For FSCs with arbitrary Hausdorff dimension $D_H = 2 + \epsilon$, we prove that there exist decoding strategies with nonzero thresholds for both the string-like and point-like syndromes under the bit-flip and phase-flip errors respectively. Moreover, for a particular FSC with Hausdorff dimension $D_H = \ln 26 / \ln 3 \approx 2.966$, we report the fault-tolerant (sustainable) threshold using a variant of the sweep decoder for the string-like syndromes in the presence of pure bit-flip noise and the code capacity threshold using the minimum weight perfect matching (MWPM) decoder for the point-like syndromes in presence of pure phase-flip errors. We chose this D_H due to numerical limitations; the decoders are proven to work for FSCs with lower Hausdorff dimension $D_H = 2 + \epsilon$ as

well. The sweep decoder needs to be adapted to the presence of the holes in the fractal lattice. We modify the local update rule of the sweep rule near the hole boundaries such that the self-correcting and the single-shot properties of the decoder are maintained for any fractal dimension $D_H=2+\epsilon$. We demonstrate the sweep decoder performance for a particular FSC with $D_H\approx 2.966$, as mentioned, by calculating logical failure rates under N rounds of noisy syndrome measurements. In the limit $N\rightarrow\infty$, we obtain the so-called sustainable threshold. The intuition for why the modified sweep decoder maintains a threshold at fractal dimensions is due to the distribution and spacing of the holes in the lattice with a scale-invariant pattern. Namely, while the presence of holes may delay the speed at which a particular error is cleaned up, the holes do not affect the correctability of the various connected components of errors.

1 Fractal surface code

In order to define the FSC, we start with the 3D surface code. The 3D surface code is defined on the cubic lattice \mathcal{L} , where qubits sit on the edges of the lattice. The code is defined by the following stabilizer generators, $A_v = \prod_{e\ni v} X_e$ and $B_f = \prod_{e\in f} Z_e$ where v , e and f denote the vertices, edges and faces (0-, 1-, 2-cells) of \mathcal{L} , and Z_e and X_e are Pauli operators associated with a given qubit sitting on the edge e . Pictorially, the bulk stabilizer generators, A_v and B_f up to their translates are as follows,



$$(1)$$

We also consider the dual lattice \mathcal{L}^* obtained from the original lattice \mathcal{L} with the following conversion of the cells: $e \leftrightarrow f$ and $v \leftrightarrow c$, where c denotes a cube (3-cell). See Fig. 1(a) for these stabilizers on both the original and dual lattice and the latter will be useful to describe the sweep decoder. Since the qubits now sit on the faces f^* of the dual lattice \mathcal{L}^* , the vertex stabilizers are supported on cubes c^* of \mathcal{L}^* while the plaquette stabilizers are associated to the edges e^* of \mathcal{L}^* such that the support is on faces f^* neighboring edge e^* . Mathematically, $A_v \equiv A_{c^*} = \prod_{f^*\in c^*} X_{f^*}$ and $B_f \equiv B_{e^*} = \prod_{f^*\ni e^*} Z_{f^*}$. The violations of X -stabilizers A_v on \mathcal{L} give rise to the e -excitations or point-like syndromes S_0 as boundaries of string-like Z error chain E_1 (1-chain), *i.e.*, $S_0 = \partial E_1$; those of Z -stabilizers B_{e^*} on \mathcal{L}^* lead to m -excitations or string-like syndromes S_1^* as boundaries of membrane-like X error chain E_2^* (2-chain), *i.e.*, $S_1^* = \partial E_2^*$. The associated pair of anti-commuting logical operators is given by (1) a string-like operator $\bar{Z} = \prod_{e\in[c_1]} Z_e$ supported on an equivalent class of 1-cycles on \mathcal{L} belonging to the 1st relative

homology group over \mathbb{Z}_2 coefficients, *i.e.*, $[c_1] \in H_1(\mathcal{L}, \mathcal{B}_e; \mathbb{Z}_2)$, which describes the class of non-contractible 1-cycles including the absolute cycles (loops) and relative cycles (open strings) terminated on the *rough boundaries* (with dangling edges), also called *e*-boundaries \mathcal{B}_e since they condense the *e*-excitations, as illustrated in Fig. 1(b); (2) a membrane-like operator $\bar{X} = \prod_{f \in [c_2^*]} X_{f^*}$ supported on an equivalent class of 2-cycles on \mathcal{L}^* belonging to the 2nd relative homology group, *i.e.*, $[c_2^*] \in H_2(\mathcal{L}^*, \mathcal{B}_m^*; \mathbb{Z}_2)$, which describes the class of non-contractible absolute 2-cycles (closed membranes) and relative 2-cycles (open membranes) terminated on the *smooth boundaries* (without dangling edges), also called *m*-boundaries \mathcal{B}_m^* since they condense the *m*-excitations.

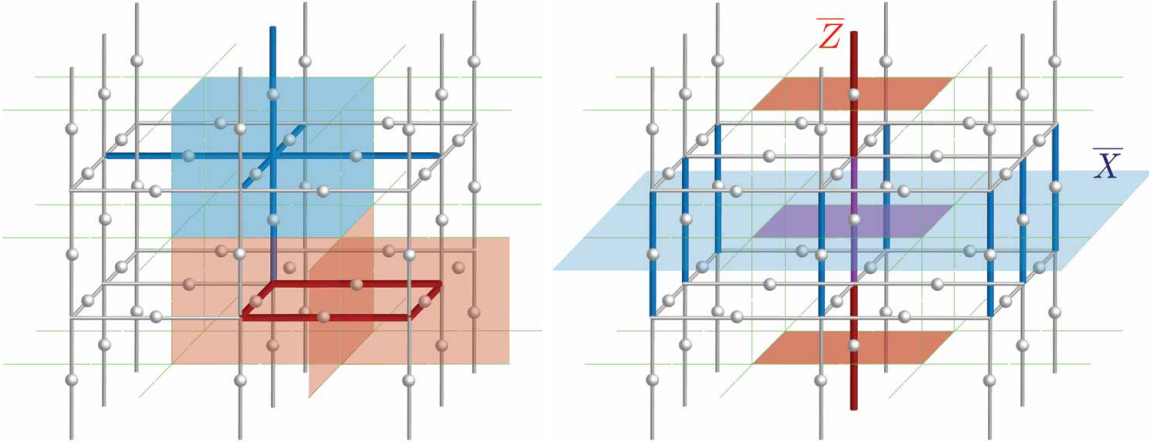


Figure 1: Left: 3D surface code on the original lattice \mathcal{L} (black) and the dual lattice \mathcal{L}^* (thin green). The edges highlighted in red (red faces on \mathcal{L}^*) denote the action of Pauli X operators, while the edges highlighted in blue (blue faces on \mathcal{L}^*) denote the action of Pauli Z operators. The vertex stabilizer $A_v = \prod_{e \ni v} X_e$ and plaquette stabilizers $B_f = \prod_{e \in f} Z_e$, where e and f refer to the edges and faces of \mathcal{L} , are shown in blue and red respectively. Right: The string logical operator \bar{Z} and the membrane logical operator \bar{X} are shown..

The FSC is obtained from the 3D surface code on a cubic lattice by punching holes with *m*-boundaries which we refer to as *m*-holes, in an iterative manner. The first iteration involves starting with the original cubic lattice of the surface code, which we call the level-0 cube. In the ℓ^{th} iteration, one divides each level- ℓ cube equally into $a \times a \times a$ level- $(\ell+1)$ cubes with linear size $1/a$ of a level- ℓ cube, and punch an *m*-hole in the center occupying $b \times b \times b$ cubes, and we obtain the Fractal Cube (FC) geometry for level ℓ as $FC(a, b, \ell)$. The fractal cube geometry is generated in the asymptotic limit, *i.e.*, $FC(a, b) \equiv \lim_{\ell \rightarrow \infty} FC(a, b, \ell)$. Requiring $b < a$, we get an asymptotic fractal cube geometry with Hausdorff dimension $D_H \rightarrow 2$ in the limit $b/a \rightarrow 1$ [1].

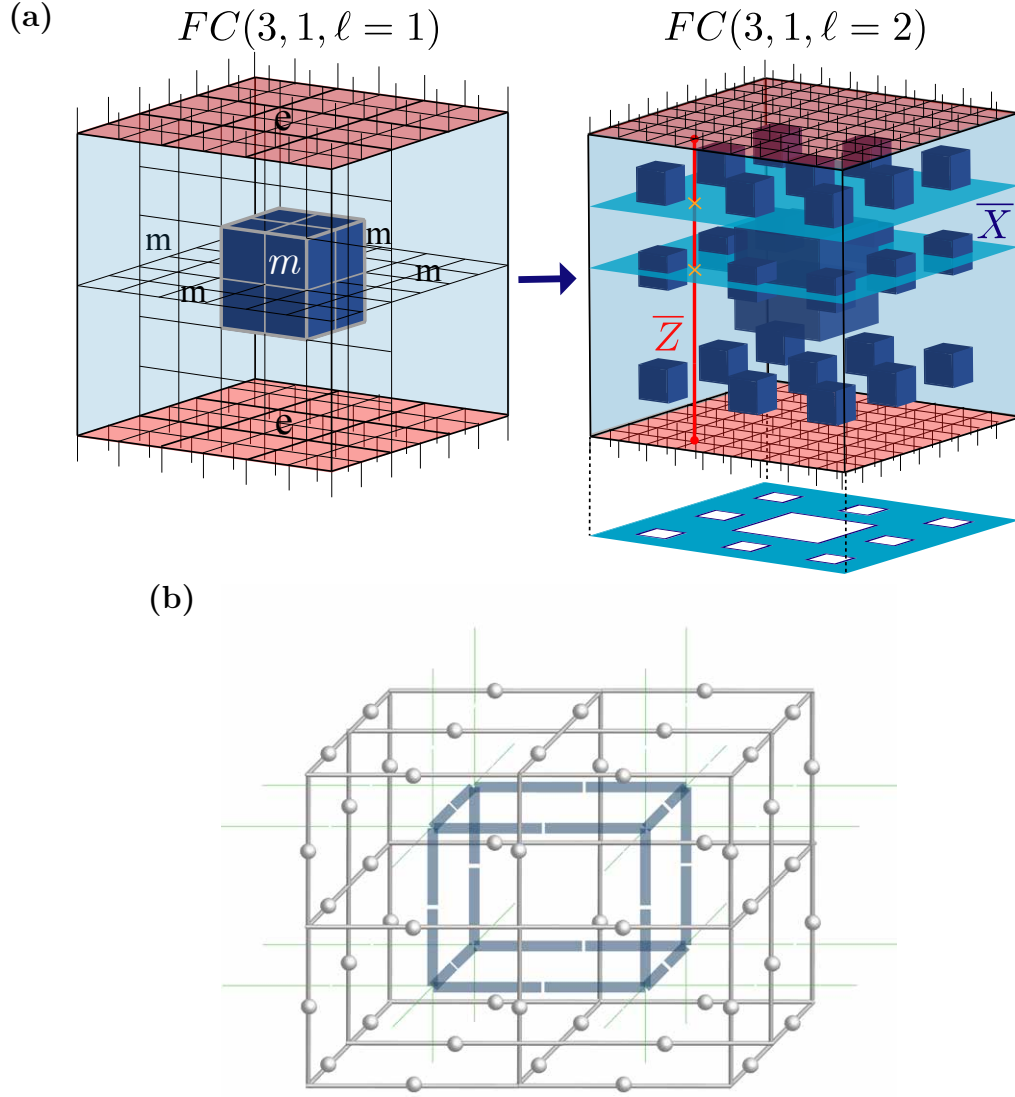


Figure 2: (a) Levels 1 and 2 of the fractal surface code are shown as $FC(3, 1, \ell = 1)$ and $FC(3, 1, \ell = 2)$. For level 1, the single m -hole is shown, and for level 2, an extra iteration of punched holes is shown. The anti-commuting pair of logical operators \bar{X} and \bar{Z} are also shown for level 2. The minimum weight representation of membrane operator \bar{X} , which is interrupted by the holes and forms a Sierpinski carpet, is shown separately at the bottom. In (b), we show a $1 \times 1 \times 1$ size m -hole. On the original lattice, the missing edges form the hole, while on the dual lattice, the m -hole is shown as a cube with gray-blue edges, where the missing qubits are given by the faces of the cube..

The code space \mathcal{H}_C is mathematically determined by the 1st relative homology group $H_1(\tilde{\mathcal{L}}, \mathcal{B}_e; \mathbb{Z}_2)$, defined on the fractal lattice $\tilde{\mathcal{L}}$. As mentioned, the m -holes in the FSC are chosen to be homeomorphic to 3-dimensional balls, and they have a trivial

contribution to the homology group, meaning that they do not encode any additional logical qubit (see Sec. II in Supplementary for a counting argument). This can be seen by the fact that logical- Z string can neither terminate on the boundaries of m -holes (due to the condensation property) nor enclose these holes since they are 3-balls. Thus, the code space can be expressed as

$$\mathcal{H}_C = \mathbb{C}^{H_1(\tilde{\mathcal{L}}, \mathcal{B}_e; \mathbb{Z}_2)} = \mathbb{C}^{\mathbb{Z}_2} = \mathbb{C}^2, \quad (2)$$

where the non-trivial contribution of \mathbb{Z}_2 to H_1 comes from the logical- Z string connecting the top and bottom e -boundaries circumventing any m -holes (see Ref. [1] for a detailed proof). We hence have one encoded logical qubit. The unique class of the dual logical- X membranes, determined by the 2nd relative homology group which is isomorphic to the above 1st relative homology group¹, is hence $H_2(\tilde{\mathcal{L}}^*, \mathcal{B}_m^*; \mathbb{Z}_2) \cong H_1(\tilde{\mathcal{L}}, \mathcal{B}_e; \mathbb{Z}_2) = \mathbb{Z}_2$. The minimum weight logical- X membrane now terminates on the m -holes and is supported on a Sierpinski carpet with Hausdorff dimension $D_H^X = \ln 8 / \ln 3 \approx 1.893$ and corresponding X -distance $d_X \sim L^{1.893}$ in the case of $FC(3, 1)$, as illustrated in Fig. 2. The overall distance is determined by the minimum length of the logical- Z string $d = \min\{d_X, d_Z\} = d_Z = L$. Since the X error is still membrane-like with string-like syndromes on its boundary, the self-correction and single-shot nature of error correction for the loop sector are still expected to survive in the case of imperfect syndrome measurements.

Decoding the FSC

We study the performance of the 3D FSC under two i.i.d. Pauli noise models, *i.e.*, the bit-flip and the phase-flip error models. The first one is described by the following Pauli noise channel,

$$\rho \rightarrow \sum_{\mathbf{e}} [p_X X_{\mathbf{e}} \rho X_{\mathbf{e}} + (1 - p_X) \rho], \quad (3)$$

where ρ is the density matrix describing the state of the code and p_X is the single-qubit X error rate. To get the phase-flip noise model, switch X to Z in Eq. (3).

According to the asymptotic definition of the fractal cube geometry, the threshold of the FSC, $p_{\text{th}}(FC(a, b))$ is defined as

$$p_{\text{th}}(FC(a, b)) = \lim_{\ell \rightarrow \infty} p_{\text{th}}(FC(a, b, \ell)) \quad (4)$$

where $p_{\text{th}}(FC(a, b, \ell))$ is the threshold for level ℓ of the FSC. In this work, we evaluate the thresholds for levels $\ell = 1, 2$ and estimate the threshold for the limit $\ell \rightarrow \infty$ *i.e.*, for $FC(a, b)$.

As mentioned, we use a cellular automaton-based decoder called the sweep decoder to decode the X errors and the MWPM decoder to decode the Z errors. However, the

¹This is due to the Poincaré-Lefschetz duality, which is also consistent with the structure of a CSS code.

sweep decoder needs to be adapted to account for the m -holes. In Sec. 2 below, we first describe how the sweep decoder works for the 3D surface code and then generalize to the FSC. The sweep decoder can be defined generally for any causal codes [23]. However, we restrict our description of the decoder to the 3D surface code on a cubic lattice for clarity.

2 Sweep Decoder

The main ingredient of the sweep decoder is the sweep rule. Intuitively speaking, the sweep rule takes into input the syndrome and applies a correction operator such that a part of the syndrome is swept towards a particular direction called the sweep direction. Eventually, under enough such sweeps, the syndrome is cleaned. For perfect measurements, the syndrome consists of closed loops; hence partial sections of a given loop move towards its remaining sections and close on to them. We illustrate the sweep rule in Fig. 3(a).

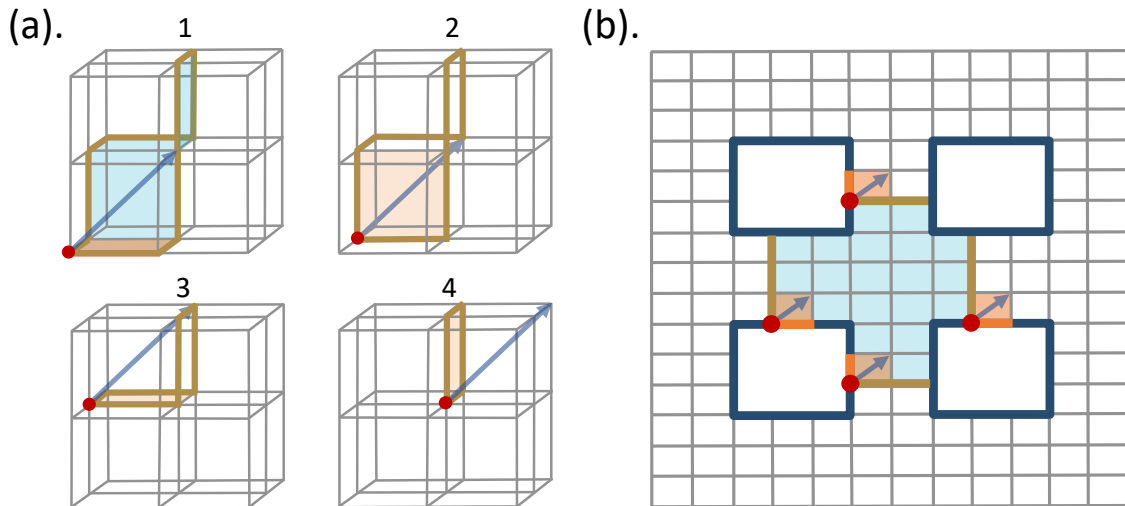


Figure 3: Illustration of the sweep rule. The sweep direction is shown using arrows next to extremal vertices in red. The plaquettes shaded in blue denote the errors that generate the syndrome, and the plaquettes shaded in orange are the ones that are swept or on which a correction is applied. (a) Sweep rule applied four times (1-4) on the syndrome (brown) that sits on the edges e^* of the dual cubic lattice \mathcal{L}^* . (b) A 2D projection of a trapped syndrome configuration (brown) between m -holes (dark blue). The modified sweep rule allows us to put imaginary syndromes on an edge (brown) in the future on the surface of the hole.

We now state the sweep rule for the regular 3D surface code on a cubic lattice \mathcal{L} formally. To do so, it is convenient to consider the dual lattice \mathcal{L}^* such that the syndromes live on the edges e^* and qubits on the faces f^* . Note that from here on, we work in this dual picture. Based on the sweep direction \hat{s} , each edge attached to a vertex is assigned to be in the future or the past of that vertex. For example, consider

a vertex \mathbf{v}^* and for each edge \mathbf{e}^* emanating from \mathbf{v}^* , associate a unit vector $\hat{\mathbf{v}}_{\mathbf{e}^*}^*$ in the direction along the edge \mathbf{e}^* starting from \mathbf{v}^* . If the dot product of $\hat{\mathbf{s}}$ with the $\hat{\mathbf{v}}_{\mathbf{e}^*}^*$ is non-negative, the edge is in the future of that vertex.

If all nontrivial syndrome edges emanating from the vertex are in the future of that vertex, we refer to the vertex as extremal. If a vertex \mathbf{v}^* is extremal, then we apply a correction operator on the face (qubit) \mathbf{f}^* enclosed by the future syndrome edges connected to \mathbf{v}^* such that the syndrome on all the edges of \mathbf{f}^* , $\mathbf{e}^* \subset \mathbf{f}^*$ is flipped. This effectively sweeps the future syndrome edges connected to the vertex to the other two edges of the face. We apply this step for all extremal vertices in parallel. Under a sufficient number of such consecutive implementations of the sweep rule, most syndrome configurations clean up in a time or number of steps that scales linearly with the system size.

We now discuss the sweep rule for the 3D FSC on a lattice with open boundaries. For the fractal lattice $\tilde{\mathcal{L}}^*$, we need a modification to the sweep rule because configurations of syndromes joining the holes can get trapped if one uses the regular sweep rule. These configurations are resistant to the regular sweep rule because there are no extremal vertices even if the sweep direction is changed. This is similar to the configuration of a pair of parallel lines that span the lattice. We illustrate a trapped configuration between holes of the 3D FSC via a 2D projection. To relieve these trapped configurations connecting holes, the sweep rule needs to be modified on the hole boundary².

For the fractal lattice $\tilde{\mathcal{L}}^*$ considered in this work, even though some of the qubits are removed in the holes made in a regular lattice \mathcal{L}^* , our goal is to imitate the sweep decoder in the regular lattice as much as possible. The modification to the sweep rule is only on the vertices \mathbf{v}_h^* that are at the boundary of the m -hole. Note that in the (original) fractal lattice \mathcal{L} , these correspond to the cubes \mathbf{c}_h that form the outermost layer of cubes inside the m -hole as shown in Fig. 2(b). Even though there are no qubits on the hole faces \mathbf{f}_h^* , we include the vertices \mathbf{v}_h^* in sweep indices, which is the set of vertices considered for application of the sweep rule. Now, we note that in the case of perfect syndrome measurements, for the bulk vertices, only an even number of syndrome edges in \mathbf{e}^* can be incident on each vertex. This is not necessarily true for the vertices \mathbf{v}_h^* . If a vertex on the hole boundary \mathbf{v}_h^* is connected to zero or more than one syndrome edge, then we apply the regular sweep rule on that vertex. If \mathbf{v}_h^* has exactly one syndrome edge connected to it, then we first check whether this syndrome edge is in its future or not. If yes, then this vertex is already extremal; however, since there is only one syndrome edge, the sweep rule is unable to specify the correction. We check the future edges of this vertex that lie on the surface of the hole and, hence, have no associated stabilizer. Among these edges, we choose one randomly and mark it as an imaginary syndrome edge associated with the vertex. Once we have included this imaginary syndrome, the regular sweep rule can specify a correction operator for the extremal vertex \mathbf{v}_h^* . To summarize, the modification to the sweep rule is as follows:

²This differs from the technique of alternating sweep directions used to address syndromes trapped at the global boundary as addressed in Ref. [23].

the vertices on the hole boundary are allowed in the sweep indices, and if a hole vertex has a single nontrivial syndrome edge connected to it, then one out of its future edges which sits on the hole boundary is assigned a syndrome.

For the case of imperfect measurements with an error rate q , we implement the modified sweep rule without change. The intuition behind this is as follows: faulty measurement could lead to broken string-like syndromes due to certain missing syndrome edges, “ghost” (false) syndromes, or, generally speaking, local deformation of the syndromes, all with small probability q . This leads to missing or wrong updates in certain local regions, which is effectively converted again to the phase-flip error with a small probability in the next round. Therefore, the decoder, applied as if to a memory with pure X error, is still expected to have a threshold.

In Sec. 2.1 below, we prove that there exist non-zero thresholds for the bit-flip noise with the sweep decoder. The essence of the proof is that the holes can, in a sense, be treated like errors since, in the worst case, any error that attaches to a hole may take extra time that is linear in that hole size to be corrected. However, due to the scale-invariance of the fractal, the number of holes of a given size drops off exponentially. Hence, the errors that attach themselves to the holes can be cleaned up sufficiently quickly with high probability since they will be far away from other holes of similar size. More specifically, while large holes are close to many smaller holes, their distance from a hole of a similar size is on the order of the size of the hole itself. As such, any smaller error will only, at most, feel the effect of one of the larger holes (not multiple) that it connects to. Therefore, following similar arguments for the 3D surface code, error patches of large size will be increasingly improbable with growing size, and while the correction of an error may be delayed if connected to a large hole, it will not result in connecting multiple large holes. It will eventually be cleaned up by the sweep decoder in a time that is linear in the size of the region that bounds the error and the holes connected to it. In the proof, we formalize these concepts in the language of error chunks and connected components as discussed in Refs. [22, 32].

2.1 Proof of sweep decoder threshold in the FSC

Let \mathcal{E} be the set of all possible Pauli X errors in the 3D fractal code. By definition, the fractal will have hole sizes at each iteration of cubic length $H_i = \alpha D^i$, where $\alpha = b/a \in (0, 1)$ is related to the a and b terms introduced in the definition of the fractal codes $\text{FC}(a, b)$ and D is the smallest sized cube when partitioning. We define all single-qubit errors to be level-0 chunks. All subsequent level chunks are defined recursively as follows: a level- n chunk is the disjoint union of two level- $(n - 1)$ chunks, $E^{[n]} = E_1^{[n-1]} \sqcup E_2^{[n-1]}$ such that the diameter of the level- n chunk is smaller or equal to $\alpha D^n/2$, that is $\text{diam}(E[n]) \leq \alpha D^n/2$.

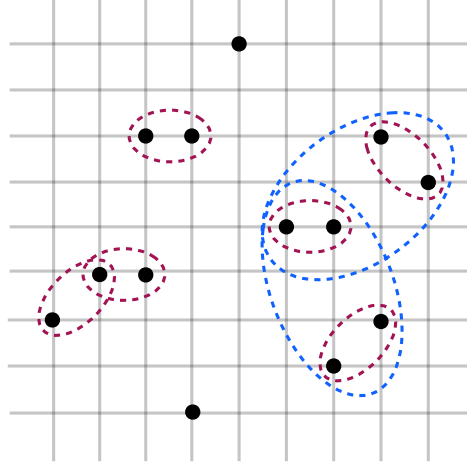


Figure 4: Example of different level- n chunks of errors in two dimensions. The black dots represent errors belonging to E_0 by definition. In this example, the level-1 chunks, dashed red lines, are pairs of errors from E_0 that are distance at most two from one another (in Manhattan distance). The level-2 chunks are formed from pairs of disjoint level-1 chunks such that all errors are within distance four from one another (again, Manhattan distance). Since there are no pair of disjoint level-2 chunks, there are no level-3 chunks (and higher). All errors that are not surrounded by a dashed line will therefore be in $F_0 = E_0 \setminus E_1$, all errors that are only surrounded by red dashed lines will be in F_1 , and finally, the errors surrounded by the blue dashed lines will belong to F_2 .

Given some error pattern E , we define the set of errors E_n to be the union over all level- n chunks, and as such: $E = E_0 \supseteq E_1 \supseteq E_2 \supseteq \dots \supseteq E_{m+1} = \emptyset$, where m is the smallest integer such that E_{m+1} is empty which will always be satisfied as the configuration space of errors \mathcal{E} is countable. Finally, by defining $F_i = E_i \setminus E_{i+1}$, every individual error from E can be classified into one of the sets F_0, \dots, F_m , which we call the chunk decomposition of E . See Fig. 4 for an illustration of different level- n chunks.

We would like to prove then that given an independent noise model for each of the physical qubits, the probability of generating a high-level chunk decreases exponentially with the level of the chunk, even in the presence of holes. We do so following the techniques developed in percolation theory, mirroring the procedure from Refs. [22, 32].

Consider a random error $E \in \mathcal{E}$ of independently generated single qubit Pauli X errors. Let $B_n(x)$ be a fixed box of linear size $H_n = \alpha D^n$ centered at x and $B_n^+(x)$ to be a box of linear size $3H_n$ centered at x . We define the following probabilities:

$$p_n(x) = \Pr[B_n(x) \text{ has a non-zero overlap with level-}n\text{chunk of } E]$$

$$\tilde{p}_n(x) = \Pr[B_n^+(x) \text{ contains a level-}n \text{ chunk of } E]$$

$$q_n(x) = \Pr[B_n^+(x) \text{ contains 2 disjoint level-}(n-1)\text{chunks of } E]$$

$$r_n(x) = \Pr[B_n^+(x) \text{ contains a level-}(n-1) \text{ chunk of } E]$$

For a given x , if $B_n(x)$ has a non-zero overlap with a level- n chunk since the diameter of a given level- n chunk is at most $\alpha D^n/2$, that chunk will necessarily be contained within $B_n^+(x)$. Moreover, if $B_n^+(x)$ contains a level- n chunk then by definition it must contain 2 disjoint level- $(n-1)$ chunks. As such, $p_n(x) \leq \tilde{p}_n(x) \leq q_n(x)$. Moreover, since an event that contributes to $r_n(x)$ is the result of two independent events contributing to $q_n(x)$, by the van den Berg and Kesten inequality, we must have $q_n(x) \leq r_n(x)^2$.

Since the box $B_n^+(x)$ is the disjoint union of $(3D)^3$ boxes of linear size H_{n-1} , we have the following:

$$r_n(x) \leq \sum_{x'} p_{n-1}(x') \leq (3D)^3 \max_{x'} p_{n-1}(x')$$

and thus,

$$p_n(x) \leq r_n(x)^2 \leq (3D)^6 \max_{x'} p_{n-1}^2(x').$$

Finally, if we iterate the process over all n :

$$p_n(x) \leq (3D)^{-6} \left((3D)^6 p_0 \right)^{2^n},$$

and p_0 is just the bare single qubit error rate ϵ . Thus, the probability of a given chunk is exponentially suppressed in its size. The final piece that is needed before arguing for the ability of the sweep decoder to succeed with high probability is the notion of a connected component. A set of errors $E \in \mathcal{E}$ is a R -connected component if it cannot be split into two disjoint non-empty sets $E = M_1 \sqcup M_2$ such that the distance $d(M_1, M_2) > R$. The critical fact is that for a given error E , any set of errors belonging to F_n will necessarily belong to a (H_n) -connected component that is sufficiently separated away from other elements of E_n . The following Lemma summarizes this.

Lemma 1 ([32]). *Let $H_n = \alpha D^n$, such that $D \geq 6$ and let $M \in \mathcal{E}$ be a set of errors belonging to a H_n -connected component of F_n . Then, M has a diameter $\leq H_n$ and $d(M, E_n \setminus M) > H_{n+1}/3$.*

Proof. The claim is that for any pair of errors $m \in M \in F_n = E_n \setminus E_{n+1}$ and $p \in E_n$ we have the following: $d(m, p) \leq H_n$ or $d(m, p) > H_{n+1}/3$. We prove this by contradiction, that is suppose $H_n < d(m, p) \leq H_{n+1}/3$. Let M_n, P_n be the level- n chunk to which m and p belong to, respectively. Therefore, by definition, since their respective diameters are bounded by $H_n/2$ and $H_n < d(m, p)$, they must be disjoint. However, we can also bound the overall distance between M_n and P_n :

$$\begin{aligned} d(M_n, P_n) &\leq \text{diam}(M_n) + d(m, p) + \text{diam}(P_n) \\ &\leq H_n/2 + H_{n+1}/3 + H_n/2 \\ &= \alpha D^n + \alpha D^{n+1}/3 \\ &\leq \alpha D^{n+1}/6 + \alpha D^{n+1}/3 = H_n/2, \end{aligned}$$

as such the union of M_n and P_n form a level- $(n+1)$ chunk, which is a contradiction to $M_n \subseteq M \in F_n$. \square

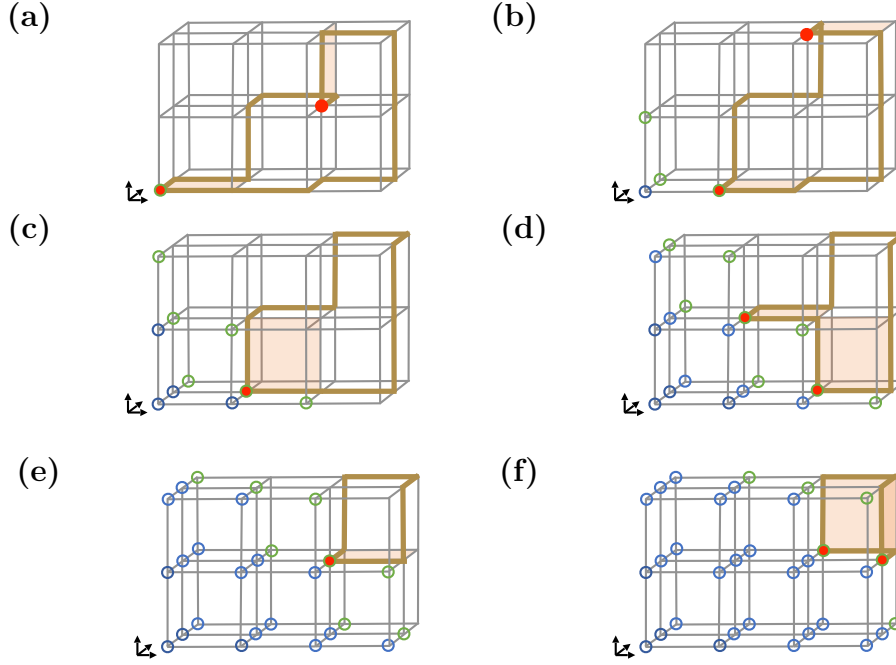


Figure 5: Example of an error sweeping. Initial error configuration is given in (a) with sweep direction in the $(+1, +1, +1)$ direction. At each step, we are guaranteed to sweep away from the green circles if an error is present there. Green circles represent a plane moving through space such that the syndrome is guaranteed to no longer be in the past light cone of this plane relative to the sweep direction. Blue circles represent vertices that are in the past of the plane. Extremal vertices are given by circles filled in red; notice that whenever the syndrome is present at a green circle, the corresponding vertex is extremal.

The conclusion of this Lemma is to separate out all errors into their various connected components and to show that the decoder will successfully address each connected component separately as they are sufficiently far away from one another. The complication arises due to the presence of holes, which can delay the cleaning up of a component, namely if an H_n -connected component of errors (thus of size $\mathcal{O}(D^n)$) was connected to a hole of size $\mathcal{O}(D^p)$, where $p > n$, then the resulting correction would take time $\mathcal{O}(D^p)$ (that is linear with the larger hole size). We show below that our variant of the sweep decoder can successively correct for larger and larger errors, even in the presence of holes.

In order to simplify the discussion of the sweep correction of errors, we provide a few useful definitions. Suppose we have an error $E \in \mathcal{E}$, we define the *error envelope* $V(E)$ of E to be the smallest cuboid that encases E . This definition will be useful for upper-bounding the sweep time of errors. We begin by presenting the following result for the sweeping of an individual error membrane without the presence of holes; we also give an illustration of a sweeping in Fig 5.

Lemma 2. *Consider an error membrane $E \in \mathcal{E}$ with error envelope $V(E)$ whose linear dimensions are: l_x, l_y, l_z . The sweep decoder will correct for such an error in at most*

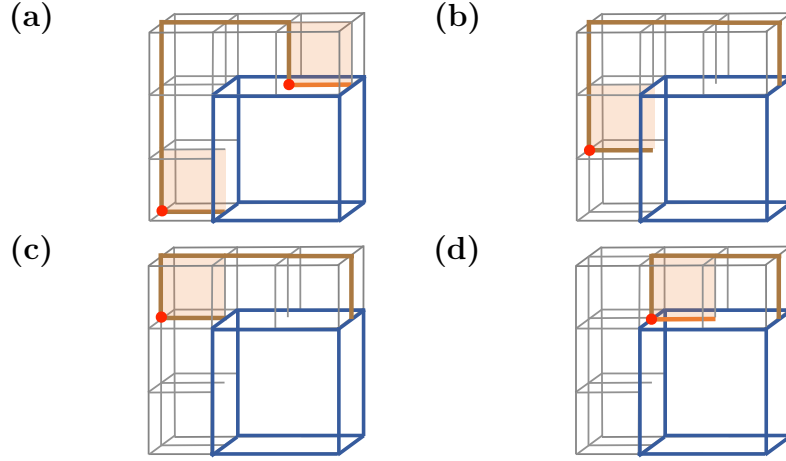


Figure 6: Example of an error sweeping connected to a hole. Initial error syndrome given in brown in (a) with sweep direction in the $(+1, +1, +1)$ direction. The hole is shown by the blue edges. Any sweep correction cannot take one outside the set of edges given since they are all within a cuboid surrounding the error syndrome and hole. The extremal vertices are shown by red points at each step, imaginary syndromes are introduced along the surface of the hole in orange to make the appropriate vertices extremal as described in the modified sweep decoder section..

$(l_x + l_y + l_z - 1)$ sweep steps.

Proof. Without loss of generality, suppose the error envelope $V(E)$ is formed by the cube whose linear coordinates are given by $[0, l_i]$, where $i \in \{x, y, z\}$ and the sweep direction follows the vector $(+1, +1, +1)$. Then, in the first step, the boundary of the error will be swept away from the corner $(0, 0, 0)$, and thus the error can no longer touch this point. In the second step, the error will be swept away from the three points distance 1 away from the origin in Manhattan distance, that is $(1, 0, 0)$, $(0, 1, 0)$, $(0, 0, 1)$. In an iterative manner, in the n -th time step, the error will be necessarily swept away from all points that are distance $(n - 1)$ in from the origin. Since the furthest point from the origin is at the opposite corner of the cuboid (l_x, l_y, l_z) and any error containing that point must have at least one point that distance 2 closer to the origin, we are guaranteed to clean up the error after $(l_x + l_y + l_z - 1)$ time steps. \square

In the presence of holes, errors may take longer to be corrected; however, we can also bound the number of time steps an individual error membrane will take to be corrected. This is pictorially shown in Fig. 6, where the error envelope must also surround the hole.

Corollary 1. *Given an error membrane $E \in \mathcal{E}$ that is connected to holes H_1, \dots, H_j . Consider the error envelope $V(E \cup H_1 \cup \dots \cup H_j)$ that encompasses the error and all holes whose linear dimensions are l_x, l_y, l_z . Then, the sweep decoder will correct for such an error in at most $(l_x + l_y + l_z - 1)$ sweep steps.*

Proof. The proof of the previous Lemma can be generalized to account for the case where the error membrane is connected to multiple holes. As stated, if we choose an

error envelope that encompasses all of the holes connected to the membrane (as well as the membrane itself), then the resulting correction will never leave the envelope. Thus, when sweeping from one of the corners in the sweep direction, we are guaranteed to clean the error in the number of sweep steps given. \square

The upshot of the above Lemma and Corollary is that a connected component of radius R will be corrected in time $\mathcal{O}(R)$ if not connected to any holes, while it will be corrected in time at most $\mathcal{O}(R + R_h)$ if connected to a hole of radius R_h . If connected to multiple holes, the corresponding correction time will be linear in the radius of the largest hole or the error itself, whichever is larger.

Let $E \in \mathcal{E}$ be an error instance. Consider all H_0 -connected components of F_0 that are distance greater than H_0 from any hole, denoted by Q_0 . Given the diameter of these components is at most H_0 , we know the decoder will correct them in time linear to the diameter, assuming there is no interference from other errors or holes. Given a connected component Q_0 is the distance at least H_0 away from any hole and distance $H_1/3$ away from any other error, this connected component will successfully be corrected. As such, all errors F_0 will be successfully corrected unless they were within distance H_0 from any hole; we label the remaining errors as $T_0 = F_0 \setminus Q_0$.

Consider now the elements from F_1 . Again, we break them into two classes, the elements Q_1 which are in H_1 -connected components whose distance is greater than H_1 from any hole of size $H_{k>1}$, and the errors T_1 which are $F_1 \setminus Q_1$. Any error in Q_1 will be corrected in time linear in H_1 as these elements are either independently corrected or they are affected by either the uncorrected errors T_0 or holes of linear size H_1 . However, since both these holes and errors are of size at most H_1 , they will only affect the cleanup time and size of the correction bubble of Q_1 by a constant factor in H_1 . The critical point here is that any error from Q_1 can only see their correction time increase by $\mathcal{O}(H_1)$ due to the uncorrected errors from F_0 or holes of size H_1 . Since neither of these objects is larger than the element from Q_1 , which is also of size $\mathcal{O}(H_1)$, such objects will not affect the decoder's ability to clean up an element from Q_1 .

We iterate this process for any n . Consider elements from F_n , breaking them into two classes: the elements labeled Q_n , which are H_n -connected components whose distance is greater than H_n from any hole of larger size $H_{k>n}$ and the complement of such elements $T_n = F_n \setminus Q_n$. Any element of Q_n will be cleaned up in time linear in H_n as it will either attach itself to a hole of similar size (or smaller) or will be affected by smaller uncorrected errors $T_{j<n}$, in either case this will not change the cleanup time for such an error, and it will be corrected. Thus by induction any level- n error that is smaller than the system size will be corrected in time at most linear in the size of the largest chunk. Therefore, to summarize, errors from Q_n are corrected in time $\mathcal{O}(H_n) = \mathcal{O}(D^n)$, these errors are either connected components of this size or connected to a hole of the given size.

2.2 Sweep decoder simulation algorithm

We now list the steps of the numerical algorithm for simulating the sweep decoder as follows-

1. $N - 1$ rounds of a) generating data errors \vec{E} with probability p_X , b) performing syndrome measurements with error probability q , and c) applying sweep rule (modified version in case of FSC) x times. We set $x = 1$ in our simulations. We change the sweep direction after every y rounds on the lattice with boundaries. We set $y = \log L$ where L is the linear size, measured in terms of the number of cubes in the original cubic lattice (vertices in the dual lattice) along one dimension. Each implementation of the sweep rule updates the data error \vec{E} to the data error times correction operator.
2. After the above $N - 1$ rounds, generate data errors again with probability p_X and perform syndrome measurements. Assume perfect syndrome measurements for this last round.
3. Timeout session: sweep rule is implemented for T steps, and sweep direction is changed after every t steps. We set $T = 32L$ and $t = L$.
4. If the syndrome is not cleaned or if the product of total error and correction acts nontrivially on the logical subspace, the decoder fails.

2.3 Sweep Decoder performance for FSC

We implemented the sweep decoder for the FSC on the fractal lattices $\tilde{\mathcal{L}}^*$ with boundaries. We study the performance of the sweep decoder for an error model with both phase-flip and measurement errors. We consider the measurement error rate q to be the same as the physical error rate p_X , *i.e.*, $q = p_X$. The detailed algorithm is presented in Methods.

We show the performance of the sweep decoder for $N = 1025$ rounds in Fig. 7 (a)-(b) and for $N = 1, 33$ rounds in Sec. III in Supplementary. We also summarize the results for the thresholds of the sweep decoder for different levels and number of rounds of stabilizer measurements, N in Fig. 7 (c). The last round is assumed to have perfect measurements³ while $N - 1$ rounds before the last round involve noisy measurements. $N = 1$ means only one round of stabilizer measurements, and those are perfect. We tabulate the numerical values of the obtained sweep decoder thresholds below,

ℓ	0	1	2
$N = 1$	15.625(8)%	15.59(2)%	15.57(2)%
$N = 33$	2.400(1)%	2.471(3)%	2.455(2)%
$N = 1025$	1.727(3)%	1.7331(7)%	1.7262(7)%

³This is a standard assumption in quantum error correction studies, which is valid at the readout stage where measurement errors become equivalent to data qubit errors.

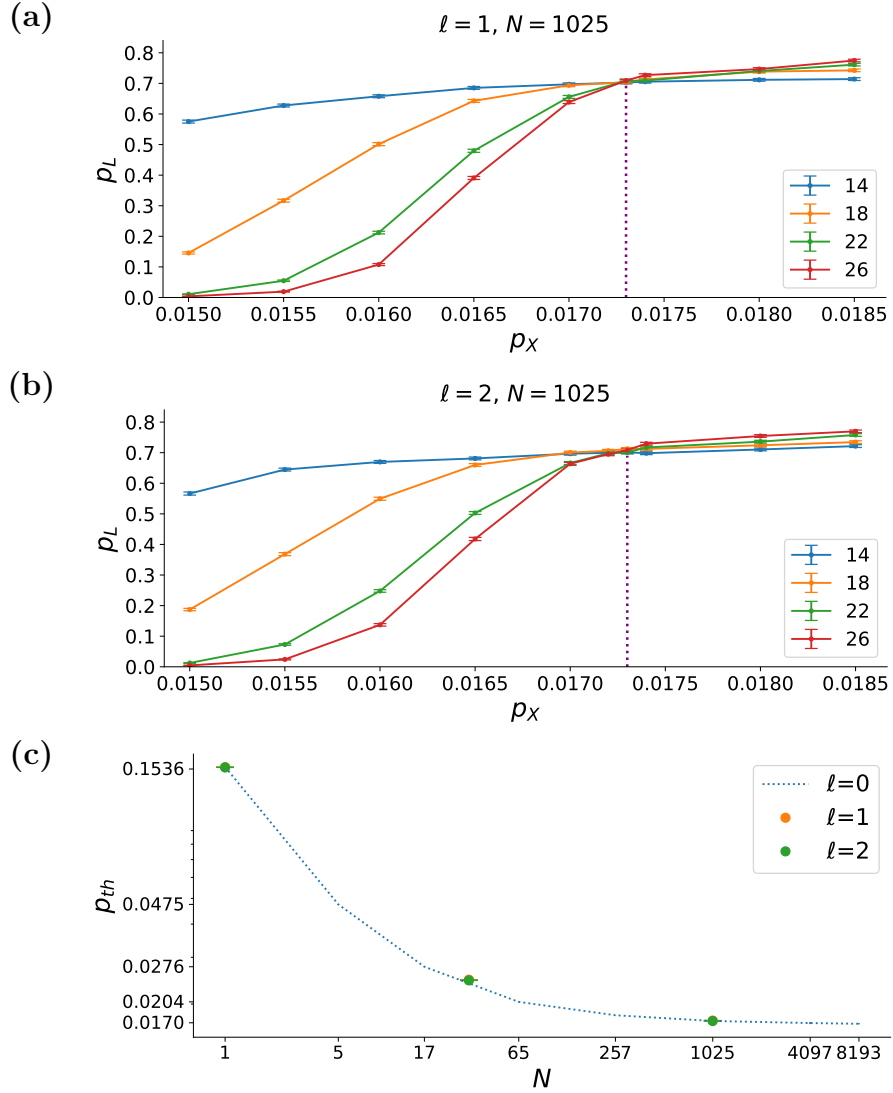


Figure 7: (a)-(b) Sweep decoder performance for $N = 1025$ rounds of stabilizer measurements for levels $\ell = 1, 2$ of the FSC. Logical failure rate p_L for different system sizes (legends) is plotted as a function of the physical X error rate $p_X = q$ where q is the measurement error rate. Each data point in the simulations was obtained with 10,000 Monte Carlo runs. (c) Sweep decoder threshold p_{th} vs number of rounds of stabilizer measurements N . The threshold values were obtained using the critical exponents method [33]. Thresholds for $\ell = 0$ (3D surface code) are taken from Ref. [23]. Our results for points $\ell = 0, N = 1, 33, 1025$ were consistent with the curve..

The threshold obtained in the limit $N \rightarrow \infty$ is the sustainable threshold. Since the $N = 1025$ thresholds for $\ell = 0, 1, 2$ are all around 1.73%, we expect the sustainable threshold for both levels $\ell = 1, 2$ to be 1.7%, which is the result obtained in Ref. [23] for level $\ell = 0$. Moreover, due to this result, we expect this sustainable threshold value to be independent of ℓ ; hence, the sustainable threshold for the asymptotic fractal cube

geometry $FC(3,1)$ is expected to be 1.7%.

Since the X -distance of the code scales with the Hausdorff dimension, D_H of the fractal lattice, as L^{D_H} , we expect the subthreshold failure rates to increase for pure (or biased) Pauli Z noise as D_H is lowered. In our simulations, we only looked at D_H close to 3 due to the limitations on scaling up the system sizes. Hence, such differences in the subthreshold failure rates are not noticeable.

On the other hand, the Z -distance of the code is unaffected under the punching of the holes; however, the effective Z -distance is increased, as there are fewer paths for the point-like syndrome after the smooth boundary holes are punched in. This is expected to have a positive effect on the threshold and the subthreshold failure rate for pure (or biased) Pauli X -noise. Below, we discuss the performance of the MWPM decoder under Pauli X -noise and show an improvement in the threshold error rates with increasing D_H . For a substantial improvement in the subthreshold failure rates, we would need to go to higher D_H .

3 MWPM decoder performance

We now discuss the thresholds of the MWPM decoder used to decode the Z errors. We focus here on the case without measurement errors, which corresponds to the code-capacity error threshold. We prove the existence of the fault-tolerant threshold for the FSC using the MWPM decoder in Sec. I of Supplementary.

As mentioned, the FSC is obtained by punching m -holes in the 3D surface code. For the MWPM decoder, the input decoding graph $\tilde{\mathcal{L}}$, whose each node corresponds to a vertex stabilizer, is a subgraph of the decoding graph \mathcal{L} of the 3D surface code, *i.e.*, $\tilde{\mathcal{L}} \subset \mathcal{L}$ since a subset of stabilizers is removed due to the m -holes. The main difference from the matching of the 3D surface code is that the logical string \bar{Z} and error chain E_1 need to circumvent all the m -holes.

We show the performance of the MWPM decoder, measured in terms of logical failure rate, for levels $\ell = 0, 1, 2$ of the FSC in Fig. 8. We apply the critical exponents method [33] to the logical failure rate data to obtain the code-capacity thresholds of the MWPM decoder for each level ℓ of the FSC defined on $FC(3,1)$ with Hausdorff dimension $D_H = 2.966$ as follows,

ℓ	0	1	2
MWPM threshold	2.886(4)%	2.931(4)%	2.947(5)%

Interestingly, the threshold p_{th} increases in the case of the FSC compared to the 3D surface code case ($\ell=0$). Based on the trend at $\ell=0, 1, 2$, we expect the threshold to be lower bounded by 2.95%. This increase, relative to $\ell = 0$ is due to the fact that one can upper bound the logical error rate to be proportional to the number of self-avoiding non-contractible cycles N_{SAP} (see Sec. I in Supplementary). The input decoding graph for the FSC, $\tilde{\mathcal{L}}$ has strictly lower N_{SAP} than that for the 3D surface code, \mathcal{L} since $\tilde{\mathcal{L}} \subset \mathcal{L}$.

The threshold of FSC can hence be proven to be strictly higher than the 3D surface code. Moreover, the fractal decoding graph is asymptotically approaching a 2D graph in the limit of $D_H = 2 + \epsilon$. The code-capacity threshold of FSC is approaching that of the 2D surface code, *i.e.*, $\lim_{\epsilon \rightarrow 0} p_{\text{th}}^{(2+\epsilon)} \approx 10.31\%$ and the phenomenological fault-tolerant threshold approaches 2.9% [4] (see Sec. I in Supplementary).

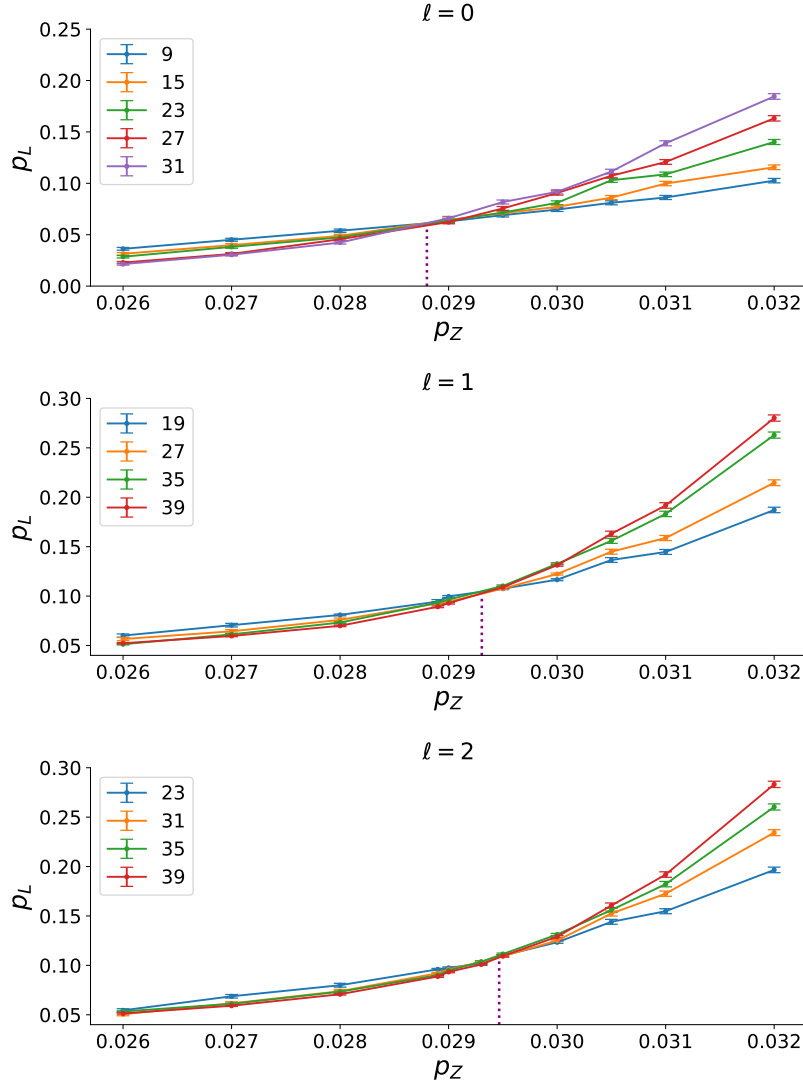


Figure 8: Code capacity performance of the MWPM decoder for pure Z noise for different *levels* of FSC. Logical failure rate p_L is plotted as a function of the physical Z error rate p_Z . The data points close to the threshold of the MWPM decoder were obtained with 100,000 Monte Carlo runs. Other points used 20,000 Monte Carlo runs..

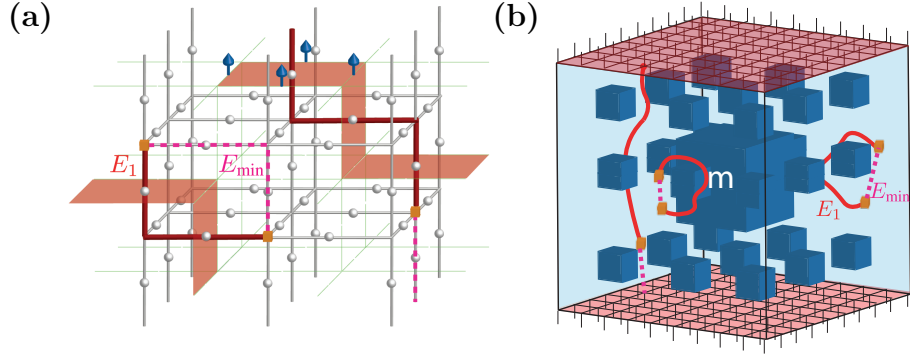


Figure 9: (a) Illustration of the RPGM. The gauge spins (blue) live on the edges of the dual lattice \mathcal{L}^* . Bold red edges represent the error chain E_1 (flux tubes) on the original lattice \mathcal{L} which penetrates the “wrong-sign” plaquettes (red) on \mathcal{L}^* . The endpoints of E_1 are monopoles (orange cube). The MWPM algorithm finds the recovery chain E_{\min} (pink dashed). A logical error occurs since $E_1 + E_{\min}$ contains a nontrivial relative cycle. (b) Illustration of the MWPM decoder on the fractal lattice $\tilde{\mathcal{L}}$, where all flux tubes circumvent the m -holes..

4 Phase transition on a fractal

Quite interestingly, the code-capacity threshold of the MWPM decoder for the FSC can be mapped to the zero-temperature phase transition in a random-plaquette gauge model (RPGM) on a fractal lattice, using the statistical-mechanical mapping [4, 33, 34] of error correction in stabilizer codes. The disordered stat-mech Hamiltonian of the fractal RPGM can be written as:

$$H = - \sum_{f^*} \tau_{f^*} U_{f^*}, \quad \text{with} \quad U_{f^*} = \prod_{e^* \in f^*} \sigma_{e^*}. \quad (5)$$

Here, U_{f^*} is the \mathbb{Z}_2 -valued “gauge flux” penetrating the plaquette (face) f^* on the dual fractal lattice $\tilde{\mathcal{L}}^*$, whose value is determined by all the gauge Ising spins $\sigma_{e^*} = \pm 1$ living on the links (edges) belonging to the plaquette f^* , as illustrated in Fig. 9(a). Each Ising spin σ_{e^*} in the fractal RPGM is associated with a stabilizer on the edge \tilde{e}^* in the code. Hence, the plaquette terms in Eq. (5) are 4-body when in the bulk, while they can be 3-body or 2-body when they share an edge(s) with an m -boundary or two m -boundaries respectively. The random coupling strength $\tau_{f^*} = \pm 1$ on the plaquette f^* represents the quenched disorder in this model. Its sign is determined by the Z error on the qubit of the fractal code: with probability p_Z an error occurs corresponding to the “wrong-sign” choice $\tau_{f^*} = -1$ (favoring nontrivial flux -1) as shown in the highlighted plaquette (red) in Fig. 9(a); with probability $1 - p_Z$ no error occurs corresponding to the “right-sign” choice $\tau_{f^*} = +1$ (favoring trivial flux $+1$). The collection of “wrong-sign” plaquettes constitute flux tubes. Note that the flux tube can terminate on the e -boundary since the dual lattice $\tilde{\mathcal{L}}^*$ contains plaquettes parallel to the e -boundary to allow the flux to penetrate through. At the same time, it is impossible for the flux tube

to terminate on the m -boundary and hence m -holes, since no plaquette parallel to the m -boundary exists on $\tilde{\mathcal{L}}^*$ [see in Fig. 9(a)].

The flux tubes on $\tilde{\mathcal{L}}^*$ correspond to the error chain E_1 (1-chain) living on the edges of the original lattice $\tilde{\mathcal{L}}$, whose endpoints $S_0 = \partial E_1$ are \mathbb{Z}_2 monopoles living on the cube \mathbf{c}^* in $\tilde{\mathcal{L}}^*$ (vertex \mathbf{v} in $\tilde{\mathcal{L}}$), corresponding to the syndromes (e -excitations) in the FSC. The MWPM decoder [35] finds a recovery 1-chain E_{\min} on $\tilde{\mathcal{L}}$ ending at the same set of monopoles (syndromes), *i.e.*, $\partial E_{\min} = S_0$, such that E_{\min} has the minimal length. This is equivalent to finding E_{\min} such that the sum of disjoint closed flux tubes $E_1 + E_{\min}$ has minimal energy at zero temperature. On the other hand, the optimal (maximal-likelihood) decoder effectively finds the recovery chain E_{\min} with minimal free energy at finite temperature along the “Nishimori line” [33] and corresponds to an optimal threshold p_{th}^* . In either case, the recovery succeeds if E_{\min} and E_1 belongs to the same homology class, *i.e.*, $E_1 + E_{\min} = \partial F$, where F is a collection of faces on $\tilde{\mathcal{L}}$, and fails if the closed flux tubes $E_1 + E_{\min}$ contains a homologically nontrivial relative cycle connecting the upper and lower e -boundaries. In the Higgs (ordered) phase where the quenched disorder strength is low ($p_Z < p_{\text{th}}^*$), the failure probability is zero in the thermodynamic limit ($L \rightarrow \infty$) since the free energy cost of such a non-contractible relative cycle diverges linearly with cycle length and become unfavorable. In contrast, in the confined (disordered) phase ($p_Z > p_{\text{th}}^*$), the failure probability approaches one in the thermodynamic limit. The error threshold p_{th} of the MWPM decoder corresponds to the zero-temperature phase transition point between the Higgs phase and a gauge glass phase, which provides a lower bound of the confinement-Higgs transition occurring exactly at the optimal threshold p_{th}^* along the “Nishimori line” [33]. Note that the main difference from the case of the RPGM on a cubic lattice is that the flux tubes need to circumvent the m -holes [shown in Fig. 8(b)]. Interestingly, the phase transition is tunable via the fractal (Hausdorff) dimension D_H .

5 Discussion

In this work, we proved that there exist decoders with nonzero thresholds for fractal surface codes (FSCs) on lattices with Hausdorff dimensions $D_H = 2 + \epsilon$. We noted that even in fractal dimensions, there exists a local decoder for the string-like syndromes. We also proved that the fault-tolerant MWPM threshold for the FSC is strictly greater than the same in the 3D surface code. Moreover, for a particular FSC with Hausdorff dimension $D_H \approx 2.966$, we demonstrated sweep decoder thresholds for bit-flip noise that are roughly the same as those for the regular 3D surface code. For the same FSC, we demonstrated code capacity MWPM threshold in the presence of phase-flip noise that is enhanced in comparison to the regular 3D surface code due to the suppression of the number of non-contractible cycles.

The MWPM threshold provides a lower bound on the confinement-Higgs transition of the RPGM on a fractal lattice. In future work, one can continue studying the optimal threshold of a maximum-likelihood decoder and the exact value of the confinement-

Higgs transition along the “Nishimori line” [4, 33].

In this work, we only studied the performance of the FSC as a quantum memory. Our work demonstrating thresholds on par with the 3D surface code motivates studies of the FSC for quantum computation. In upcoming work, we address implementation of the non-Clifford CCZ gate in a single-shot manner in the FSC, which implies that the space-time overhead scales like $\mathcal{O}(d^{2+\epsilon})$ ⁴, which would be a fundamental space-time improvement on techniques such as those from Refs. [36, 37] which use 2D lattices to simulate the action of 3D topological codes for computation.

Lastly, our work inspires further studies of quantum codes on fractal lattices embedded in three dimensions. For instance, the subsystem surface code [38] and the gauge color code [39], which gauge-fix to the 3D surface code and 3D color code, respectively, have properties of confinement and single-shot error correction, could be studied on fractal lattices. We leave this to a forthcoming work.

Acknowledgements.— We thank John Smolin and Andrew Cross for the valuable discussions and the help with the large-scale cluster simulations. AD thanks Michael Vasmer for valuable discussions and for sharing the sweep decoder data for the 3D surface code. AD thanks the IBM internship program for support, during which a part of this work was completed. AD is also supported by the Simons Foundation through the collaboration on Ultra-Quantum Matter (651438, AD) and by the Institute for Quantum Information and Matter, an NSF Physics Frontiers Center (PHY-1733907). TJO and GZ are supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, and Co-design Center for Quantum Advantage (C2QA) under contract number DE-SC0012704. The code used for the simulations is publicly available at [GitHub](#).

Author contributions.— All authors instigated the study and developed the decoding protocols. A.D. implemented the modified Sweep Decoder, while A.D. and G.Z. implemented the MWPM decoder. T.J., G.Z., and A.D. developed the proofs for the decoders. All authors contributed to the scientific discussion, running simulations, and writing of the manuscript.

Corresponding authors.— Correspondence to any author.

References

- [1] Guanyu Zhu, Tomas Jochym-O’Connor, and Arpit Dua. “Topological order, quantum codes and quantum computation on fractal geometries” (2021). [arXiv:2108.00018](#).
- [2] S. B. Bravyi and A. Yu. Kitaev. “Quantum codes on a lattice with boundary” (1998). [arXiv:quant-ph/9811052](#).
- [3] Alexei Y. Kitaev. “Fault-tolerant quantum computation by anyons”. *Annals of Physics* **303**, 2–30 (2003).

⁴Manuscript in preparation

- [4] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. “Topological quantum memory”. *Journal of Mathematical Physics* **43**, 4452–4505 (2002).
- [5] H. Bombin and M. A. Martin-Delgado. “Topological quantum distillation”. *Physical Review Letters* **97** (2006).
- [6] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. “Surface codes: Towards practical large-scale quantum computation”. *Physical Review A* **86** (2012).
- [7] Sergey Bravyi and Robert König. “Classification of topologically protected gates for local stabilizer codes”. *Physical Review Letters* **110** (2013).
- [8] Tomas Jochym-O’Connor, Aleksander Kubica, and Theodore J. Yoder. “Disjointness of stabilizer codes and limitations on fault-tolerant logical gates”. *Phys. Rev. X* **8**, 021047 (2018).
- [9] Sergey Bravyi and Alexei Kitaev. “Universal quantum computation with ideal clifford gates and noisy ancillas”. *Phys. Rev. A* **71**, 022316 (2005).
- [10] Daniel Litinski. “A game of surface codes: Large-scale quantum computing with lattice surgery”. *Quantum* **3**, 128 (2019).
- [11] Michael A. Levin and Xiao-Gang Wen. “String-net condensation: A physical mechanism for topological phases”. *Phys. Rev. B* **71**, 045110 (2005).
- [12] Robert Koenig, Greg Kuperberg, and Ben W. Reichardt. “Quantum computation with turaev–viro codes”. *Annals of Physics* **325**, 2707–2749 (2010).
- [13] Alexis Schotte, Guanyu Zhu, Lander Burgelman, and Frank Verstraete. “Quantum error correction thresholds for the universal fibonacci turaev-viro code”. *Phys. Rev. X* **12**, 021012 (2022).
- [14] Guanyu Zhu, Ali Lavasani, and Maissam Barkeshli. “Universal logical gates on topologically encoded qubits via constant-depth unitary circuits”. *Phys. Rev. Lett.* **125**, 050502 (2020).
- [15] Ali Lavasani, Guanyu Zhu, and Maissam Barkeshli. “Universal logical gates with constant overhead: instantaneous dehn twists for hyperbolic quantum codes”. *Quantum* **3**, 180 (2019).
- [16] Guanyu Zhu, Ali Lavasani, and Maissam Barkeshli. “Instantaneous braids and dehn twists in topologically ordered states”. *Phys. Rev. B* **102**, 075105 (2020).
- [17] Guanyu Zhu, Mohammad Hafezi, and Maissam Barkeshli. “Quantum origami: Transversal gates for quantum computation and measurement of topological order”. *Phys. Rev. Research* **2**, 013285 (2020).
- [18] Aleksander Kubica, Beni Yoshida, and Fernando Pastawski. “Unfolding the color code”. *New Journal of Physics* **17**, 083026 (2015).
- [19] Michael Vasmer and Dan E. Browne. “Three-dimensional surface codes: Transversal gates and fault-tolerant architectures”. *Physical Review A* **100**, 012312 (2019).
- [20] Héctor Bombín. “Gauge color codes: optimal transversal gates and gauge fixing in topological stabilizer codes”. *New J. Phys.* **17**, 083002 (2015).
- [21] Héctor Bombín. “Single-shot fault-tolerant quantum error correction”. *Phys. Rev. X* **5**, 031043 (2015).

- [22] Aleksander Kubica and John Preskill. “Cellular-automaton decoders with provable thresholds for topological codes”. *Phys. Rev. Lett.* **123**, 020501 (2019).
- [23] Michael Vasmer, Dan E. Browne, and Aleksander Kubica. “Cellular automaton decoders for topological quantum codes with noisy measurements and beyond” (2020). [arXiv:2004.07247](#).
- [24] Benjamin J. Brown, Daniel Loss, Jiannis K. Pachos, Chris N. Self, and James R. Wootton. “Quantum memories at finite temperature”. *Rev. Mod. Phys.* **88**, 045005 (2016).
- [25] Austin G. Fowler, Adam C. Whiteside, and Lloyd C. L. Hollenberg. “Towards practical classical processing for the surface code”. *Physical Review Letters* **108** (2012).
- [26] Fernando Pastawski, Lucas Clemente, and Juan Ignacio Cirac. “Quantum memories based on engineered dissipation”. *Phys. Rev. A* **83**, 012304 (2011).
- [27] Justin L. Mallek, Donna-Ruth W. Yost, Danna Rosenberg, Jonilyn L. Yoder, Gregory Calusine, Matt Cook, Rabindra Das, Alexandra Day, Evan Golden, David K. Kim, Jeffery Knecht, Bethany M. Niedzielski, Mollie Schwartz, Arjan Sevi, Corey Stull, Wayne Woods, Andrew J. Kerman, and William D. Oliver. “Fabrication of superconducting through-silicon vias” (2021). [arXiv:2103.08536](#).
- [28] D. Rosenberg, D. Kim, R. Das, D. Yost, S. Gustavsson, D. Hover, P. Krantz, A. Melville, L. Racz, G. O. Samach, and et al. “3d integrated superconducting qubits”. *npj Quantum Information* **3** (2017).
- [29] Jerry Chow, Oliver Dial, and Jay Gambetta. “IBM Quantum breaks the 100-qubit processor barrier” (2021).
- [30] Sara Bartolucci, Patrick Birchall, Hector Bombin, Hugo Cable, Chris Dawson, Mercedes Gimeno-Segovia, Eric Johnston, Konrad Kieling, Naomi Nickerson, Mihir Pant, Fernando Pastawski, Terry Rudolph, and Chris Sparrow. “Fusion-based quantum computation” (2021). [arXiv:2101.09310](#).
- [31] Héctor Bombín, Isaac H. Kim, Daniel Litinski, Naomi Nickerson, Mihir Pant, Fernando Pastawski, Sam Roberts, and Terry Rudolph. “Interleaving: Modular architectures for fault-tolerant photonic quantum computing” (2021). [arXiv:2103.08612](#).
- [32] Sergey Bravyi and Jeongwan Haah. “Quantum self-correction in the 3d cubic code model”. *Phys. Rev. Lett.* **111**, 200501 (2013).
- [33] 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**, 31–58 (2003).
- [34] Helmut G. Katzgraber, H. Bombin, and M. A. Martin-Delgado. “Error threshold for color codes and random three-body ising models”. *Phys. Rev. Lett.* **103**, 090501 (2009).
- [35] Jack Edmonds. “Paths, trees, and flowers”. *Canadian Journal of Mathematics* **17**, 449–467 (1965).
- [36] Hector Bombin. “2d quantum computation with 3d topological codes” (2018). [arXiv:1810.09571](#).
- [37] Benjamin J. Brown. “A fault-tolerant non-clifford gate for the surface code in two dimensions”. *Science Advances* **6** (2020).

- [38] Aleksander Kubica and Michael Vasmer. “Single-shot quantum error correction with the three-dimensional subsystem toric code” (2021). [arXiv:2106.02621](#).
- [39] H. Bombin. “Gauge color codes: Optimal transversal gates and gauge fixing in topological stabilizer codes” (2015). [arXiv:1311.0879](#).
- [40] Michael John George Vasmer. “Fault-tolerant quantum computing with three-dimensional surface codes”. PhD thesis. UCL (University College London). (2019).

A Proof of the fault-tolerant threshold of a matching decoder for correcting point-like syndromes

In this section, we prove the existence of the fault-tolerant and code-capacity thresholds of the Z errors in fractal surface codes, for the minimum-weight perfect matching (MWPM) decoder, following the methods from Ref. [4]. In particular, we consider the presence of measurement errors under a phenomenological noise model: the Pauli- Z error rate and measurement error rate are denoted by p and q respectively⁵. We also put bounds on the fault-tolerant and code-capacity thresholds of FSCs and show that they are strictly higher than the counterparts of the 3D surface code. Moreover, we show that in the asymptotic limit of $D_H = 2 + \epsilon$, both types of thresholds approach those of the 2D surface code.

A.1 Fault-tolerant MWPM thresholds

The FSC is defined on a 3D fractal lattice $\tilde{\mathcal{L}}^{(D_H)}$ embedded in (being a subset of) a 3D cubic lattice $\mathcal{L}^{(3)}$, *i.e.*, $\tilde{\mathcal{L}}^{(D_H)} \subset \mathcal{L}^{(3)}$, with m -holes being removed from $\mathcal{L}^{(3)}$. Here, D_H represents the Hausdorff dimension of the fractal (space) lattice, and the tilde symbols will always indicate the fractal case from here on. In order to correct the measurement noise, we need to perform d rounds [4] of error correction, where d is the code distance. The corresponding space-time code is defined on the space-time lattice $\tilde{\mathcal{L}}^{(D_H)} \times \mathbb{I}_t$ embedded in a 4D hypercubic lattice $\mathcal{L}^{(4)} \equiv \mathcal{L}^{(3)} \times \mathbb{I}_t$. Here, \mathbb{I}_t represents a 1D lattice along the time direction with d edges and $d + 1$ vertices.

On the 4D space-time lattice $\tilde{\mathcal{L}}^{(D_H)} \times \mathbb{I}_t$, a qubit error event occurs on a space edge $l_{\text{s.p.}}$ at a certain time step t . These space edges are just edges belonging to the space lattice $\tilde{\mathcal{L}}^{(D_H)}$. On the other hand, a syndrome (X -stabilizer measurement) is located at a vertex (v, t) on the space-time lattice, where v corresponds to the vertex label on the space lattice \mathcal{L}_{FSC} and t labels the time step. This syndrome is hence denoted by $s(v, t) \in \{0, 1\}$. Now we define the modified syndrome at time t to be the difference between syndromes at time $t+1$ and time t on the same vertex v on the space lattice, *i.e.*, $s'(v, t) = s(v, t+1) - s(v, t)$. For the special case of $t = d$, we assign $s'(v, d) = s(v, d)$. Therefore, a measurement error at time t and vertex v of the space lattice corresponds to a time edge l_{T} connecting the space-time vertex $(v, t+1)$ and (v, t) . For example, if a single measurement error occur on this time edge l_{T} , both its neighboring syndromes $s'(v, t)$ and $s'(v, t+1)$ will be highlighted, *i.e.*, $s'(v, t) = s'(v, t+1) = 1$.

As we see, both the qubit and measurement errors occur on the edges (1-cells) of the 4D space-time lattice. In the absence of measurement errors ($q = 0$), qubit errors just reside on the edges of the 3D space lattice. Therefore, we can describe generic errors by an error chain E , which mathematically corresponds to a 1-chain of the cell complex,

⁵We suppress the subscript Z of the notation p_Z used for the Z -error rate in the main text

i.e., the 4D or 3D lattice⁶. The error chain E is characterized by a function $n_E(e)$ that takes an edge e to the \mathbb{Z}_2 -coefficient $n_E(e) \in \{0, 1\}$, where $n_E(e) = 1$ corresponds to the edge being occupied by the error chain E . For simplicity, we just consider the isotropic case that qubit and measurement error rates are the same, *i.e.*, $p = q$. The probability that error chain E occurs is

$$\begin{aligned} \Pr(E) &= \prod_e (1-p)^{1-n_E} \\ &= \left[\prod_e (1-p) \right] \prod_e \left(\frac{p}{1-p} \right)^{n_E(e)}. \end{aligned} \quad (6)$$

The boundary of the error chain E is the collection of highlighted modified syndromes on the vertices (0-cells) denoted by S' , *i.e.*, $S' = \partial E$, which is a 0-chain of the cell complex.

Given the measured syndrome information S' , the MWPM decoder needs to guess a recovery chain E_{\min} which is in the same homology class as the actual error chain E . First, the recovery chain E_{\min} should also have the highlighted syndromes S' as its boundary, *i.e.*,

$$\partial E_{\min} = S' = \partial E. \quad (7)$$

Moreover, the recovery succeeds if E_{\min} is in the same homology class as E , which means the following condition needs to be satisfied:

$$E + E_{\min} = \partial F, \quad (8)$$

where the 1-chain $E + E_{\min}$ is a cycle, *i.e.*, with no boundary: $\partial(E + E_{\min}) = 0$. In addition, this 1-cycle $E + E_{\min}$ also needs to be the boundary of a collection of faces F (2-cells), *i.e.*, ∂F , and is hence contractible, *i.e.*, homologically trivial. The recovery fails if $E + E_{\min}$ contains homologically non-trivial cycles which wrap around a 3-torus or connect two different e -boundaries on $\tilde{\mathcal{L}}^{(D_H)}$. From now on, we focus on the case of a 3-torus, *i.e.*, with periodic boundary condition. The threshold with external e -boundaries is expected to be similar.

Apart from an overall normalization, the error chain E occurs with probability $(\frac{p}{1-p})^{|E|}$ according to Eq. (6), where $|E|$ denotes the total number of edges on the error chain E , *i.e.*, the chain length. The MWPM decoder aims to find E_{\min} that maximizes this probability, which is equivalent to minimizing $|E| \log\left(\frac{1-p}{p}\right)$ and effectively the chain length $|E|$ for fixed p .

Now we consider bounding the likelihood of homologically non-trivial cycles being contained in $E + E_{\min}$, which corresponds to the logical failure rate of the MWPM decoder.

We consider a particular cycle C on the space-time lattice $\tilde{\mathcal{L}}^{(D_H)} \times \mathbb{I}_t$ with $|C| \equiv l$ edges. The actual error chain E contains $|E|$ edges, and the estimated recovery chain

⁶From now on, we suppress the subscript of the error chain E_1 used in the main text for clarity.

E_{\min} from the MWPM algorithm contains $|E_{\min}|$ edges. We then ask the probability that the cycle C is contained in $E + E_{\min}$.

First, we can get the following inequality for the chain length:

$$|E| + |E_{\min}| \geq |E + E_{\min}| \geq |C| \equiv l. \quad (9)$$

The first “ \geq ” is due to the fact that some of the edges of E and E_{\min} can overlap and cancel at their binary (\mathbb{Z}_2) sum $E + E_{\min}$, while the equality holds when there is no overlap between E and E_{\min} , *i.e.*, $\text{supp}(E) \cap \text{supp}(E_{\min}) = \emptyset$, where $\text{supp}(E)$ denotes the support of the error chain E , *i.e.*, the set of edges $\{l\}$ with $n_E(l) = 1$. The second “ \geq ” simply attributes to the fact that C is a subset of $E + E_{\min}$. Therefore, we can get the following inequality between the chain probability (up to a normalization constant) of E , E_{\min} and C :

$$\left(\frac{p}{1-p}\right)^{|E_{\min}|} \left(\frac{p}{1-p}\right)^{|E|} \leq \left(\frac{p}{1-p}\right)^l. \quad (10)$$

Moreover, since we have taken E_{\min} to be the minimal-length chain with the boundary being the highlighted syndrome set S' , we have $|E_{\min}| \leq |E|$ and hence

$$\left(\frac{p}{1-p}\right)^{|E|} \leq \left(\frac{p}{1-p}\right)^{|E_{\min}|}. \quad (11)$$

The combination of Eq. (10) and Eq. (11) leads to the following inequality:

$$\left(\frac{p}{1-p}\right)^{|E|} \leq \left[\left(\frac{p}{1-p}\right)^l\right]^{\frac{1}{2}}. \quad (12)$$

Now we consider the probability $P(l)$ that a particular cycle with l edges is contained in $E + E_{\min}$. There are in total 2^l ways to distribute errors (edges contained in E) at locations on the specified chain since each edge either has the error or not. For specified error locations, the probability for these errors to occur is

$$p^{|E|}(1-p)^{l-|E_C|} = (1-p)^l \left(\frac{p}{1-p}\right)^{|E_C|}. \quad (13)$$

where E_C is the part of E that overlaps with C . Therefore, we get the following probability

$$\begin{aligned} \text{Pr}(l) &= 2^l (1-p)^l \left(\frac{p}{1-p}\right)^{|E_C|} \\ &\leq 2^l (1-p)^l \left(\frac{p}{1-p}\right)^{|E|} \leq 2^l p^{\frac{l}{2}} (1-p)^{\frac{l}{2}}, \end{aligned} \quad (14)$$

where the inequality follows from Eq. (12).

Next, we can bound the probability of any cycle with l edges contained in the chain $E + E_{\min}$ via path counting. The cycle C can be considered as a walk on a lattice which begins and ends at a randomly chosen point on the cycle. We need to estimate the likelihood that the closed walk is homologically nontrivial (non-contractible). The walks corresponds to connected chain of errors and visit any given edge at most once. It will be convenient to further restrict the walks to be self-avoiding walks (SAWs) which visit any given vertex at most once, with the exception of the starting/ending point which is revisited. Given any homologically non-trivial closed walk, one can obtain a closed SAW (self-avoiding polygon: SAP) by eliminating some homologically trivial cycles from the walk, which does not change the homology of the cycle and hence the presence or absence of the logical error.

In order to estimate the logical error rate, we consider SAPs lying between two time slices separated by time steps T , and assume $T = O(d)$, where $d = L$ is the code distance which also equals the linear system size in this code family. We denote the number of SAPs with H space (horizontal) edges and V time (vertical) edges by $N_{\text{SAP}}(H, V)$, and one can express the total number of edges as the sum of the number of the two types of edges: $|C| = H + V$. A self-avoiding random walk can start at any of the $d^{n-1}T$ sites on the n -dimensional space-time lattice. In the case of SAP, the starting point can be chosen to be any of the points on the SAP. The probability that $E + E_{\min}$ contains any SAP with H space edges and V time edges obeys the following inequality:

$$\text{Pr}_{\text{SAP}}(H, V) \leq d^{n-1}T N_{\text{SAP}}(H, V) 2^{H+V} p^{\frac{H+V}{2}} (1-p)^{\frac{H+V}{2}}. \quad (15)$$

The minimal homologically nontrivial cycle needs to contain at least d space (horizontal) edges, *i.e.*, $H \geq d$. Therefore, we can bound the logical failure rate as

$$\begin{aligned} \text{Pr}_{\text{fail}} &\leq \sum_V \sum_{H \geq d} \text{Pr}_{\text{SAP}}(H, V) \\ &\leq d^{n-1}T \sum_V \sum_{H \geq d} N_{\text{SAP}}(H, V) [4p(1-p)]^{\frac{H+V}{2}}. \end{aligned} \quad (16)$$

Now we aim to obtain a bound for the fault-tolerant threshold. The number of SAPs does not distinguish the difference between space (horizontal) and time (vertical) edges, and we hence have $N_{\text{SAP}}(H, V) \equiv N_{\text{SAP}}^{(n)}(l)$, where n denotes the dimension of the space-time lattice and l denotes the total number of edges on the SAP. For a (hyper)cubic lattice in n dimensions, the first step of the SAP can choose any of the $2n$ directions, while the subsequent steps will have at most $2n - 1$ directions due to self avoidance. Therefore, one can get the following naive bound

$$N_{\text{SAP}}^{(n)}(l) \leq 2n(2n - 1)^{l-1}. \quad (17)$$

However, there exist tighter bounds, found using results in self-avoiding polygons for

the (hyper)cubic lattice in $n = 2$, $n = 3$ and $n = 4$ [4]:

$$N_{\text{SAP}}^{(2)}(l) \leq P_2(l)(\mu_2)^l, \quad \mu_2 \approx 2.638, \quad (18)$$

$$N_{\text{SAP}}^{(3)}(l) \leq P_3(l)(\mu_3)^l, \quad \mu_3 \approx 4.684, \quad (19)$$

$$N_{\text{SAP}}^{(4)}(l) \leq P_4(l)(\mu_4)^l, \quad \mu_4 \approx 6.77, \quad (20)$$

where $P_2(l)$, $P_3(l)$ and $P_4(l)$ are polynomials.

In the fault-tolerant case, we first consider the 4D hypercubic space-time lattice $\mathcal{L}^{(4)} \equiv \mathcal{L}^{(3)} \times l_t$, and we can hence substitute Eq. (20) into Eq. (16) and obtain

$$\text{Pr}_{\text{fail}}^{(3+1)} \leq d^3 T \sum_V \sum_{H \geq d} P_4(H+V) [4\mu_4^2 p(1-p)]^{\frac{H+V}{2}}. \quad (21)$$

Using $H+V \geq d$ and imposing the following condition

$$p(1-p) < (4\mu_4^2)^{-1}, \quad (22)$$

we obtain the following inequality

$$\begin{aligned} \text{Pr}_{\text{fail}}^{(3+1)} &\leq d^3 T \sum_V \sum_{H \geq d} P_4(H+V) [4\mu_4^2 p(1-p)]^{d/2} \\ &< Q_4(d, T) [4\mu_4^2 p(1-p)]^{d/2}, \end{aligned} \quad (23)$$

where $Q_4(d, T)$ is some polynomial of d and T . The second inequality in Eq. (23) comes from the fact that there are in total $2d^3 T$ space edges and $d^3 T$ time edges in the 4D hypercubic space-time lattice, so there are at most $2d^6 T$ terms in the sum in the first line of Eq. (23) which is absorbed into $Q_4(d, T)$. Therefore, as long as Eq. (22) is satisfied, the logical failure rate Pr_{fail} decays exponentially with the code distance d and hence approaches 0 in the thermodynamic limit $d \rightarrow \infty$. One can then obtain an analytic lower bound of the fault-tolerant threshold of the usual 3D surface code based on Eq. (22) [4]:

$$p_{\text{th}}^{(3+1)} > 0.00548. \quad (24)$$

Now we switch to the case of the fractal surface code defined on the space-time lattice $\tilde{\mathcal{L}}^{(D_H)} \times l_t$ which is embedded in the 4D hypercubic space-time lattice $\mathcal{L}^{(4)} \equiv \mathcal{L}^{(3)} \times l_t$. The only difference from the hypercubic case is that we need to replace $N_{\text{SAP}}^{(4)}(l)$ in Eq. (16) with $\tilde{N}_{\text{SAP}}^{(D_H+1)}(l)$, which represents the total number of self-avoiding polygon on the fractal space-time lattice with Hausdorff dimension $D_H + 1$. Since the fractal space-time lattice is just a subset of the 4D hypercubic lattice, *i.e.*, $\tilde{\mathcal{L}}^{(D_H)} \times l_t \subset \mathcal{L}^{(4)}$ with edges in the hole regions being removed, some SAPs present in the hypercubic lattice is hence removed in the fractal lattice case. Therefore, the number of SAPs in the fractal space-time lattice must be strictly smaller than the SAPs in the 4D hypercubic lattice, *i.e.*,

$$\tilde{N}_{\text{SAP}}^{(D_H+1)}(l) < N_{\text{SAP}}^{(4)}(l). \quad (25)$$

Based on Eq. (25) and the bound in Eq. (16), one gets the following bound:

$$\widetilde{\text{Pr}}_{\text{fail}}^{(D_H+1)} < \text{Pr}_{\text{fail}}^{(3+1)}. \quad (26)$$

where $\widetilde{\text{Pr}}_{\text{fail}}^{(D_H+1)}$ and $\text{Pr}_{\text{fail}}^{(3+1)}$ represent the logical failure rate of the fractal surface code with Hausdorff dimension D_H and the 3D surface code respectively in the fault-tolerance context. Since one has $\text{Pr}_{\text{fail}}^{(3+1)} \rightarrow 0$ in the thermodynamic limit ($d \rightarrow \infty$) and below the fault-tolerant error threshold, *i.e.*, $p < p_{\text{th}}^{(3+1)}$, one will also get $\widetilde{\text{Pr}}_{\text{fail}}^{(D_H+1)} \rightarrow 0$ in the thermodynamic limit due to the bound in Eq. (26). This proves that the fractal surface code is also fault-tolerant below a certain threshold. Furthermore, the fault-tolerant threshold for the fractal surface code must be strictly larger than the 3D surface code, *i.e.*,

$$\widetilde{p}_{\text{th}}^{(D_H+1)} > p^{(3+1)}, \quad (27)$$

since for any $p < p^{(3+1)}$ one always has $\widetilde{\text{Pr}}_{\text{fail}}^{(D_H+1)} \rightarrow 0$ in the thermodynamic limit. When the Hausdorff dimension of the fractal code is asymptotically approaching 3D, the threshold will also approach to that of the 3D surface code, *i.e.*, $\lim_{D_H \rightarrow 3} \widetilde{p}_{\text{th}}^{(D_H+1)} \rightarrow p^{(3+1)}$. According to the derived bound in Eq. (24), we can also analytically bound the error threshold for the fractal surface code, *i.e.*,

$$\boxed{\widetilde{p}_{\text{th}}^{(D_H+1)} > 0.00548 \quad (\text{analytic}).} \quad (28)$$

Meanwhile, existing numerical simulation in the literature gives an estimation of the fault-tolerant threshold of the 3D surface code $p^{(3+1)} \approx 0.0125$ [40] under the phenomenological noise model. Therefore, according to the bound in Eq. (27), we can improve the lower bound for the fractal code under phenomenological noise to be

$$\boxed{\widetilde{p}_{\text{th}}^{(D_H+1)} > 0.0125 \quad (\text{numerical}).} \quad (29)$$

Next, we consider the upper bound and asymptotic limit of the family of fractal surface codes with Hausdorff dimension D_H asymptotically approaching $2 + \epsilon$. We first consider the fault-tolerant threshold of a 2D surface code, which effectively corresponds to a matching problem on a 3D cubic space-time lattice $\mathcal{L}^{(3)} \equiv \mathcal{L}^{(2)} \times \mathbb{I}_t$, where $\mathcal{L}^{(2)}$ represents the 2D space lattice. Similar to the derivation of Eq. (23), we can obtain the following bound:

$$\text{Pr}_{\text{fail}}^{(2+1)} < Q_3(d, T) [4\mu_3^2 p(1-p)]^{d/2}, \quad (30)$$

given the following condition:

$$p(1-p) < (4\mu_3^2)^{-1}. \quad (31)$$

Here, $Q_3(d, T)$ is some polynomial of d and T . This in turn leads to the following analytic bound on the fault-tolerant threshold of the 2D surface code.

$$p_{\text{th}}^{(2+1)} > 0.0114. \quad (32)$$

Now when we gradually reduce the fractal dimension D_H in the family of fractal surface codes defined on the fractal cube geometry, the fractal space-time lattice $\tilde{\mathcal{L}}^{D_H} \times \mathbb{I}_t$ is asymptotically approaching a 3D cubic lattice, *i.e.*,

$$\tilde{\mathcal{L}}^{(2+\epsilon)} \times \mathbb{I}_t \rightarrow \mathcal{L}^{(2)} \times \mathbb{I}_t = \mathcal{L}^{(3)}, \quad (33)$$

which means all the quantitative properties of the fractal lattice is asymptotically approaching those of the cubic lattice, including the scaling of the number of SAP:

$$\lim_{\epsilon \rightarrow 0} \tilde{N}_{\text{SAP}}^{(2+\epsilon+1)}(\ell) \sim N_{\text{SAP}}^{(3)}(\ell) \quad (34)$$

This in turn leads to the following asymptotic threshold:

$$\lim_{\epsilon \rightarrow 0} \tilde{p}_{\text{th}}^{(2+\epsilon+1)} = p_{\text{th}}^{(2+1)}, \quad (35)$$

and the upper bound for the whole family of fractal surface code

$$\tilde{p}_{\text{th}}^{(D_H+1)} < p_{\text{th}}^{(2+1)}. \quad (36)$$

According to Eq. (32) and (34), we can get the following strict bound on the asymptotic threshold:

$$\boxed{\lim_{\epsilon \rightarrow 0} \tilde{p}_{\text{th}}^{(2+\epsilon+1)} > 0.0114 \quad (\text{analytic}).} \quad (37)$$

Meanwhile, existing numerical MWPM simulation gives an estimation of the fault-tolerant threshold of the 2D surface code $p^{(2+1)} \approx 0.029$ [4, 33] under the phenomenological noise model. Therefore, we can get an estimation based on the numerical value as:

$$\boxed{\lim_{\epsilon \rightarrow 0} \tilde{p}_{\text{th}}^{(2+\epsilon+1)} \approx 0.029 \quad (\text{numerical}).} \quad (38)$$

In sum, we have shown that the fault-tolerant threshold of the fractal surface codes under pure Pauli-Z and measurement errors is strictly higher than the 3D surface code, and asymptotically approaching the value of the 2D surface code, which implies a significant practical advantage due to its additional ability of performing logical CCZ gate.

A.2 Code capacity MWPM thresholds

Finally, we also discuss the code-capacity error threshold, *i.e.*, in the absence of measurement error ($q = 0$), which is numerically studied in the main text. In this case, we only consider the space lattice. For the 3D and 2D surface codes, the thresholds correspond to the matching problems on a 3D lattice $\mathcal{L}^{(3)}$ and 2D lattice $\mathcal{L}^{(2)}$ respectively. Similar to the derivation of Eq. (23), we can obtain the following bounds:

$$\Pr_{\text{fail}}^{(3)} < Q_3(d, T)[4\mu_3^2 p(1-p)]^{d/2}, \quad (39)$$

$$\Pr_{\text{fail}}^{(2)} < Q_2(d, T)[4\mu_2^2 p(1-p)]^{d/2}, \quad (40)$$

given the following conditions respectively:

$$p(1-p) < (4\mu_3^2)^{-1}, \quad (41)$$

$$p(1-p) < (4\mu_2^2)^{-1}. \quad (42)$$

We hence get the following analytic bound on the code-capacity error thresholds for 3D and 2D surface codes:

$$p_{\text{th}}^{(3)} > 0.0114, \quad (43)$$

$$p_{\text{th}}^{(2)} > 0.0373. \quad (44)$$

Similar to Eq. (27) and Eq. (36) and the corresponding arguments in the fault-tolerant scenario, we obtain the upper and lower bound of the code-capacity error threshold of the family of fractal codes with Hausdorff dimension $2 < D_H < 3$ in terms of the thresholds of the 2D and 3D surface codes, *i.e.*,

$$p_{\text{th}}^{(3)} < \tilde{p}_{\text{th}}^{(D_H)} < p_{\text{th}}^{(2)}. \quad (45)$$

We hence get the following analytic lower bound:

$$\boxed{\tilde{p}_{\text{th}}^{(D_H)} > 0.0114 \quad (\text{analytic})} \quad (46)$$

and the analytic bound on the asymptotic code-capacity threshold at $D_H = 2 + \epsilon$:

$$\boxed{\lim_{\epsilon \rightarrow 0} \tilde{p}_{\text{th}}^{(2+\epsilon)} > 0.0373 \quad (\text{analytic}).} \quad (47)$$

Meanwhile, existing numerical simulation gives the estimate of the code-capacity threshold of the 3D surface code $p^{(3)} \approx 0.029$ under the phenomenological noise model [33, 40], which is exactly the same as the fault-tolerant threshold of the 2D surface code $p^{(2+1)}$ mentioned above. On the other hand, the numerical estimate of the code-capacity threshold of the 2D surface code is $p^{(2)} \approx 0.1031$ [33]. Therefore, we get the following numerical lower bound of the class of fractal surface codes:

$$\boxed{\tilde{p}_{\text{th}}^{(D_H)} > 0.029 \quad (\text{numerical}),} \quad (48)$$

and the numerical estimation on the asymptotic code-capacity threshold:

$$\boxed{\lim_{\epsilon \rightarrow 0} \tilde{p}_{\text{th}}^{(2+\epsilon)} \approx 0.1031 \quad (\text{numerical}).} \quad (49)$$

Indeed, our numerical simulation on the fractal $FC(3, 1)$ in the main text shows that $\tilde{p}_{\text{th}}^{(D_H=2.966)} > p_{\text{th}}^{(3)} \approx 0.029$.

B Counting argument for topological degeneracy from m -holes

In the main text, we presented a homological argument to show that the m -holes do not encode any logical information. Here, we confirm that via a simple counting argument.

A single $L \times L \times L$ m -hole changes the number of Z stabilizers N_Z to $N'_Z = N_Z - 3L^2(L - 1)$ since there are $3(L - 1)$ planes from which L^2 plaquette operators are removed. The number of X stabilizers changes from N_X to $N'_X = N_X - (L - 1)^3$ because $(L - 1)^3$ operators lose all their qubits inside the hole. The number of relations among the Z and X stabilizers also changes from R_Z and R_X to $R'_Z = R_Z - (L^3 - 1)$ and $R'_X = R_X$ respectively. The $L^3 - 1$ comes from the relations on L^3 cubes that are removed but leaving a single relation coming from plaquette operators on the surface of the m -hole. The number of physical qubits changes from N_q to $N'_q = N_q - 3L(L - 1)^2$ since there are $3L$ planes, each with $(L - 1)^2$ physical qubits, inside the $L \times L \times L$ m -hole. Thus, we have the number of encoded qubits k' as

$$\begin{aligned}
 k' &= N'_q - (N'_S - R') \\
 &= N'_q - (N'_Z + N'_X - R'_Z - R'_X) \\
 &= N_q - 3L(L - 1)^2 \\
 &\quad - (N_Z - 3L^2(L - 1) + N_X - (R_Z - L^3 + 1) - R_X) \\
 &= N_q - (N_S - R) \\
 &= k
 \end{aligned}$$

where $N_S = N_Z + N_X$, $R = R_Z + R_X$ and k is the number of encoded qubits before making the m -hole. The argument generalizes to $L_x \times L_y \times L_z$ m -holes and an arbitrary number of them.

C Sweep decoder threshold plots for rounds $N = 0, 32$

Below, we show the threshold plots for levels $\ell = 1, 2$ of the FSC for the number of rounds of measurements as $N = 1, 33$.

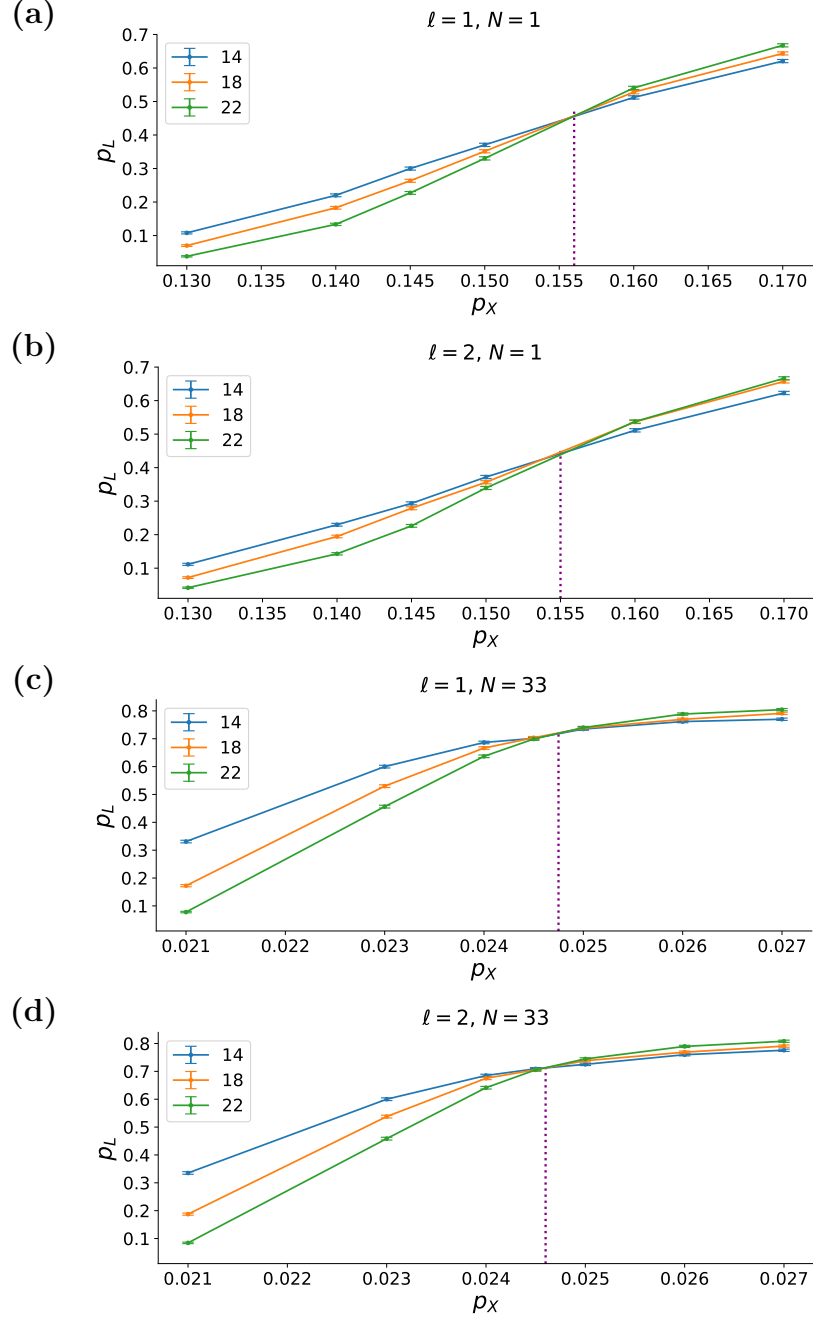


Figure 10: Sweep decoder performance for $N = 0, 32$ rounds of stabilizer measurements for levels $\ell = 1, 2$ of the FSC. Logical failure rate is plotted as a function of the physical error rate p_X . The measurement error rate q is set to be the same as the physical error rate p_X .

References

- [1] Guanyu Zhu, Tomas Jochym-O'Connor, and Arpit Dua. “Topological order, quantum codes and quantum computation on fractal geometries” (2021). [arXiv:2108.00018](#).
- [2] S. B. Bravyi and A. Yu. Kitaev. “Quantum codes on a lattice with boundary” (1998). [arXiv:quant-ph/9811052](#).
- [3] Alexei Y. Kitaev. “Fault-tolerant quantum computation by anyons”. [Annals of Physics](#) **303**, 2–30 (2003).
- [4] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. “Topological quantum memory”. [Journal of Mathematical Physics](#) **43**, 4452–4505 (2002).
- [5] H. Bombin and M. A. Martin-Delgado. “Topological quantum distillation”. [Physical Review Letters](#) **97** (2006).
- [6] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. “Surface codes: Towards practical large-scale quantum computation”. [Physical Review A](#) **86** (2012).
- [7] Sergey Bravyi and Robert König. “Classification of topologically protected gates for local stabilizer codes”. [Physical Review Letters](#) **110** (2013).
- [8] Tomas Jochym-O'Connor, Aleksander Kubica, and Theodore J. Yoder. “Disjointness of stabilizer codes and limitations on fault-tolerant logical gates”. [Phys. Rev. X](#) **8**, 021047 (2018).
- [9] Sergey Bravyi and Alexei Kitaev. “Universal quantum computation with ideal clifford gates and noisy ancillas”. [Phys. Rev. A](#) **71**, 022316 (2005).
- [10] Daniel Litinski. “A game of surface codes: Large-scale quantum computing with lattice surgery”. [Quantum](#) **3**, 128 (2019).
- [11] Michael A. Levin and Xiao-Gang Wen. “String-net condensation: A physical mechanism for topological phases”. [Phys. Rev. B](#) **71**, 045110 (2005).
- [12] Robert Koenig, Greg Kuperberg, and Ben W. Reichardt. “Quantum computation with turaev–viro codes”. [Annals of Physics](#) **325**, 2707–2749 (2010).
- [13] Alexis Schotte, Guanyu Zhu, Lander Burgelman, and Frank Verstraete. “Quantum error correction thresholds for the universal fibonacci turaev-viro code”. [Phys. Rev. X](#) **12**, 021012 (2022).
- [14] Guanyu Zhu, Ali Lavasani, and Maissam Barkeshli. “Universal logical gates on topologically encoded qubits via constant-depth unitary circuits”. [Phys. Rev. Lett.](#) **125**, 050502 (2020).
- [15] Ali Lavasani, Guanyu Zhu, and Maissam Barkeshli. “Universal logical gates with constant overhead: instantaneous dehn twists for hyperbolic quantum codes”. [Quantum](#) **3**, 180 (2019).
- [16] Guanyu Zhu, Ali Lavasani, and Maissam Barkeshli. “Instantaneous braids and dehn twists in topologically ordered states”. [Phys. Rev. B](#) **102**, 075105 (2020).
- [17] Guanyu Zhu, Mohammad Hafezi, and Maissam Barkeshli. “Quantum origami: Transversal gates for quantum computation and measurement of topological order”. [Phys. Rev. Research](#) **2**, 013285 (2020).

- [18] Aleksander Kubica, Beni Yoshida, and Fernando Pastawski. “Unfolding the color code”. *New Journal of Physics* **17**, 083026 (2015).
- [19] Michael Vasmer and Dan E. Browne. “Three-dimensional surface codes: Transversal gates and fault-tolerant architectures”. *Physical Review A* **100**, 012312 (2019).
- [20] Héctor Bombín. “Gauge color codes: optimal transversal gates and gauge fixing in topological stabilizer codes”. *New J. Phys.* **17**, 083002 (2015).
- [21] Héctor Bombín. “Single-shot fault-tolerant quantum error correction”. *Phys. Rev. X* **5**, 031043 (2015).
- [22] Aleksander Kubica and John Preskill. “Cellular-automaton decoders with provable thresholds for topological codes”. *Phys. Rev. Lett.* **123**, 020501 (2019).
- [23] Michael Vasmer, Dan E. Browne, and Aleksander Kubica. “Cellular automaton decoders for topological quantum codes with noisy measurements and beyond” (2020). [arXiv:2004.07247](#).
- [24] Benjamin J. Brown, Daniel Loss, Jiannis K. Pachos, Chris N. Self, and James R. Wootton. “Quantum memories at finite temperature”. *Rev. Mod. Phys.* **88**, 045005 (2016).
- [25] Austin G. Fowler, Adam C. Whiteside, and Lloyd C. L. Hollenberg. “Towards practical classical processing for the surface code”. *Physical Review Letters* **108** (2012).
- [26] Fernando Pastawski, Lucas Clemente, and Juan Ignacio Cirac. “Quantum memories based on engineered dissipation”. *Phys. Rev. A* **83**, 012304 (2011).
- [27] Justin L. Mallek, Donna-Ruth W. Yost, Danna Rosenberg, Jonilyn L. Yoder, Gregory Calusine, Matt Cook, Rabindra Das, Alexandra Day, Evan Golden, David K. Kim, Jeffery Knecht, Bethany M. Niedzielski, Mollie Schwartz, Arjan Sevi, Corey Stull, Wayne Woods, Andrew J. Kerman, and William D. Oliver. “Fabrication of superconducting through-silicon vias” (2021). [arXiv:2103.08536](#).
- [28] D. Rosenberg, D. Kim, R. Das, D. Yost, S. Gustavsson, D. Hover, P. Krantz, A. Melville, L. Racz, G. O. Samach, and et al. “3d integrated superconducting qubits”. *npj Quantum Information* **3** (2017).
- [29] Jerry Chow, Oliver Dial, and Jay Gambetta. “IBM Quantum breaks the 100-qubit processor barrier” (2021).
- [30] Sara Bartolucci, Patrick Birchall, Hector Bombin, Hugo Cable, Chris Dawson, Mercedes Gimeno-Segovia, Eric Johnston, Konrad Kieling, Naomi Nickerson, Mihir Pant, Fernando Pastawski, Terry Rudolph, and Chris Sparrow. “Fusion-based quantum computation” (2021). [arXiv:2101.09310](#).
- [31] Héctor Bombín, Isaac H. Kim, Daniel Litinski, Naomi Nickerson, Mihir Pant, Fernando Pastawski, Sam Roberts, and Terry Rudolph. “Interleaving: Modular architectures for fault-tolerant photonic quantum computing” (2021). [arXiv:2103.08612](#).
- [32] Sergey Bravyi and Jeongwan Haah. “Quantum self-correction in the 3d cubic code model”. *Phys. Rev. Lett.* **111**, 200501 (2013).
- [33] 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**, 31–58 (2003).

- [34] Helmut G. Katzgraber, H. Bombin, and M. A. Martin-Delgado. “Error threshold for color codes and random three-body ising models”. *Phys. Rev. Lett.* **103**, 090501 (2009).
- [35] Jack Edmonds. “Paths, trees, and flowers”. *Canadian Journal of Mathematics* **17**, 449–467 (1965).
- [36] Hector Bombin. “2d quantum computation with 3d topological codes” (2018). [arXiv:1810.09571](#).
- [37] Benjamin J. Brown. “A fault-tolerant non-clifford gate for the surface code in two dimensions”. *Science Advances* **6** (2020).
- [38] Aleksander Kubica and Michael Vasmer. “Single-shot quantum error correction with the three-dimensional subsystem toric code” (2021). [arXiv:2106.02621](#).
- [39] H. Bombin. “Gauge color codes: Optimal transversal gates and gauge fixing in topological stabilizer codes” (2015). [arXiv:1311.0879](#).
- [40] Michael John George Vasmer. “Fault-tolerant quantum computing with three-dimensional surface codes”. PhD thesis. UCL (University College London). (2019).