

Disjointness of Stabilizer Codes and Limitations on Fault-Tolerant Logical Gates

Tomas Jochym-O'Connor,^{1,2} Aleksander Kubica,^{2,3,4} and Theodore J. Yoder^{5,6}

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

²Institute for Quantum Information & Matter, California Institute of Technology, Pasadena, California 91125, USA

³Perimeter Institute for Theoretical Physics, Waterloo, Ontario N2L 2Y5, Canada

⁴Institute for Quantum Computing, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada

⁵Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

⁶IBM T.J. Watson Research Center, Yorktown Heights, New York 10598, USA



(Received 28 October 2017; revised manuscript received 3 March 2018; published 21 May 2018)

Stabilizer codes are among the most successful quantum error-correcting codes, yet they have important limitations on their ability to fault tolerantly compute. Here, we introduce a new quantity, the disjointness of the stabilizer code, which, roughly speaking, is the number of mostly nonoverlapping representations of any given nontrivial logical Pauli operator. The notion of disjointness proves useful in limiting transversal gates on any error-detecting stabilizer code to a finite level of the Clifford hierarchy. For code families, we can similarly restrict logical operators implemented by constant-depth circuits. For instance, we show that it is impossible, with a constant-depth but possibly geometrically nonlocal circuit, to implement a logical non-Clifford gate on the standard two-dimensional surface code.

DOI: [10.1103/PhysRevX.8.021047](https://doi.org/10.1103/PhysRevX.8.021047)

Subject Areas: Quantum Physics, Quantum Information

I. INTRODUCTION

Quantum error-correcting codes form the foundation of scalable quantum computing [1–3]. By construction, quantum codes serve as quantum memories by protecting encoded data from a noisy environment and successfully extending the storage time, at least if the noise is sufficiently small. However, a quantum computer should do more than just store quantum data; It needs to also apply logical operations to the data [4,5]. These operations must therefore be implemented fault tolerantly upon quantum codes.

Generally, operators are fault tolerant if they do not couple too many qubits within a particular code block. This condition is sufficient to limit the spread of errors and also guarantee that, if parts of the circuitry implementing the operator were to fail, not many qubits would be affected. With respect to some partitioning of the code qubits into small, disjoint subsets Q_i , a transversal operator acts on each subset of qubits Q_i independently. For a family of codes with increasing size, a constant-depth logical operator is implementable by a constant (independent of the code size) depth circuit over the subsets Q_i . Transversal and constant-depth circuits are some of the simplest possible

fault-tolerant operators, both theoretically and experimentally, so it is important to understand exactly what logical operators they can implement.

Unfortunately, the set of transversal or, more generally, constant-depth logical operators is inherently limited, with computational universality generally incommensurate with the error-correction capabilities of the code. In particular, there is a no-go theorem from Eastin and Knill, which states that transversal operators on any nontrivial quantum code belong to a finite group and thus cannot be universal [6,7]. Similar no-go theorems limiting logical operators to be in a finite level of the Clifford hierarchy were derived for transversal single-qubit gates and two-qubit diagonal gates on stabilizer codes [8], as well as for constant-depth, local circuits on stabilizer and subsystem topological codes [9,10]. The latter result has an important implication—one cannot achieve a universal gate set with constant-depth local circuits on two-dimensional (2D) topological codes such as those in Refs. [11,12]. We also remark that one can consider more general models beyond stabilizer codes, such as 2D topological quantum field theories, and characterize the set of gates implementable by locality-preserving unitaries [13,14].

Here, we address several related questions regarding transversal and constant-depth logical operators on stabilizer codes using a new quantity called the disjointness of the code. The disjointness, roughly speaking, is the number of mostly nonoverlapping representatives of any given nontrivial logical Pauli operator. We use the disjointness to show that all transversal logical operators on stabilizer

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

codes must be in the Clifford hierarchy, as conjectured by Zeng *et al.* [7]. Moreover, we find explicit upper bounds on the level attainable. In particular, this result rules out the possibility of transversal implementation of operators that can be used to bootstrap up the Clifford hierarchy, such as the Toffoli gate [15]. Importantly, our result, when applied to families of codes of growing size, restricts constant-depth circuits to the Clifford hierarchy, regardless of their geometric locality. For instance, for the 2D surface code on a square lattice of size $l \times O(l)$, we find that even nonlocal constant-depth circuits cannot implement logical non-Clifford operators. Asymmetry of logical operators appears in our bounds as a necessary condition for possessing constant-depth circuits for non-Clifford gates, such as those on 3D color and toric codes [16,17] and on asymmetric 2D Bacon-Shor codes [18].

II. INTUITION

In this section, we sketch out the proof that constant-depth circuits, even with gates that are geometrically nonlocal, cannot implement logical non-Clifford operators on the 2D surface code of size $l \times O(l)$ (see Fig. 1). We use the following two key ideas: (i) There are many non-overlapping representatives for logical Pauli \bar{X} and \bar{Z} operators; (ii) logical operators supported on a correctable region are trivial.

In order to find out what logical gate a unitary U implements, it is sufficient to characterize the action of U on the logical Pauli operators. Let $[A, B] = ABA^\dagger B^\dagger$ represent the group commutator of two unitaries A and B [9]. We know that for any two logical Pauli operators $\bar{P}, \bar{Q} \in \{\bar{X}, \bar{Z}\}$, if the group commutator $[[U, \bar{P}], \bar{Q}]$ is a trivial logical operator, then the unitary U implements a logical Clifford operator [19].

Let us pick a representative p of the logical operator $\bar{P} \in \{\bar{X}, \bar{Z}\}$, such that $|\text{supp}(p)| = O(l)$. We denote by

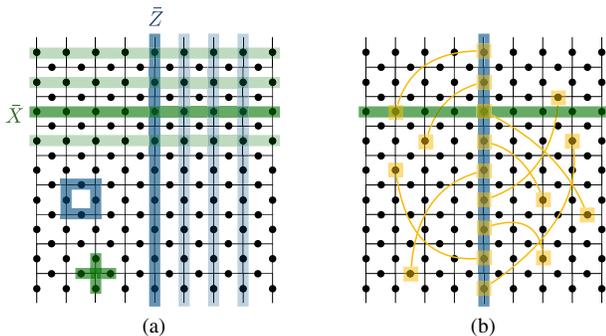


FIG. 1. (a) The 2D surface code has X -vertex and Z -plaquette stabilizer generators and many nonoverlapping representatives of logical Pauli \bar{X} (green) and \bar{Z} (blue) operators. (b) For a given representative of \bar{Z} (blue) and any constant-depth circuit U with possibly geometrically nonlocal gates (depicted in yellow), one can always find a representative of \bar{X} (green), such that $U\bar{Z}U^\dagger$ and \bar{X} overlap only on a constant number of qubits.

$\text{supp}(A)$ the set of qubits an operator A acts on nontrivially (we later generalize this notion). Since we assume that U is constant depth, then $|\text{supp}([U, p])| = O(l)$. Note that a tensor product of Pauli Z operators on qubits along any vertical path on the lattice would implement the logical Pauli \bar{Z} [see Fig. 1(a)]. Similarly, Pauli X operators along any horizontal path implement the logical Pauli \bar{X} . Thus, for any operator $\bar{Q} \in \{\bar{X}, \bar{Z}\}$, we can choose $O(l)$ different, nonoverlapping representatives. Using the pigeonhole principle, we are guaranteed to find a representative q of \bar{Q} , such that it has constant overlap with $[U, p]$ [see Fig. 1(b) for an example]. This, in turn, implies that the operator $[[U, p], q]$ is supported on a constant-size region, $|\text{supp}([U, p], q)| = O(1)$. Since the distance of the code is $O(l)$, the region $\text{supp}([U, p], q)$ is correctable. We conclude that $[[U, p], q]$ can only be a trivial logical operator, and thus U implements a logical Clifford operator.

The property of any stabilizer code that we would like to abstract from the provided example of the 2D surface code is the existence of several disjoint representatives of the same logical Pauli operator. In the following sections, we introduce a notion of disjointness of a stabilizer code, which quantitatively captures that property. We remark that the disjointness of the 2D surface code is $O(l)$ since we can find a set of $O(l)$ nonoverlapping representatives of either \bar{X} or \bar{Z} .

As a final remark, consider \bar{Y} . We argued above that $[U, \bar{X}]$ and $[U, \bar{Z}]$ are logical Pauli operators, which implies that $[U, \bar{Y}] = [U, \bar{X}]\bar{X}[U, \bar{Z}]\bar{Z}$ is a logical Pauli as well. Yet, proving this directly by an argument analogous to the one presented for \bar{X} and \bar{Z} does not work exactly. This is because any two different representatives of \bar{Y} necessarily overlap on at least one qubit, and the argument breaks down since it is impossible to find a set of $O(l)$ disjoint representatives of \bar{Y} . Nevertheless, we can find a set of l representatives of \bar{Y} such that any single qubit is in support of at most two representatives from the set. We call such a set of representatives 2-disjoint. Given this 2-disjoint set, the pigeonhole argument can again be used to find a representative q of \bar{Y} such that $[[U, p], q] = O(1)$, where p is the previously chosen representative of a logical Pauli $\bar{P} \in \{\bar{X}, \bar{Z}\}$ of weight $O(l)$. As we will see in the rest of the paper, considering overlapping representatives of logical Pauli operators and c -disjoint sets proves very useful in constraining fault-tolerant logical gates.

III. PRELIMINARIES

Let us consider systems composed of m -dimensional qudits, $m \geq 2$. The Pauli group on a set of n qudits, denoted \mathcal{P}_n , is generated by the X - and Z -type operators [20]

$$X = \sum_{j=0}^{m-1} |j \oplus 1\rangle\langle j|, \quad Z = \sum_{j=0}^{m-1} \omega_m^j |j\rangle\langle j|, \quad (1)$$

where addition \oplus inside bra-kets is modulo m and $\omega_m = \exp(2\pi i/m)$. Letting \mathcal{U}_n denote the group of n -qudit unitaries, we note that \mathcal{P}_n is a subgroup of \mathcal{U}_n because $X, Z \in \mathcal{U}_n$.

Any Pauli group \mathcal{P} can be used to define a hierarchy of n -qudit unitaries called the Clifford hierarchy [21]. The M th level of this hierarchy is a finite set of unitaries (if the global phases are ignored) recursively defined as

$$C_1(\mathcal{P}) = \mathcal{P}, \quad (2)$$

$$C_M(\mathcal{P}) = \{U \in \mathcal{U}_n : [U, p] \in C_{M-1}(\mathcal{P}), \forall p \in \mathcal{P}\}. \quad (3)$$

The first and second levels of the hierarchy correspond to the Pauli and Clifford groups, respectively.

In this article, we focus our attention on a particularly popular class of quantum codes—stabilizer codes [22]. A stabilizer code is defined by the stabilizer group $\mathcal{S} = \langle s_1, s_2, \dots, s_{n-k} \rangle \subseteq \mathcal{P}_n$, which is generated by $n-k$ mutually commuting Pauli operators. The codespace \mathcal{C} is a subspace of the Hilbert space $\mathcal{H} \simeq (\mathbb{C}^m)^{\otimes n}$ on n qudits, which is the simultaneous (+1)-eigenspace of all stabilizer generators s_i . We denote by $[[n, k]]$ a qudit stabilizer code, which uses n physical qudits to encode k logical ones.

For any stabilizer code, a logical operator is a unitary on the Hilbert space \mathcal{H} that maps states in \mathcal{C} to states in \mathcal{C} . In particular, logical Pauli operators can be found as elements of the normalizer $\mathcal{N}(\mathcal{S})$ of the stabilizer group \mathcal{S} in the Pauli group \mathcal{P}_n . We choose $2k$ generators $\bar{X}_i, \bar{Z}_i \in \mathcal{P}_n$ of the logical Pauli group $\bar{\mathcal{P}}_k$ that commute with all stabilizer generators and also satisfy

$$[\bar{X}_i, \bar{Z}_j] = \omega_m^{-\delta_{ij}} I, \quad (4)$$

$$[\bar{X}_i, \bar{X}_j] = [\bar{Z}_i, \bar{Z}_j] = I. \quad (5)$$

We define \mathcal{L} to be the set of all nontrivial logical Pauli operators as follows:

$$\mathcal{L} = \left\{ \mathcal{S} \prod_{i=1}^k \bar{X}_i^{a_i} \bar{Z}_i^{a_{i+k}} : a \in \{0, 1, \dots, m-1\}^{2k} \setminus \{0\}^{2k} \right\}. \quad (6)$$

We remark that each element $G \in \mathcal{L}$ is a coset of \mathcal{S} in $\mathcal{N}(\mathcal{S})$, although in examples we abuse notation and equate G with the logical Pauli that it corresponds to (e.g., $\bar{X} = \mathcal{S}\bar{X} \in \mathcal{L}$). Also, G contains $|\mathcal{S}| = m^{n-k}$ representatives of the same nontrivial logical operator.

IV. TRANSVERSAL GATES

All manipulations of a quantum code should be fault tolerant, in the sense that they do not spread errors throughout the system in an uncontrollable way. The simplest example of such a manipulation is a transversal logical operator U . Typically, when one says U is

transversal, it means that U is a tensor product of single-qudit unitaries. However, we consider a more general definition of a transversal gate [23] Partition the set of n physical qudits, labeled by integers from $[n] = \{1, 2, \dots, n\}$, into N disjoint, nonempty subsets $Q_i \subseteq [n]$, namely,

$$[n] = Q_1 \cup Q_2 \cup \dots \cup Q_N. \quad (7)$$

Then, we say that an n -qudit unitary U is transversal if it can be decomposed as $U = \bigotimes_{i=1}^N U_i$, where each unitary U_i acts only on qudits in the subset Q_i . The support of U , denoted by $\text{supp}(U) \subseteq [N]$, is the index set of all subsets Q_i , on which U acts nontrivially. The typical notion of transversal gate now simply corresponds to the partition into single qudits, $Q_i = \{i\}$.

We emphasize that for a given code, the set of transversal logical operators can depend on the choice of the qudit partition. In particular, if the partition is not fixed, then one can achieve a universal gate set of transversal operators, as in the following example [24].

Example 1. Consider the $[[105, 1]]$ code, which is a concatenation of the Steane 7-qubit code with the 15-qubit Reed-Muller code. We illustrate this code in Fig. 2 as a 7×15 array of qubits. We consider two qubit partitions: (a) Each Q_i is a subset of 7 qubits from the i th column; (b) each Q_i is a subset of 15 qubits from the i th row. With respect to the first and second partitions, the $[[105, 1]]$ code has, correspondingly, transversal logical $T = \text{diag}(1, e^{2\pi i/8})$ and Hadamard gates. For more details, see Ref. [24].

In contrast, we fix a partition and prove limitations on logical operators with respect to that partition. For instance, in this fixed-partition scenario, Ref. [6] implies that the group of transversal operators is finite and therefore not universal.

Transversal unitaries are a special case of what we call q -local operators of depth h (with respect to the partition $\{Q_i\}$). A unitary U is q -local of depth one, if it is

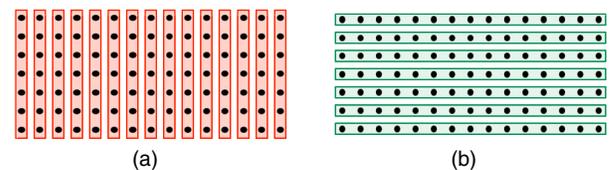


FIG. 2. Two different partitions of the $[[105, 1]]$ qubit stabilizer code, which is a concatenation of the 7-qubit and 15-qubit codes. Each row of the 7×15 array is an instance of the 15-qubit code that encodes one of the qubits of the 7-qubit code. Depending on the partition of the qubits, the code can have either a transversal logical (a) $T = \text{diag}(1, \exp 2\pi i/8)$ or (b) Hadamard gate. Here, a transversal unitary with respect to a given partition is a tensor product of unitaries, each of which is supported only on qubits within one subset of the partition (red or green box).

transversal with respect to a second, ‘‘coarse-grained’’ partition $\{R_j\}$, where each R_j is the union of at most q of the Q_i . Accordingly, a q -local unitary of depth h is a product of h q -local unitaries of depth one. We note that transversal operators are 1-local unitaries of depth one.

V. DISTANCE AND DISJOINTNESS

A fundamental property of stabilizer codes is the distance. Typically, one says that the code has distance d if it can detect any error that affects at most $d - 1$ qudits. Here, however, we consider distance with respect to the qudit partition $\{Q_i\}$. First, we define the distance $d(G)$ of the nontrivial logical operator $G \in \mathcal{L}$ to be the size of the smallest support of any of its representatives:

$$d(G) = \min_{g \in G} |\text{supp}(g)|. \quad (8)$$

Then, we introduce two notions of the distance d_\downarrow and d_\uparrow , the min- and max-distance of the code, as follows:

$$d_\downarrow = \min_{G \in \mathcal{L}} d(G), \quad d_\uparrow = \max_{G \in \mathcal{L}} d(G). \quad (9)$$

We call a code ‘‘error detecting’’ iff its min-distance d_\downarrow is greater than 1. Note that the min-distance d_\downarrow is never greater than the (standard) distance d of the code. Also, if we choose a single-qudit partition, then those two quantities coincide, $d_\downarrow = d$.

In this article, we propose a new quantity for quantum stabilizer codes, the disjointness, which proves remarkably useful for establishing limitations on logical gates. First, for any nontrivial logical operator $G \in \mathcal{L}$ and a positive integer $c \geq 1$, we define c -disjointness $\Delta_c(G)$ to be the maximal number (divided by c) of representatives of G chosen in such a way that at most c representatives have support on any Q_i , a subset of the qudit partition:

$$\Delta_c(G) = \frac{1}{c} \max_{A \subseteq G} \{ |A| : \text{at most } c \text{ elements } a \in A \text{ have support on any } Q_i \}. \quad (10)$$

We call the set A in Eq. (10) c -disjoint. To build intuition about the c -disjointness, consider a small example.

Example 2. Consider the $[[4, 2]]$ qubit code with the stabilizer group $\mathcal{S} = \langle X^{\otimes 4}, Z^{\otimes 4} \rangle$ and the single-qubit partition. There are four equivalent logical operators implementing a logical $\bar{X}_1 = X_1 X_2$, which form a set

$$G = \{X_1 X_2, X_3 X_4, Y_1 Y_2 Z_3 Z_4, Z_1 Z_2 Y_3 Y_4\}. \quad (11)$$

The set $\{X_1 X_2, X_3 X_4\}$ is a maximal 1-disjoint set, $\{X_1 X_2, X_3 X_4, Y_1 Y_2 Z_3 Z_4\}$ is a maximal 2-disjoint set, and G itself is a maximal 3-disjoint set. Thus, $\Delta_1(G) = 2$, $\Delta_2(G) = 3/2$, $\Delta_3(G) = 4/3$.

Now, we are ready to define the disjointness Δ of a code.

Definition 1 (disjointness). For any n -qudit stabilizer code with the set of nontrivial logical operators \mathcal{L} and a qudit partition $[n] = Q_1 \cup Q_2 \cup \dots \cup Q_N$, the disjointness is defined as

$$\Delta = \max_{c \geq 1} \min_{G \in \mathcal{L}} \Delta_c(G). \quad (12)$$

We illustrate disjointness with the following example of the 2D surface code.

Example 3. Consider the 2D surface code of size $l \times l$ encoding one logical qubit [25] and the single-qubit partition. We have $d_\uparrow = d(\bar{Y}) = 2l - 1$ and $d_\downarrow = d(\bar{X}) = d(\bar{Z}) = l$. Moreover, there are exactly l representatives of \bar{X} with weight l , and they are all disjoint. Thus, $\Delta_1(\bar{X}) = l$. Similarly, $\Delta_1(\bar{Z}) = l$. In contrast, different representatives of \bar{Y} necessarily overlap, but we can nevertheless find l representatives of minimal weight $2l - 1$, such that each qubit is in the support of at most two of them. Those representatives of \bar{Y} form a 2-disjoint set. Thus, $\Delta_1(\bar{Y}) = 1$, but $\Delta_2(\bar{Y}) = l/2$. Now, by Eq. (12), $\Delta \geq \min_{G \in \mathcal{L}} \Delta_c(G)$, where $c(G)$ may depend on G . Thus, even by just evaluating $\Delta_1(\bar{X})$, $\Delta_2(\bar{Y})$, and $\Delta_1(\bar{Z})$, we are able to conclude that the disjointness Δ of the surface code satisfies $\Delta \geq l/2$.

The disjointness Δ turns out to be an important quantity characterizing stabilizer codes. In particular, we use it to find bounds on the level of the logical Clifford hierarchy achievable with transversal (see Theorem 5 in Sec. VI) or constant-depth (see Theorem 9 in Sec. VII) logical unitaries. To facilitate further discussion, we present key properties of the disjointness.

Lemma 2 [properties of disjointness]. For any $[[n, k]]$ stabilizer code and any partition $[n] = Q_1 \cup Q_2 \cup \dots \cup Q_N$, the disjointness satisfies

- (i) $1 \leq \Delta \leq \min(d_\downarrow, N/d_\uparrow)$.
- (ii) $\Delta > 1$ iff the stabilizer code is error detecting, i.e., $d_\downarrow > 1$.

Proof.—We begin by proving four bounds on c -disjointness that together imply case (i). In particular, let $G, G' \in \mathcal{L}$ be two noncommuting, nontrivial logical operators. In other words, $[g, g'] \neq I$ for all $g \in G$ and $g' \in G'$. Then, for any $1 \leq c \leq m^{n-k}$ (recall m is the qudit dimension),

$$1 \leq \Delta_c(G) \leq m^{n-k}/c, \quad (13)$$

$$\Delta_c(G) \leq d(G'), \quad (14)$$

$$\Delta_c(G) d(G) \leq N. \quad (15)$$

Moreover, each upper bound holds for all $c \geq 1$.

The lower bound in Eq. (13) is true because any $c \leq m^{n-k}$ elements of G form a c -disjoint set of size c . The upper

bound in Eq. (13) results because any c -disjoint set $A \subseteq G$ satisfies $|A| \leq |G| = m^{n-k}$. As a result of the upper bound, for any $c > m^{n-k}$, $\Delta_c(G) < 1$. Along with the lower bound, this implies $\min_{c \geq 1} \Delta_c(G) = \min_{1 \leq c \leq m^{n-k}} \Delta_c(G)$ for all $G \in \mathcal{L}$, which simplifies the definition of disjointness in Eq. (12).

For Eq. (14), choose a maximal c -disjoint set $A \subseteq G$ and a representative $g' \in G'$ of minimal support. In other words, $|A| = c\Delta_c(G)$ and $|\text{supp}(g')| = d(G')$. By definition, every $g \in A$ does not commute with g' . Thus, g and g' must have nontrivial overlap, $|\text{supp}(g) \cap \text{supp}(g')| \geq 1$. Consider any collection $H \subseteq [N]$ of some qudit subsets Q_i . Since at most c elements of A intersect at any subset of qudits Q_i , we have the inequality

$$\sum_{g \in A} |\text{supp}(g) \cap H| = \sum_{g \in A} \sum_{i \in H} |\text{supp}(g) \cap \{i\}| \quad (16)$$

$$\leq \sum_{i \in H} c \cdot 1 = c|H|. \quad (17)$$

Therefore, $c\Delta_c(G) = |A| = \sum_{g \in A} 1 \leq \sum_{g \in A} |\text{supp}(g) \cap \text{supp}(g')| \leq cd(G')$, proving Eq. (14).

Similarly, Eq. (15) follows from Eqs. (16) and (17) by setting $H = [N]$ and using $|\text{supp}(g)| \geq \min_{p \in G} |\text{supp}(p)| = d(G)$.

To get case (i) from Eqs. (13)–(15), note that they each hold for all c , so we can replace $\Delta_c(G)$ with $\max_{c \geq 1} \Delta_c(G)$ in all three equations. Since Eq. (13) also holds for all $G \in \mathcal{L}$, minimizing it over G immediately implies $1 \leq \Delta$ as well. In Eq. (14), take $G' \in \mathcal{L}$ such that $d(G') = d_\downarrow$ (and G to be any anticommuting logical Pauli), and in Eq. (15), take $G \in \mathcal{L}$ so that $d(G) = d_\uparrow$ to conclude $\Delta \leq \max_{c \geq 1} \Delta_c(G) \leq d_\downarrow$ and $\Delta \leq \max_{c \geq 1} \Delta_c(G) \leq N/d_\uparrow$, respectively.

We now prove case (ii). First, note that the implication $d_\downarrow = 1 \Rightarrow \Delta = 1$ follows from case (i). To show $d_\downarrow > 1 \Rightarrow \Delta > 1$, we establish a stronger fact: For all $G \in \mathcal{L}$, if $d_\downarrow > 1$, then for $c = \lceil d(G)/(d_\downarrow - 1) \rceil$, we have $\Delta_c(G) \geq 1 + 1/\lceil d(G)/(d_\downarrow - 1) \rceil$. We make use of the following version of the cleaning lemma.

Lemma 3 (cleaning lemma [26,27]). For any nontrivial logical operator $G \in \mathcal{L}$ and any collection $R \subseteq [N]$ of qudit subsets Q_i such that $|R| < d_\downarrow$, there exists a representative $g \in G$ not supported on R , i.e., $\text{supp}(g) \cap R = \emptyset$.

Suppose g is a minimal weight representative of G and set $H = \text{supp}(g)$. Without loss of generality, we assume $H = [d(G)]$ (which might involve relabeling the qudit subsets Q_i). For any $R \in H$ with $|R| < d_\downarrow$, Lemma 3 guarantees that we can find $g_R \in G$ that is not supported on R . Partition H into $c = \lceil d(G)/(d_\downarrow - 1) \rceil$ disjoint sets R_1, R_2, \dots, R_c , and take $A = \{g, g_{R_1}, g_{R_2}, \dots, g_{R_c}\}$. By construction, there are at most $c = |A| - 1$ elements of the set A intersecting at any qudit subset Q_i . Namely, if $i \in H$,

then $i \notin \text{supp}(g_{R_j})$ for the j such that $i \in R_j$, whereas if $i \notin H$, then $i \notin \text{supp}(g)$.

Thus, the set A can serve as an example of a c -disjoint subset of G , and we obtain a lower bound $\Delta_c(G) \geq |A|/c = 1 + 1/c > 1$ on the c -disjointness of G . This, in turn, implies that the disjointness Δ of the code is greater than 1, $\Delta \geq 1 + 1/\lceil d_\uparrow/(d_\downarrow - 1) \rceil > 1$, finishing the proof of case (ii). \square

Certain codes even have disjointness saturating the upper bound in Lemma 2(i), as in the following example.

Example 4. Consider the family of Reed-Muller codes $\llbracket n = 2^{D+1} - 1, k = 1 \rrbracket$ for $D \geq 2$, which coincides with a family of color codes of distance three in D spatial dimensions [28–30]. We consider the single-qubit partition. The two smallest codes in this family correspond to the 7-qubit Steane and the 15-qubit Reed-Muller codes. The distance of logical $\bar{X}, \bar{Y}, \bar{Z}$ operators satisfies $d(\bar{X}) = d(\bar{Y}) = 2^D - 1$ and $d(\bar{Z}) = 3$. Thus, $d_\downarrow = 3$ and $d_\uparrow = 2^D - 1$. There are 2^{D+1} representatives of \bar{X} , and $2^{D+1} - 1$ of them have minimal support. The set of minimal representatives of \bar{X} is, in fact, d_\uparrow -disjoint, and therefore $\Delta_{d_\uparrow}(\bar{X}) = n/d_\uparrow$. Moreover, for each representative g of \bar{X} , one can always find at least one representative of \bar{Z} (and thus of \bar{Y}) supported on $\text{supp}(g)$. We obtain that $\Delta_{d_\uparrow}(\bar{Z}), \Delta_{d_\uparrow}(\bar{Y}) \geq \Delta_{d_\uparrow}(\bar{X}) = n/d_\uparrow$, which results in a bound on the disjointness $\Delta \geq n/d_\uparrow$. However, $\Delta \leq n/d_\uparrow$ from Lemma 2(i), implying $\Delta = n/d_\uparrow$.

The c -disjointness $\Delta_c(G)$ of a nontrivial logical operator $G \in \mathcal{L}$ quantifies how well G can be “cleaned” (in the sense of Ref. [26]) from an arbitrary subset of qudits. We conclude this section with a useful lemma that is needed to prove the main results of our work.

Lemma 4 (scrubbing lemma). Consider a nontrivial logical operator $G \in \mathcal{L}$ and a collection $H \subseteq [N]$ of qudit subsets Q_i . For any $1 \leq c \leq m^{n-k}$, there exists a representative $g \in G$ such that

$$\Delta_c(G)|\text{supp}(g) \cap H| \leq |H|. \quad (18)$$

Proof.—Let $A \subseteq G$ be a maximal c -disjoint set, $|A| = c\Delta_c(G)$. Then, choosing $g = \arg\min_{g' \in A} |\text{supp}(g') \cap H|$, we have

$$\Delta_c(G)|\text{supp}(g) \cap H| \quad (19)$$

$$\leq \frac{1}{c} \sum_{g' \in A} |\text{supp}(g') \cap H| \leq |H|, \quad (20)$$

where we use Eqs. (16) and (17) for the second inequality. \square

We note that if $\Delta_c(G) = 1$, then the bound in Lemma 4 is trivial, $|\text{supp}(g) \cap H| \leq |H|$. We get a nontrivial bound

whenever $\Delta_c(G) > 1$, which is exactly the situation for error-detecting stabilizer codes; see Lemma 2(ii).

VI. LIMITATIONS ON TRANSVERSAL GATES

In this section, we use the disjointness to bound the transversal logical gates on any error-detecting stabilizer code to the Clifford hierarchy of the logical Pauli group $\bar{C}_M = C_M(\bar{P})$. We start with a theorem for transversal operators on a single code block, which we later generalize to operators between r code blocks.

Theorem 5. Consider a stabilizer code with min-distance d_\downarrow , max-distance d_\uparrow , and disjointness Δ . If M is an integer satisfying

$$d_\uparrow < d_\downarrow \Delta^{M-1}, \quad (21)$$

then all transversal logical operators are in the M th level of the Clifford hierarchy \bar{C}_M .

Although we prove this theorem here for subspace stabilizer codes, we can extend it to subsystem stabilizer codes as well; see Appendix A for details.

Proof.—Let $G_j \in \mathcal{L}$ be any nontrivial logical Pauli operator, and let K_0 be a transversal logical operator. We choose a representative g_1 of G_1 to have minimal support, $|\text{supp}(g_1)| = d(G_1)$. For $j \geq 1$, we recursively define $K_j = [K_{j-1}, g_j]$, which is a transversal logical operator, and we find $g_{j+1} \in G_{j+1}$ satisfying Lemma 4 with $H = \text{supp}(K_j)$. Notice that bounding the support of the group commutator of two transversal operators U_1, U_2 is especially simple,

$$\text{supp}([U_1, U_2]) \subseteq \text{supp}(U_1) \cap \text{supp}(U_2), \quad (22)$$

which leads to the following bound for $j > 1$:

$$|\text{supp}(K_j)| \leq |\text{supp}(K_{j-1}) \cap \text{supp}(g_j)| \quad (23)$$

$$\leq |\text{supp}(K_{j-1})| / \Delta_{c_j}(G_j), \quad (24)$$

where the first and second inequalities were obtained by using Eq. (22) and Lemma 4, respectively. Since we may choose arbitrary c_j , we set $c_j = \text{argmax}_{c \geq 1} \Delta_c(G_j)$. Now, using Eq. (23) recursively, we find

$$|\text{supp}(K_M)| \leq |\text{supp}(K_1)| \prod_{j=2}^M \Delta_{c_j}(G_j)^{-1} \quad (25)$$

$$\leq d_\uparrow / \Delta^{M-1} \leq d_\downarrow, \quad (26)$$

where in the second inequality we used $|\text{supp}(K_1)| \leq |\text{supp}(g_1)| \leq d(G_1) \leq d_\uparrow$ and $\Delta \geq \Delta_{c_j}(G_j)$. Since $|\text{supp}(K_M)|$ is smaller than the min-distance d_\downarrow of the code, K_M has to be a trivial logical operator. Therefore, by definition of the

Clifford hierarchy, we recursively obtain that K_{M-j} is a logical operator from the j th level. In particular, K_0 must be in the M th level \bar{C}_M . \square

We remark that Theorem 5 implies that transversal operators on a single code block of any error-detecting code must be in a finite level of the Clifford hierarchy. Namely, from Lemma 2(ii), we get $\Delta > 1$, and thus we can always find an integer $M = \lceil \log_\Delta(d_\uparrow/d_\downarrow) \rceil$ satisfying Eq. (21). We illustrate Theorem 5 with the following examples.

Example 5. The non-CSS 5-qubit stabilizer code [31,32] has the stabilizer group $\mathcal{S} = \langle Z_1 Z_2 X_3 X_5, X_1 Z_2 Z_3 X_4, X_2 Z_3 Z_4 X_5, X_1 X_3 Z_4 Z_5 \rangle$ and logical Pauli representatives $\bar{X} = X^{\otimes 5}$, and $\bar{Z} = Z^{\otimes 5}$ has $d_\uparrow = d_\downarrow = 3$ and $\Delta = 5/3$ with respect to the single-qubit partition. Thus, $d_\uparrow < d_\downarrow \Delta$, so transversal logical gates must be in the Clifford group. In fact, the 5-qubit code has a transversal logical Clifford gate SH .

Example 6. As we already discussed in Example 4, the Reed-Muller code $[[n = 2^{D+1} - 1, k = 1]]$ has parameters $d_\downarrow = 3$, $d_\uparrow = 2^D - 1$, and $\Delta = n/d_\uparrow$. Thus, Theorem 5 implies that the code can have transversal logical gates from at most the M th level of the Clifford hierarchy, where $M = \lceil \log_\Delta(d_\uparrow/d_\downarrow) \rceil = D$. In fact, the Reed-Muller code saturates this bound for any $D \geq 2$ since it has a transversal logical $\bar{R}_D = \text{diag}(1, e^{2\pi i/2^D})$ gate.

Example 7. Depending on the qubit partition of the $[[105, 1]]$ code from Example 1, its parameters are as follows: (a) $d_\downarrow = 3$, $d_\uparrow = 7$, and $\Delta = 15/7$ or (b) $d_\downarrow = d_\uparrow = 3$ and $\Delta = 7/3$. Thus, Theorem 5 limits transversal logical gates with respect to the qubit partition to (a) the third level of the Clifford hierarchy and (b) the Clifford group. We emphasize that the transversal gates on the $[[105, 1]]$ code saturate those bounds [24].

It is possible to treat multiple code blocks (these need not even be the same code) as one large effective code. If the b th code block has partition $\{Q_i^{(b)}\}$, one can define a partition $\{Q_i\}$ of the effective code with each Q_i consisting of (at most) one subset $Q_i^{(b)}$ from each code block. Moreover, if the partitions of each code block have distance $d_\downarrow^{(b)} > 1$, the partition of the effective code will also have $d_\downarrow > 1$. Then, applying Theorem 5 to the effective code leads to the following corollary.

Corollary 6. Transversal gates on error-detecting stabilizer codes must be in the Clifford hierarchy.

This in turn implies that the group of transversal logical gates on stabilizer codes is finite and not universal, providing an alternative proof of the main result of Ref. [7].

There are subtleties with this simple argument for multicode block operators. First, it leaves the possibility that the achievable level of the Clifford hierarchy might depend on the number of considered code blocks. Second,

the bound on the level is not conveniently stated in terms of d_\downarrow , d_\uparrow , Δ of the base code but rather in terms of the effective multiblock code. We address both of these issues in Appendix B, with more detailed arguments for the multicode block case. We summarize the results with the following version of Theorem 5 for stabilizer codes with multiple code blocks.

Theorem 7 (multi code block case). Consider an $[[n, k]]$ stabilizer code constructed from m -dimensional qudits. With respect to a partition of the qudits into N subsets Q_i , let the code's parameters be d_\downarrow , d_\uparrow , and Δ . Now, consider r code blocks of this code, and let $r' = \min(r, N!m^{n-k})$. If M is an integer satisfying

$$r'd_\uparrow(1 - (1 - 1/\Delta)^{r'})^{M-1} < d_\downarrow, \quad (27)$$

then all transversal logical operators on r code blocks are in the M th level of the hierarchy \bar{C}_M .

We remark that Theorem 7 can do more than rule out universal sets of transversal operators. Any set of operators that is capable of bootstrapping itself up the Clifford hierarchy indefinitely cannot be transversally implemented either. A simple example is the Toffoli gate, which can be used to implement an M -qubit controlled- X gate $C^{M-1}X$ for any M . Since the $C^{M-1}X$ gate is in the M th level, no stabilizer code can implement the Toffoli gate transversally; see Appendix B for more details. We remark that the same limitation on the transversal Toffoli gate was recently proved for most quantum codes by using entirely different means [15].

Finally, a further generalization of Theorem 5 comes by considering logical operators $K_0 = UP$ that can be written as a product of a transversal operator U and a permutation P of the subsets Q_i (allowing for a different permutation on every code block). We can similarly restrict such logical operators to the Clifford hierarchy; see Appendix C. However, for $r > 1$, these logical operators do not form a group, so there is no obvious analog of the Eastin-Knill theorem [6] for them.

VII. LIMITATIONS ON SHALLOW CIRCUITS

Our methods are powerful enough to put limitations on transversal as well as shallow-depth circuits which implement logical operators on stabilizer codes with respect to the given qudit partition. In this section, we find bounds on the level of the Clifford hierarchy achievable by q -local circuits of depth h (which may be geometrically nonlocal). The key ingredient needed to derive explicit bounds in terms of parameters of the code (d_\downarrow , d_\uparrow , Δ) and of the circuit (q , h) is the following lemma.

Lemma 8. Let A be a transversal operator and U be a q -local circuit of depth h . Then,

$$|\text{supp}([U, A])| \leq q^h |\text{supp}(U) \cap \text{supp}(A)|. \quad (28)$$

Proof.—First, we express the transversal operator $A = \prod_{i \in \text{supp}(A)} A_i$ as a product of operators A_i , each of which is supported only on one of the qudit subsets, i.e., $|\text{supp}(A_i)| = 1$. Then, $\text{supp}(A_i^\dagger) \subseteq \text{supp}(UA_iU^\dagger)$ and $|\text{supp}(UA_iU^\dagger)| \leq q^h$. Let $\mathcal{I} = \text{supp}(U) \cap \text{supp}(A)$, and then $[U, A] = (\prod_{i \in \mathcal{I}} UA_iU^\dagger) \prod_{i \in \mathcal{I}^c} A_i^\dagger$. Note that for any two operators V and W , we have $\text{supp}(VW) \subseteq \text{supp}(V) \cup \text{supp}(W)$. Using this fact, we get

$$\text{supp}([U, A]) \subseteq \bigcup_{i \in \mathcal{I}} \text{supp}(UA_iU^\dagger) \cup \bigcup_{i \in \mathcal{I}^c} \text{supp}(A_i^\dagger) \quad (29)$$

$$= \bigcup_{i \in \mathcal{I}} \text{supp}(UA_iU^\dagger), \quad (30)$$

and then using the union bound, we arrive at

$$|\text{supp}([U, A])| \leq \sum_{i \in \mathcal{I}} |\text{supp}(UA_iU^\dagger)| \leq |\mathcal{I}|q^h. \quad (31)$$

This finishes the proof since $|\mathcal{I}| = |\text{supp}(U) \cap \text{supp}(A)|$. \square

With Lemma 8, we update Eq. (23) to read

$$|\text{supp}(K_j)| \leq q^{h_{j-1}} |\text{supp}(K_{j-1}) \cap \text{supp}(g_j)| \quad (32)$$

$$\leq q^{h_{j-1}} |\text{supp}(K_{j-1})| / \Delta_{c_j}(G_j), \quad (33)$$

where $h_{j-1} = 2^{j-1}h$ is an upper bound on the depth of K_{j-1} [33]. Accordingly, by repeating the argument recursively, we obtain a version of Theorem 5 for q -local circuits of depth h .

Theorem 9 (shallow circuit case). Consider a stabilizer code with min-distance d_\downarrow , max-distance d_\uparrow , and disjointness Δ . If M is an integer satisfying

$$d_\uparrow q^{(2^M - 1)h} < d_\downarrow \Delta^{M-1}, \quad (34)$$

then all logical operators implemented by q -local circuits of depth h are in the M th level of the hierarchy \bar{C}_M .

We remark that, unlike in Theorem 5 for transversal operators, there is no guarantee that there exists M satisfying Eq. (34) for $q > 1$. Nevertheless, the shallow circuit version Theorem 9 is still useful for bounding logical gates on code families in the asymptotic limit. Namely, consider a family of codes $[[n(l), k(l)]]$ with parameters $d_\downarrow(l)$, $d_\uparrow(l)$, and $\Delta(l)$ with respect to some qudit partitions, parametrized by a positive integer l . We say that the code family has a q -local logical gate of depth h if there exists a constant l_0 such that for all $l \geq l_0$ one can implement the logical gate in the corresponding codes with some q -local circuits of depth h . To rule out logical gates from outside the M th level of the hierarchy \bar{C}_M with constant depth $h = h(l)$ and constant locality $q = q(l)$, it is

therefore sufficient to consider the limit of Eq. (34). We arrive at the following corollary.

Corollary 10. If for a family of stabilizer codes $[[n(l), k(l)]]$ with parameters $d_{\downarrow}(l)$, $d_{\uparrow}(l)$, and $\Delta(l)$ there exists an integer M satisfying

$$\lim_{l \rightarrow \infty} \frac{d_{\uparrow}(l)}{d_{\downarrow}(l)\Delta(l)^{M-1}} = 0, \quad (35)$$

then for any constants q and h , all q -local logical gates of depth h are in the M th level of the hierarchy \bar{C}_M .

We require that the limit vanish with l (rather than, say, just being less than 1) so that we can ignore the factors of constant locality and depth that appear in Eq. (34).

We conclude this section with a few examples illustrating the usefulness of Corollary 10.

Example 8. Consider the family of surface codes on square lattices of size $l \times l$. As shown in Example 3, the code parameters are $d_{\downarrow}(l) = l$, $d_{\uparrow}(l) = 2l - 1$, and $\Delta(l) \geq l/2$. Since for $M > 1$ we have

$$0 \leq \frac{d_{\uparrow}(l)}{d_{\downarrow}(l)\Delta(l)^{M-1}} \leq 2^{M-1} \frac{2l-1}{l^M} \xrightarrow{[l \rightarrow \infty]} 0, \quad (36)$$

then constant-depth, constant-locality circuits on surface codes can only implement logical Clifford gates.

Surprisingly, asymmetric 2D codes can have transversal logical non-Clifford gates. For instance, asymmetric Bacon-Shor codes have the transversal logical CCZ gate [18]. We emphasize that the asymmetry in the weight of different logical Pauli operators affects the ability to bound logical gates.

Example 9. Consider the stabilizer code family of asymmetric Bacon-Shor codes in the Z gauge on square lattices $l \times l^a$, $a \geq 1$. The code parameters $d_{\downarrow}(l) = l$ and $d_{\uparrow}(l) = l^a + l - 1$ are asymptotically different. Similarly to Example 3, we find $\Delta(l) \geq l/2$. For $M > a$, we have

$$0 \leq \frac{d_{\uparrow}(l)}{d_{\downarrow}(l)\Delta(l)^{M-1}} \leq 2^{M-1} \frac{l^a + l - 1}{l^M} \xrightarrow{[l \rightarrow \infty]} 0, \quad (37)$$

and thus constant-depth, constant-locality logical circuits on asymmetric Bacon-Shor codes are restricted to the $(\lfloor a \rfloor + 1)$ th level of the hierarchy $\bar{C}_{\lfloor a \rfloor + 1}$.

The multiblock versions of the asymptotic arguments [taking the limit of Eq. (27)] in these two examples yield the same bounds.

One can also generalize Example 8 to other topological codes that are equivalent to the D -dimensional toric code, such as the color code [17]. Choose logical Pauli X and Z operators to have representatives of dimensionality $D - s$ and s , where $1 \leq s \leq \lfloor D/2 \rfloor$. Then, given linear lattice size $O(l)$, the code parameters are $d_{\downarrow} = O(l^s)$, $d_{\uparrow} = O(l^{D-s})$, and $\Delta = O(l^s)$, and thus from Corollary 10, their logical

gates implemented via constant-depth (possibly geometrically nonlocal) circuits are limited to the M th level of the Clifford hierarchy, where $M = \lfloor (D - s)/s \rfloor + 1$. Note that as in Example 9, the greater the asymmetry of the support of the logical operators (or, in other words, the difference in the dimensionality of those operators), the higher the level of the Clifford hierarchy that is accessible. It is unclear though how to bound disjointness on more exotic topological codes with fractal-like logical operators, such as Haah's cubic code [34].

VIII. DISCUSSION

We have provided explicit upper bounds on the level of the Clifford hierarchy that is accessible for logical operators on any stabilizer code, which are implemented by transversal and constant-depth circuits. We expect our techniques to apply similarly to stabilizer codes composed of qudits, which differ in local dimension. As long as stabilizers and Pauli logical operators are tensor products of Pauli operators on physical qudits, the results and proofs presented should carry through. For example, an extension of our theorems to the dressed logical operators in stabilizer subsystem codes is provided in Appendix A.

At the same time, an alternative hierarchy bounding theorem exists without making use of the disjointness. We notice that in the proof of Theorem 5 instead of Lemma 4, we could use the following simple corollary of the Cleaning Lemma 3: For any nontrivial logical operator $G \in \mathcal{L}$ and a collection $H \subseteq [N]$ of qudit subsets Q_i satisfying $|H| \leq d_{\downarrow} - 1$, one can find a representative $g \in G$ such that $|H \cap \text{supp}(g)| \leq |H| - (d_{\downarrow} - 1)$. We follow the same recursive reduction of support of K_j as in Theorem 5 and obtain that if M is an integer satisfying

$$d_{\uparrow} < d_{\downarrow} + (M - 1)(d_{\downarrow} - 1), \quad (38)$$

then all transversal logical gates are in the M th level of the Clifford hierarchy [35]. Such an integer always exists if the stabilizer code is error detecting, i.e., $d_{\downarrow} > 1$. We note that the bound on M from Eq. (38) is rather loose. In particular, transversal logical gates on asymmetric Bacon-Shor codes of size $l \times O(l^2)$ are only restricted to the $O(l)$ th level, which is not useful for large l . On the other hand, Theorem 5 limits the gates to the third level, which is indeed accessible in this code family, as we have seen in Example 9. However, one can generally strengthen the bound in Eq. (38) by changing the partition. For instance, any transversal gate on the single-qudit partition would also be transversal on any other partition $\{Q_i\}$, so Eq. (38) applied to $\{Q_i\}$ can supply a bound that is stronger than the bound resulting from Eq. (38) applied to the single-qudit partition. Finding appropriate ‘‘coarsened’’ partitions can make using Eq. (38) unwieldy but could be similar to an exact calculation of the disjointness in difficulty. Further

work could be done to compare bounds obtained from Eqs. (38) and (21).

While our main results are derived without assumptions of geometric locality, we can derive even stronger bounds by assuming geometric locality of the circuits. For instance, the D -dimensional surface code encoding one logical qubit cannot implement non-Clifford logical operators with geometrically local, constant-depth circuits. The argument follows exactly the same lines as that in Sec. II, relying essentially on the ability to choose representatives g_1, g_2 of any two logical Pauli operators such that $|\text{supp}(g_1) \cap \text{supp}(g_2)| = O(1)$. Since a geometrically local circuit U cannot greatly distort the support of these representatives, $|\text{supp}([U, g_1, g_2])| = O(1)$ as well. The same argument holds if one superimposes D copies of the D -dimensional surface code, each encoding one logical qubit; geometrically local, constant-depth circuits still implement at most logical Cliffords. However, a gate from the D th level (namely $C^{D-1}Z$) is accessible if one rotates the D copies relative to one another [17].

The notion of disjointness for stabilizer codes, which we introduced, appears to be difficult to calculate exactly. If stabilizer codes have some underlying structure, such as Reed-Muller codes in Example 4 or topological codes in Examples 8 and 9, then we can find bounds on the disjointness, and this usually suffices to establish limits on the accessible level of the Clifford hierarchy. We believe that it is a challenging open problem to find efficient methods to compute (or approximate) the disjointness for an arbitrary stabilizer code. This problem, however, might be substantially simpler for topological codes, where one could exploit code and lattice symmetries. Also, it would be interesting to find possible relations to other new stabilizer code quantities, such as the price [36].

ACKNOWLEDGMENTS

The authors would like to thank Ben Brown, Steve Flammia, and Daniel Gottesman for helpful discussions. In particular, we would like to thank Michael Beverland for comments on the manuscript and for also showing us that the transversal gates of all stabilizer and subsystem codes are restricted to the Clifford hierarchy in unpublished work with John Preskill [35]. T. J. acknowledges the support from the Walter Burke Institute for Theoretical Physics in the form of the Sherman Fairchild Fellowship. A. K. acknowledges funding provided by the Simons Foundation through the ‘‘It from Qubit’’ Collaboration, as well as by the Institute for Quantum Information and Matter (IQIM), an NSF Physics Frontiers Center (NSF Grant No. PHY-1125565), with support from the Gordon and Betty Moore Foundation (GBMF-12500028). Research at Perimeter Institute is supported by the Government of Canada through Industry Canada and by the Province of Ontario through the Ministry of Research and Innovation. T. Y. is grateful for support from the Department of Defense (DoD) through the

National Defense Science and Engineering Graduate (NDSEG) Fellowship program and also an IBM Ph.D. Fellowship award. The authors acknowledge the MIT Open Access Article Publication Subvention Fund as well as IQIM for support in making this work available for open access publication.

APPENDIX A: SUBSYSTEM CODES

Subsystem stabilizer codes are subspace stabilizer codes $[[n, k, d]]$ for which we designate k_l qubits as logical and k_g qubits as gauge, $k_l + k_g = k$. Only the logical qubits contain protected information; the state of the gauge qubits can change arbitrarily during a computation [37]. Therefore, a logical operator U (a unitary on n qubits), by definition, acts on the logical + gauge qubits as a tensor product $A \otimes B$ (a unitary on k qubits), with A acting on the logical qubits and B on the gauge qubits. It cannot be otherwise—if U were not a tensor product over logical and gauge, then the change to the logical state would depend on the state of the gauge qubits, contradicting the principle that the state of the gauge qubits is irrelevant.

Now, we consider a transversal logical operator U on the subsystem code (such that it decomposes like $A \otimes B$ as above) and place A somewhere in the Clifford hierarchy. As before, we can use the fact that $A \in C_m$ if and only if, for all $p_j \in C_1$,

$$\dots[[A, p_1], p_2], \dots, p_m] = I. \quad (\text{A1})$$

However, because $[a \otimes b, c \otimes d] = [a, c] \otimes [b, d]$, we can instead show that for all $p_j \in C_1$, there are unitaries β_j and B on the gauge qubits such that

$$\dots[[A \otimes B, p_1 \otimes \beta_1], p_2 \otimes \beta_2], \dots, p_m \otimes \beta_m] = I \otimes B. \quad (\text{A2})$$

If we take β_j to also be Pauli operators, then $p_j \otimes \beta_j$ is just a so-called ‘‘dressed’’ logical Pauli operator of the subsystem code, which can be implemented transversally.

Consider a subsystem code with gauge group T and a logical Pauli operator \bar{p} . The whole set of logically equivalent dressed operators is $G = \bar{p}T$, a coset of the gauge group, which we call a dressed coset. Note that in the case of subspace stabilizer codes (discussed in the main text) with the stabilizer group S , we considered $G = \bar{p}S$. The dressed distance of G is the size of the support of the smallest element of G ,

$$d(G) = \min_{g \in G} |\text{supp}(g)|. \quad (\text{A3})$$

If \mathcal{L} is the set of dressed cosets for all nontrivial logical Pauli operators (i.e., all $4^{k_l} - 1$ of them), then the subsystem code also has a dressed min-distance and a dressed max-distance,

$$d_{\downarrow} = \min_{G \in \mathcal{L}} d(G), \quad (\text{A4})$$

$$d_{\uparrow} = \max_{G \in \mathcal{L}} d(G). \quad (\text{A5})$$

Likewise, we define the c -disjointness of a dressed coset G to be

$$\Delta_c(G) = \frac{1}{c} \max_{A \subseteq G} \{ |A| : \text{at most } c \text{ elements } a \in A \text{ have support on any } \mathcal{Q}_i \}, \quad (\text{A6})$$

i.e., exactly the same as for the nondressed cosets of a subspace code. Note that, if the stabilizer of the code is $S = Z(T)$ (i.e., the center of the gauge group), then

$$\Delta_c(\bar{p}S) \leq \Delta_c(\bar{p}T), \quad (\text{A7})$$

simply because $S \leq T$. The disjointness of the code is defined as before as well:

$$\Delta = \max_{c \geq 1} \min_{G \in \mathcal{L}} \Delta_c(G). \quad (\text{A8})$$

Now, the key to proving a Clifford hierarchy bound is a scrubbing theorem. But since all these definitions for subsystem codes are the same as for the subspace versions, the scrubbing lemma is also the same as stated in Lemma 4. In other words, for any $c \geq 1$, $G \in \mathcal{L}$, and $H \subseteq [N]$, it allows us to find a dressed Pauli operator $g \in G$ such that

$$\Delta_c(G) |\text{supp}(g) \cap H| \leq |H|. \quad (\text{A9})$$

Using this scrubbing lemma to reduce the support of the nested commutators $K_0 = U$, $K_j = [K_{j-1}, g_j]$ below the dressed distance d_{\downarrow} , we can prove the same bounding theorem: If

$$d_{\uparrow} < d_{\downarrow} \Delta^{M-1}, \quad (\text{A10})$$

then transversal logical gates are in \bar{C}_M .

The final question is whether, for the subsystem code with the gauge group T , the min-distance $d_{\downarrow}^{(T)} > 1$ implies that the disjointness $\Delta^{(T)} > 1$, as in the case of subspace codes. Note that if $d_{\downarrow}^{(T)} > 1$, then $d_{\downarrow}^{(S)} \geq d_{\downarrow}^{(T)} > 1$ for the subspace code with the stabilizer group $S = Z(T)$. Then, using Lemma 2(ii), we obtain that the disjointness of the subspace code $\Delta^{(S)} > 1$, and subsequently $\Delta^{(T)} \geq \Delta^{(S)} > 1$ by using Eq. (A7). Thus, we have bounded all transversal gates on error-detecting subsystem stabilizer codes to the Clifford hierarchy.

In the remainder of this appendix, we discuss how the techniques of gauge fixing fit together with our results. We also consider transversal code switching (called transversal ‘‘morphisms’’ in Ref. [9]) more generally.

Gauge fixing refers to the process of measuring the k_g gauge qubits of a subsystem stabilizer code and placing them in some fixed stabilizer state (e.g., $|0^{\otimes k_g}\rangle$). The result is a subspace stabilizer code (since there are effectively no more gauge operators), which we call a gauge-fixed code. This process is not transversal by our definitions, but it allows one to switch between using transversal logical operators on the subsystem code and transversal logical operators on the gauge-fixed code. Logical operators can be transversal on one of these two codes without being transversal on the other, and thus, together, they can get around the no-go theorems to form a universal set.

The following example exemplifies how gauge fixing works. The 15-qubit Reed-Muller subsystem code has $k_l = 1$ logical qubit and $k_g = 6$ gauge qubits [38]. The subsystem code has a transversal H gate that acts as H on the logical qubit and $H^{\otimes 6}$ on the gauge qubits. The gauge-fixed code in which the gauge qubits are in the state $|0^{\otimes 6}\rangle$ has a transversal T gate. Gauge fixing can be used after applying the transversal H to return the gauge qubits to $|0^{\otimes 6}\rangle$ so that T can be applied.

In addition, fault-tolerant operators would be transversal operators that, when applied to one gauge-fixed code, transform to another gauge-fixed code (of the same underlying subsystem code) while applying a logical operator to the logical qubits. More generally, Bravyi and König [9] define transversal morphisms, which are transversal operators that take a codespace of one subspace stabilizer code C_A to another C_B . Both codes involve the same number of physical and logical qubits. Following arguments like in the main text, the disjointness bounds such logical operators to the M th level of the hierarchy if

$$d_{\uparrow}^{(A)} < d_{\downarrow}^{(B)} (\Delta^{(B)})^{M-1}, \quad (\text{A11})$$

where we have included superscripts for the distance and disjointness parameters of the two codes C_A and C_B . It follows that morphisms between two particular codes are limited to being nonuniversal.

Still, one may wonder whether transversal morphisms that keep changing to new codes can form a universal, or at least infinite, set. For instance, we might start in a code $C = C_I$ for which transversal H leaves us in code C_H and transversal T leaves us in C_T . From C_H , we might have a transversal T that leaves us in code C_{TH} and a transversal H that leaves us in C_{HH} , and so on. This construction can be represented by a graphlike structure. The vertices of the graph correspond to different subspace stabilizer codes C_i . The edges incident to a code C_0 correspond to the transversal morphisms applicable to C_0 , with each edge leading to the code that the morphism switches to.

However, this scenario cannot achieve universality because of the following contradiction that makes use of Eq. (A11). First, note that for any finite number of qubits n , there are only a finite number of subspace stabilizer codes,

so the number of vertices in the graph is finite. Taking any code in the graph, say, C_0 , we can, by assumption, apply a universal and thus infinite set of logical gates to it via transversal morphisms, so C_0 has an infinite number of edges. Choose some neighbor C_1 of C_0 . There must be an infinite number of different transversal morphisms (i.e., edges) that take C_0 to C_1 . Yet, Eq. (A11) implies that there should only be a finite number (bounded within some level of the hierarchy) of such logical operators. It also does not seem possible to get around this by adding or subtracting qubits so that the codes change size, unless one were willing to add an infinite number of qubits over the course of the computation.

APPENDIX B: LOGICAL GATES ON MULTIPLE CODE BLOCKS

In this section, we describe how to restrict gates on multiple code blocks to the Clifford hierarchy. We flesh out the argument in the main text by giving formulas for d_\downarrow , d_\uparrow , Δ of the multicode block code in terms of those for the single code block. Then, we argue how the bound on the level of the Clifford hierarchy obtainable by transversal gates can be made independent of the number of code blocks.

Consider r code blocks of the same $[[n, k]]$ base stabilizer code, each with identical [40] qudit partitions, which we write as $\{Q_i^{(b)}\}$, where the superscript $b = 1, 2, \dots, r$ represents the code block. Like in the main text, we say there are N subsets $Q_i^{(b)}$ for each b , and we use d_\downarrow , d_\uparrow , Δ to denote the quantities of the base code.

The effective stabilizer code is formed by treating all r code blocks as a $[[rn, rk]]$ stabilizer code, and the qudits of the effective code can be partitioned into subsets $\{Q_i\}$, each consisting of one subset from each of the r code blocks,

$$Q_i = \bigcup_{b=1}^r Q_{\sigma_b(i)}^{(b)}. \quad (\text{B1})$$

Here, $\sigma_b: [N] \rightarrow [N]$ is an (arbitrary) permutation of the partitions of code block b . This completes the partitioning of the effective code in such a way that the effective code's min-distance equals that of the base code, $d_{\downarrow, \text{eff}} = d_\downarrow$. We also note the simple bound on the effective code's max-distance $d_{\uparrow, \text{eff}} \leq rd_\uparrow$.

The final quantity to address is the disjointness of the effective code Δ_{eff} . To do this, we prove a more general version of Lemma 4 for multiple code blocks, and we let this inform the definition of Δ_{eff} . Start by establishing some notation. Let \mathcal{L}_{eff} denote the set of nontrivial logical cosets of the effective code. Note that $G \in \mathcal{L}_{\text{eff}}$ means, by definition, that

$$G = \bigotimes_{b=1}^r G^{(b)}, \quad (\text{B2})$$

where $G^{(b)} \in \mathcal{L} \cup \{\mathcal{S}\}$ are logical cosets of the base code and at least one is nontrivial (i.e., in \mathcal{L}).

Lemma 11. Let $G \in \mathcal{L}_{\text{eff}}$ and $H \subseteq [N]$. Then, for any c_1, c_2, \dots, c_r , there exists a representative $g \in G$ such that

$$|\text{supp}(g) \cap H| \leq \left(1 - \prod_b (1 - \Delta_{c_b}(G^{(b)})^{-1})\right) |H|, \quad (\text{B3})$$

where the product ranges only over nontrivial cosets in the decomposition of G [Eq. (B2)].

Proof.—Without loss of generality, we say that only the first $r_0 \leq r$ cosets in Eq. (B2) are nontrivial. We decompose $g \in G$ as $g = \bigotimes_{b=1}^r g^{(b)}$ with $g^{(b)} \in G^{(b)}$. Our task is to find $g^{(b)}$ such that Eq. (B3) holds. Start by noting

$$\text{supp}(g) = \bigcup_{b=1}^{r_0} \text{supp}(g^{(b)}), \quad (\text{B4})$$

$$H \cap \text{supp}(g) = \bigcup_{b=1}^{r_0} (H \cap \text{supp}(g^{(b)})). \quad (\text{B5})$$

Say that we have already chosen $g^{(1)}, g^{(2)}, \dots, g^{(j-1)}$. Then, we need only minimize the intersection of $g^{(j)}$ with

$$H_{j-1} := H - \bigcup_{b=1}^{j-1} (H \cap \text{supp}(g^{(b)})), \quad (\text{B6})$$

the set of partitions in H that are still unaffected. By Lemma 4, we can find $g^{(j)} \in G^{(j)}$ such that

$$|\text{supp}(g^{(j)}) \cap H_{j-1}| \leq |H_{j-1}| / \Delta_{c_j}(G^{(j)}). \quad (\text{B7})$$

Note the relations

$$H_0 = H, \quad (\text{B8})$$

$$H_j = H_{j-1} - (\text{supp}(g^{(j)}) \cap H_{j-1}), \quad (\text{B9})$$

$$H - H_{r_0} = H \cap \text{supp}(g). \quad (\text{B10})$$

Thus, Eq. (B7) implies

$$\begin{aligned} |H_j| &= |H_{j-1} - (\text{supp}(g^{(j)}) \cap H_{j-1})| \\ &\geq (1 - \Delta_{c_j}(G^{(j)})^{-1}) |H_{j-1}|. \end{aligned} \quad (\text{B11})$$

Repetitive use of Eq. (B11) gives us the bound

$$|H_{r_0}| \geq \prod_{b=1}^{r_0} (1 - \Delta_{c_b}(G^{(b)})^{-1}) |H|, \quad (\text{B12})$$

from which we conclude

$$|H - H_{r_0}| = |H \cap \text{supp}(g)| \leq \left(1 - \prod_{b=1}^{r_0} (1 - \Delta_{c_b} (G^{(b)})^{-1})\right) |H|. \quad (\text{B13})$$

This completes the proof. \square

We can simplify Eq. (B3) by choosing specific c_b such that $\Delta \leq \Delta_{c_b} (G^{(b)})$ and find a $g \in G$ such that

$$|H \cap \text{supp}(g)| \leq (1 - (1 - 1/\Delta)^{r_0}) |H| \quad (\text{B14})$$

$$\leq (1 - (1 - 1/\Delta)^r) |H|. \quad (\text{B15})$$

The latter form of the right-hand side implies that defining

$$\Delta_{\text{eff}} := \frac{1}{1 - (1 - 1/\Delta)^r} \quad (\text{B16})$$

will result in a theorem analogous to Theorem 5 but for multiple code blocks.

Theorem 12. If $d_{\uparrow, \text{eff}} < d_{\downarrow, \text{eff}} \Delta_{\text{eff}}^{M-1}$, then all transversal gates on r code blocks are in \bar{C}_M .

Proof.—This follows the same lines as the proof of Theorem 5 but using Eq. (B14) in place of Lemma 4. \square

Of course, we can write the condition of Theorem 12 solely in terms of the single code block parameters d_{\downarrow} , d_{\uparrow} , Δ . In other words, if

$$rd_{\uparrow} < d_{\downarrow} \frac{1}{(1 - (1 - 1/\Delta)^r)^{M-1}}, \quad (\text{B17})$$

then all transversal gates are in \bar{C}_M . Since $\Delta > 1$ for error-detecting stabilizer codes [Lemma 2(ii)], $\Delta_{\text{eff}} > 1$ as well, and the right-hand side of Eq. (B17) must exceed the left for some sufficiently large $M \geq M_0$.

However, given only the arguments until now, it is still possible that M_0 depends on r and even that increasing r arbitrarily can increase M_0 arbitrarily as well. This would imply that high-level transversal gates between different code blocks are easier to find than transversal single-block gates. While this may be true to some extent, there is a limit, which we describe now.

The key is to realize in what instances we can find $g^{(j)}$ so that $|H_j| = |H_{j-1}|$ in Eq. (B11). This happens when we can choose $g^{(j)}$ so that

$$H \cap \text{supp}(g^{(j)}) \subseteq \bigcup_{b=1}^{j-1} H \cap \text{supp}(g^{(b)}). \quad (\text{B18})$$

For instance, in the simple case when σ_b are each the identity permutation, then whenever $G^{(b_1)} = G^{(b_2)} = G'$, we might as well choose the same representative $g' \in G'$ for both $g^{(b_1)}$ and $g^{(b_2)}$ because then

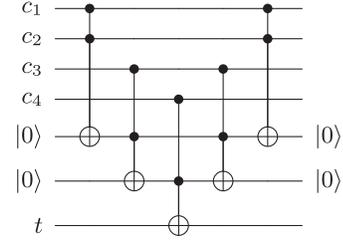


FIG. 3. Making a C^4X gate with controls c_1 , c_2 , c_3 , c_4 and target t from five Toffoli gates and two ancillas.

$$H \cap \text{supp}(g^{(b_1)}) = H \cap \text{supp}(g^{(b_2)}). \quad (\text{B19})$$

Moreover, we are guaranteed to start repeating cosets in the decomposition in Eq. (B2) when $r > m^{n-k}$, so effectively, we can replace r in Eq. (B17) with $\min(r, m^{n-k})$, thus achieving an r -independent bound.

When σ_b is arbitrary, we can make the same argument when $G^{(b_1)} = G^{(b_2)}$ and $\sigma_{b_1} = \sigma_{b_2}$. Since there are finitely many permutations as well, we can replace r in Eq. (B17) with $\min(r, N!m^{n-k})$, which admittedly is large but at least finite. The previous arguments complete the proof of Theorem 7.

The upshot of these finite bounds on M_0 is that we can state further no-go theorems on what particular gates can be implemented on stabilizer codes, such as in the following.

Corollary 13. No error-detecting stabilizer code (on qubits) can implement Toffoli transversally.

Proof.—There is a well-known construction [41] where, for any integer $w \geq 2$, $2w - 3$ Toffoli gates and $2w - 1$ qubits ($w - 2$ of which are ancillas) suffice to make C^wX , i.e., X with w control qubits. See Fig. 3 for an example with $w = 4$. Since $C^wX \in C_{w+1}$, we see that having transversal Toffoli would imply transversal gates in every level of the Clifford hierarchy. But this is ruled out by the finite bound on the level argued for above. \square

The same conclusion was shown for most quantum codes in Ref. [15] by reduction to bounds in homomorphic encryption. Our proof technique can be applied to any other set of gates that is, like Toffoli, capable of bootstrapping itself indefinitely up the hierarchy.

APPENDIX C: TRANSVERSAL GATES WITH PERMUTATIONS

In this section, we extend Theorem 7 to the case of permuting transversal operators K_0 , which are those that can be written as $K_0 = UP$ for transversal U and permutation P of the partitions \mathcal{H}_i separately for each code block.

As in the proof of Theorem 5, take an arbitrary sequence of cosets $G_1, G_2, \dots \in \mathcal{L}$ and define $K_1 = [K_0, g_1] = UPg_1PU^\dagger g_1^\dagger$ and $K_j = [K_{j-1}, g_j]$ for some choices of $g_j \in G_j$. The key thing to notice is that the recursive reduction of support of the K_j is modified only at K_1 . Take

$g_1 \in G_1 \in \mathcal{L}$ to have minimal support $|\text{supp}(g_1)| = d(G_1)$ so that

$$|\text{supp}(K_1)| \leq 2d(G_1), \quad (\text{C1})$$

simply because Pg_1P may have disjoint support from g_1^\dagger . Bounding the supports of K_j can then be done exactly as in the proof of Theorem 5. More generally, the argument for Theorem 7 found in Appendix B can incorporate the observation in Eq. (C1) to show the following.

Theorem 14. Consider a stabilizer code with quantities $d_\downarrow, d_\uparrow, \Delta$. Let $r' = \min(r, N!m^{n-k})$. If

$$2r'd_\uparrow(1 - (1 - 1/\Delta)^{r'})^{M-1} < d_\downarrow, \quad (\text{C2})$$

then all permuting transversal gates on r code blocks are in \tilde{C}_M . When $r = 1$,

$$2d_\uparrow < d_\downarrow \Delta^{M-1} \quad (\text{C3})$$

implies the same for one code block.

Notice that for single code blocks $r = 1$, the permuting transversal operators K_0 do form a group, and thus this theorem has a corollary that the group of permuting transversal operators on a single code block is finite and nonuniversal.

-
- [1] P. W. Shor, *Scheme for Reducing Decoherence in Quantum Computer Memory*, *Phys. Rev. A* **52**, R2493 (1995).
- [2] A. M. Steane, *Error Correcting Codes in Quantum Theory*, *Phys. Rev. Lett.* **77**, 793 (1996).
- [3] J. Preskill, *Fault-Tolerant Quantum Computation*, *Int. Quantum Comput. Inf.* **213** (1998).
- [4] D. Gottesman, *Theory of Fault-Tolerant Quantum Computation*, *Phys. Rev. A* **57**, 127 (1998).
- [5] A. Y. Kitaev, *Fault-Tolerant Quantum Computation by Anyons*, *Ann. Phys. (Amsterdam)* **303**, 2 (2003).
- [6] B. Eastin and E. Knill, *Restrictions on Transversal Encoded Quantum Gate Sets*, *Phys. Rev. Lett.* **102**, 110502 (2009).
- [7] B. Zeng, A. W. Cross, and I. L. Chuang, *Transversality Versus Universality for Additive Quantum Codes*, *IEEE Trans. Inf. Theory* **57**, 6272 (2011).
- [8] J. T. Anderson and T. Jochym-O'Connor, *Classification of Transversal Gates in Qubit Stabilizer Codes*, *Quantum Inf. Comput.* **16**, 0771 (2016).
- [9] S. Bravyi and R. König, *Classification of Topologically Protected Gates for Local Stabilizer Codes*, *Phys. Rev. Lett.* **110**, 170503 (2013).
- [10] F. Pastawski and B. Yoshida, *Fault-Tolerant Logical Gates in Quantum Error-Correcting Codes*, *Phys. Rev. A* **91**, 012305 (2015).
- [11] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, *Topological Quantum Memory*, *J. Math. Phys. (N.Y.)* **43**, 4452 (2002).
- [12] H. Bombin and M. A. Martin-Delgado, *Topological Quantum Distillation*, *Phys. Rev. Lett.* **97**, 180501 (2006).
- [13] M. E. Beverland, O. Buerschaper, R. Koenig, F. Pastawski, J. Preskill, and S. Sijher, *Protected Gates for Topological Quantum Field Theories*, *J. Math. Phys. (N.Y.)* **57**, 022201 (2016).
- [14] P. Webster and S. D. Bartlett, *Locality-Preserving Logical Operators in Topological Stabiliser Codes*, *Phys. Rev. A* **97**, 012330 (2018).
- [15] M. Newman and Y. Shi, *Limitations on Transversal Computation through Quantum Homomorphic Encryption*, [arXiv:1704.07798](https://arxiv.org/abs/1704.07798).
- [16] H. Bombin, *Gauge Color Codes: Optimal Transversal Gates and Gauge Fixing in Topological Stabilizer Codes*, *New J. Phys.* **17**, 083002 (2015).
- [17] A. Kubica, B. Yoshida, and F. Pastawski, *Unfolding the Color Code*, *New J. Phys.* **17**, 083026 (2015).
- [18] T. J. Yoder, *Universal Fault-Tolerant Quantum Computation with Bacon-Shor Codes*, [arXiv:1705.01686](https://arxiv.org/abs/1705.01686).
- [19] Since we restrict unitary operators to the Clifford group, it is sufficient to consider generators \bar{X} and \bar{Z} of the logical Pauli group. However, restricting operators to levels of the Clifford hierarchy beyond the third requires considering all logical Pauli operators.
- [20] Qubit Pauli operators $m = 2$ are traditionally defined to be generated by X, Z , and also $Y = iXZ$. Doing so does not change our results.
- [21] D. Gottesman and I. L. Chuang, *Demonstrating the Viability of Universal Quantum Computation Using Teleportation and Single-Qubit Operations*, *Nature (London)* **402**, 390 (1999).
- [22] D. Gottesman, Ph.D. thesis, California Institute of Technology, 1997.
- [23] Our definition is nevertheless still consistent with the definition of Eastin and Knill [6].
- [24] T. Jochym-O'Connor and R. Laflamme, *Using Concatenated Quantum Codes for Universal Fault-Tolerant Quantum Gates*, *Phys. Rev. Lett.* **112**, 010505 (2014).
- [25] S. B. Bravyi and A. Y. Kitaev, *Quantum Codes on a Lattice with Boundary*, [arXiv:quant-ph/9811052](https://arxiv.org/abs/quant-ph/9811052).
- [26] S. Bravyi and B. Terhal, *A No-Go Theorem for a Two-Dimensional Self-Correcting Quantum Memory Based on Stabilizer Codes*, *New J. Phys.* **11**, 043029 (2009).
- [27] B. Yoshida and I. L. Chuang, *Framework for Classifying Logical Operators in Stabilizer Codes*, *Phys. Rev. A* **81**, 052302 (2010).
- [28] A. M. Steane, *Quantum Reed-Muller Codes*, *IEEE Trans. Inf. Theory* **45**, 1701 (1999).
- [29] J. T. Anderson, G. Duclos-Cianci, and D. Poulin, *Fault-Tolerant Conversion between the Steane and Reed-Muller Quantum Codes*, *Phys. Rev. Lett.* **113**, 080501 (2014).
- [30] A. Kubica and M. E. Beverland, *Universal Transversal Gates with Color Codes: A Simplified Approach*, *Phys. Rev. A* **91**, 032330 (2015).
- [31] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wootters, *Mixed-State Entanglement and Quantum Error Correction*, *Phys. Rev. A* **54**, 3824 (1996).
- [32] R. Laflamme, C. Miquel, J. P. Paz, and W. H. Zurek, *Perfect Quantum Error Correcting Code*, *Phys. Rev. Lett.* **77**, 198 (1996).

- [33] Since $K_j = K_{j-1}g_jK_{j-1}^\dagger g_j^\dagger$, one immediately obtains $h_j \leq 2h_{j-1} + 2$, where $h_0 = h$. However, the transversal operator g_j can be absorbed into neighboring gates in the circuit and, as a result, does not increase the circuit depth. Thus, we can remove the additive constant from the recursion.
- [34] J. Haah, *Local Stabilizer Codes in Three Dimensions without String Logical Operators*, *Phys. Rev. A* **83**, 042330 (2011).
- [35] M. E. Beverland and J. Preskill (private communication).
- [36] F. Pastawski and J. Preskill, *Code Properties from Holographic Geometries*, *Phys. Rev. X* **7**, 021022 (2017).
- [37] D. Poulin, *Stabilizer Formalism for Operator Quantum Error Correction*, *Phys. Rev. Lett.* **95**, 230504 (2005).
- [38] A. Paetznick and B.W. Reichardt, *Universal Fault-Tolerant Quantum Computation with Only Transversal Gates and Error Correction*, *Phys. Rev. Lett.* **111**, 090505 (2013).
- [39] Taking the blocks to be the same is for simplification of the argument only. For instance, we do not have to deal with different quantities d_\downarrow , d_\uparrow , Δ for each code. Running through a more general argument where the stabilizer codes are allowed to be different is possible, and similarly, this restricts transversal gates to the Clifford hierarchy.
- [40] Again, the sameness of the partitions can be relaxed at the cost of notational encumbrance.
- [41] M. A. Nielsen and I.L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, England, 2010).