

Efficient decoding of random errors for quantum expander codes

Omar Fawzi* Antoine Grospellier† Anthony Leverrier†

Abstract

We show that quantum expander codes, a constant-rate family of quantum LDPC codes, with the quasi-linear time decoding algorithm of Leverrier, Tillich and Zémor can correct a constant fraction of random errors with very high probability. This is the first construction of a constant-rate quantum LDPC code with an efficient decoding algorithm that can correct a linear number of random errors with a negligible failure probability. Finding codes with these properties is also motivated by Gottesman’s construction of fault tolerant schemes with constant space overhead.

In order to obtain this result, we study a notion of **α -percolation**: for a random subset E of vertices of a given graph, we consider the size of the largest connected α -subset of E , where X is an α -subset of E if $|X \cap E| \geq \alpha|X|$.

1 Introduction

A major goal of quantum information research is to build a large quantum computer that would exploit the laws of quantum mechanics to solve problems out of reach for classical computers. Perhaps the main difficulty of this program lies with the fragility of quantum information. While classical information can easily be stored and manipulated without error, this is not the case of quantum information which will tend to decohere quickly as soon as it interacts with the environment. Finding the right balance between protecting the qubits from decoherence and allowing them to be accessed in order to perform a computation turned out to be an extremely challenging task.

A natural solution to protect quantum information is to encode the logical qubits into a larger set of physical qubits thanks to quantum error correction techniques. Classically, good error correcting codes that feature a constant encoding rate, a minimal distance linear in their length and efficient decoding algorithms exist; this is for instance the case of low-density parity-check (LDPC) codes [24]. The situation for quantum codes is **direr**. Here, the LDPC condition is particularly appealing for implementation, but finding good quantum LDPC codes has proven difficult. The canonical quantum LDPC code is the toric code due to Kitaev [18], which is indeed local since only involving 4-body correlations and can be decoded efficiently thanks to Edmonds algorithm for minimum weight perfect matching [8] (or more recently [5]). Unfortunately, the toric code as well as its surface code generalizations display a zero rate, in the sense that the ratio between encoded and physical qubits tends to zero when the code size tends to infinity. The minimum distance of the toric code, on the other hand, is quite good and grows as $\Theta(n^{1/2})$ for a code of length n . This is essentially the best available value for LDPC codes since the current record is

*ENS de Lyon, UMR 5668 LIP - CNRS - UCBL - INRIA - Université de Lyon, 69364 Lyon, France. omar.fawzi@ens-lyon.fr

†Inria, SECRET Project, 2 Rue Simone Iff, 75012 Paris Cedex, France. antoine.grospellier@inria.fr, anthony.leverrier@inria.fr

$\Omega(n^{1/2} \log^{1/4} n)$ due to Freedman, Meyer and Luo [10] and it is a fascinating theoretical open question to construct quantum LDPC codes with minimum distance $\Omega(n^{1/2+\epsilon})$ for some constant $\epsilon > 0$. However, even if for the toric code and its generalizations there exist errors of weight $O(\sqrt{n})$ that cannot be corrected, it turns out that there are efficient decoding algorithms that can, with high probability, correct *random errors* on a linear number of qubits provided the error rate is below some threshold [8].

In this paper, we ask the following natural question: are there quantum LDPC codes with efficient decoding algorithms that display constant rate and can correct a linear number of random errors with very high probability? This question is also motivated by reducing the space overhead for fault-tolerant quantum computation. One of the crowning theoretical achievements in the field of quantum computation is the *threshold theorem* [2] which shows that an arbitrarily long computation can be performed with an imperfect implementation (noisy qubits, imperfect gates), provided that the noise level is below some constant threshold. This is achieved thanks to *quantum fault-tolerance* techniques, by replacing the target logical circuit using m qubits and containing T gates by a fault-tolerant circuit using $O(m \text{polylog}(mT))$ qubits. In a recent breakthrough paper [13], Gottesman has argued that the polylogarithmic space overhead isn't necessary and is in fact a consequence of using zero-rate codes such as the toric code in the fault-tolerant circuit. In contrast, one can achieve a *constant space overhead* by relying on constant-rate quantum LDPC codes satisfying a number of desirable properties.

The most important such property is that it should be able to efficiently correct with very high probability¹ random errors provided the error rate is below some threshold.² A main contender for such a family of codes is obtained from the hypergraph-product construction of Tillich and Zémor [27]. Starting with good classical LDPC codes, the construction yields quantum LDPC codes with constant rate and minimum distance growing like $n^{1/2}$. The reason they cannot directly be plugged in Gottesman's construction is the absence of an efficient decoding algorithm to correct a linear number of random errors. Other code constructions such as *hyperbolic codes in 4 dimensions* are also a candidate since they combine a constant rate with a polynomial minimum distance [14], but they lack an efficient decoding algorithm providing the required error suppression, see however [15, 21]. Finally, hyperbolic surface codes can also achieve constant rate but only with a logarithmic minimum distance [10, 17, 30]. While such codes come with an efficient decoding algorithm, for instance Edmonds' maximum matching algorithm [9], the success probability will not be very high and will scale like $1 - 1/\text{poly}(n)$ because the minimum distance is only logarithmic. As a result, they cannot be used to provide quantum fault tolerance with constant overhead and error thresholds independent of how the computation size scales with the number of qubits.

Our main result is to show that the family of *quantum expander codes* of Ref. [20] combine the following desirable properties: they are constant rate, with a quasi-linear time decoding algorithm that corrects adversarial errors of weight $O(\sqrt{n})$ and corrects random errors of linear size with very high probability provided that the error rate is below some constant value.

Main result and proof techniques

In this paper, we consider *quantum expander codes*, a family of constant-rate quantum hypergraph-product codes [27], and show that the efficient decoding algorithm introduced in Ref. [20] can correct a constant fraction of random errors. More precisely, we prove the existence of a threshold for various models of noise, including depolarizing noise, as well as more adversarial models which are relevant for quantum fault-tolerance. The decoding algorithm is inspired by the bit-flip

¹We use the term very high probability to refer to a probability $1 - \epsilon(n)$ where $\epsilon(n)$ is negligible, i.e., $n^c \epsilon(n) \rightarrow 0$ as $n \rightarrow \infty$ for any constant c .

²As is standard in the fault tolerance literature, we use the word “threshold” to mean that below this value, errors are corrected with high probability. And we do not care about what happens above this value.

decoding algorithm studied by Sipser and Spielman in the context of classical expander codes [25], and works by finding error patterns of small size that decrease the weight of the syndrome. For this reason, we'll refer to it as “small-set-flip” decoding algorithm in the sequel. Similarly to the classical case, it runs in quasi-linear time.

Our approach for the analysis of the decoding algorithm is inspired by the previous work of Kovalev and Pryadko [19] who studied the behaviour of the maximum likelihood decoding algorithm (that has exponential running time in general) applied to hypergraph-product codes and established the existence of a threshold for the random independent error model. Here, we use similar techniques to the study of a sub-optimal but efficient decoding algorithm, and also find the existence of a threshold.

Technically, we represent the set of qubits as a graph $\mathcal{G} = (\{1, \dots, n\}, \mathcal{E})$ called *adjacency graph* where the vertices correspond to the qubits of the code and two qubits are linked by an edge if there is a stabilizer generator that acts on the two qubits. The approach is then to show that provided the vertices E corresponding to the error do not form large *connected* subsets, the error can be corrected by the decoding algorithm. How large the connected subsets are allowed to be is related to the minimum distance of the code for the maximum-likelihood decoder, or to the maximum size of correctable errors for more general decoders. This naturally leads to studying the size of the largest connected subset of a randomly chosen set of vertices of a graph. This is also called site percolation on finite graphs and its connection with the performance of decoding algorithms is old and was used for example in [8], [6], [7] for the surface codes and in [19] for the maximum likelihood decoder of hypergraph product codes.

Here, in order to analyse the efficient small-set-flip decoding algorithm for expander graphs, we identify a slightly more complex notion of connectivity as relevant. Namely, instead of studying the size of the largest connected subset of E , we study the size of the largest connected α -subset of E . We say that X is an α -subset of E if $|X \cap E| \geq \alpha|X|$. Note that for $\alpha = 1$, this is the same as X is a subset of E . Our main technical contribution is to show that for some relevant noise models, if the probability of error of each qubit is below some threshold depending on α and the degree of \mathcal{G} , then the probability that a random set E has a connected α -subset of size $\omega(\log n)$ vanishes as n grows. As the minimum distance of expander codes is $\Omega(\sqrt{n})$ and the efficient decoding algorithm can correctly decode this many errors, we obtain the claimed result.

In fact, our analysis is not restricted to quantum expander codes and applies to any “local” decoding algorithm (see Section 3.3 for a definition). We show that a “local” decoding algorithm with parameter α can correct errors on a subset of qubits E if the size of the largest connected α -subset of E is of size less than t , where t is such that the decoding algorithm can correct all errors of weight at most t . A “local” decoding algorithm here refers to a decoding algorithm where in each step errors on distant qubits are decoded independently, and is satisfied by the small-set-flip decoding algorithm for quantum expander codes. Similarly, the bit-flip decoding algorithm for classical expander codes as well as the maximum likelihood decoders for classical LDPC codes or quantum CSS LDPC codes are local in this sense. Our main theorem is the following:

Theorem 1. *Consider a quantum expander code with sufficient expansion and the small-set-flip decoding algorithm (Algorithm 1). Then there exists a probability $p_0 > 0$ and constants C, C' such that if the noise parameter satisfies $p < p_0$, the small-set-flip decoding algorithm corrects a random error with probability at least $1 - Cn \left(\frac{p}{p_0}\right)^{C'\sqrt{n}}$.*

These results are shown in the independent error model, and also in the local stochastic error model that will be defined in Section 3.1 and is intermediate between independent and adversarial errors, and is particularly relevant in the context of fault tolerance. The proof of Theorem 1 can be found at the end of Section 4. We point out that the case of quantum CSS LDPC codes with

the (inefficient) maximum likelihood decoder was already studied in Ref. [19] for the independent error model and in Ref. [13] for local stochastic noise.

The family of quantum expander codes with the small-set-flip decoding algorithm thus satisfies the requirements of the construction of [13] for a fault-tolerant scheme with constant space overhead. However, to plug these codes in the construction, we need in addition to analyze errors on the syndrome measurements. We believe that such syndrome errors can be analyzed similarly as in [13], but a full analysis remains to be done.

The manuscript is structured as follows. In Section 2, we recall the constructions of classical and quantum expander codes and describe their respective decoding algorithms. In Section 3, we consider a generalized version of the small-set-flip decoding algorithm for quantum expander codes and introduce a notion of “locality” for decoding algorithms as well as associated parameters α and t . We then prove that the algorithm will correct any error E as long as all connected α -subsets are of size at most t . In Section 4, we study the notion of α -percolation with the tools of percolation theory and show that for $t = \omega(\log n)$, there exists a threshold below which a random set E will have connected α -subsets of size at most t with high probability. Finally, in Section 5, we show that the efficient small-set-flip decoding algorithm for quantum expander codes satisfies the locality condition necessary to apply our results.

2 Classical and Quantum expander codes

2.1 Classical expander codes

A linear classical error correcting code \mathcal{C} of dimension k and length n is a subspace of \mathbb{F}_2^n of dimension k . Mathematically, it can be defined as the kernel of an $(n - k) \times n$ matrix H , called the parity-check matrix of the code: $\mathcal{C} = \{x \in \mathbb{F}_2^n : Hx = 0\}$. The minimum distance d_{\min} of the code is the minimum Hamming weight of a nonzero codeword: $d_{\min} = \min\{|x| : x \in \mathcal{C}, x \neq 0\}$. Such a linear code is often denoted as $[n, k, d_{\min}]$. It is natural to consider families of codes, instead of single instances, and study the dependence between the parameters n, k and d_{\min} . In particular, a family of codes has *constant rate* if $k = \Theta(n)$. Another property of interest of a linear code is the weight of the rows and columns of the parity-check matrix H . If these weights are upper bounded by a constant, then we say that the code is a *low-density parity-check* (LDPC) code [11]. This property is particularly attractive because it allows for efficient decoding algorithms, based on message passing for instance.

An alternative description of a linear code is via a bipartite graph known as its *factor graph*. Let $G = (V \cup C, \mathcal{E})$ be a bipartite graph, with $|V| = n_V$ and $|C| = n_C$. With such a graph, we associate the $n_C \times n_V$ matrix H , whose rows are indexed by the vertices of C , whose columns are indexed by the vertices of V , and such that $H_{cv} = 1$ if v and c are adjacent in G and $H_{cv} = 0$ otherwise. The binary linear code \mathcal{C}_G associated with G is the code with parity-check matrix H . The graph G is the *factor graph* of the code \mathcal{C}_G , V is the set of *bits* and C is the set of *check-nodes*.

It will be convenient to describe codewords and error patterns as subsets of V : the binary word $e \in \mathbb{F}_2^{n_V}$ is described by a subset $E \subseteq V$ whose indicator vector is e . Similarly we define the *syndrome* of a binary word either as a binary vector of length n_C or as a subset of C :

$$\sigma(e) := He \in \mathbb{F}_2^{n_C}, \quad \sigma(E) := \bigoplus_{v \in E} \Gamma(v) \subseteq C.$$

In this paper, the operator \oplus is interpreted either as the symmetric difference of sets or as the bit-wise exclusive disjunction depending on whether errors and syndromes are interpreted as sets or as binary vectors.

A family of codes that will be central in this work are those associated to so-called *expander graphs*, that were first considered by Sipser and Spielman in [25].

Definition 2 (Expander graph). Let $G = (V \cup C, \mathcal{E})$ be a bipartite graph with left and right degrees bounded by d_V and d_C respectively. Let $|V| = n_V$ and $|C| = n_C$. We say that G is (γ_V, δ_V) -left-expanding for some constants $\gamma_V, \delta_V > 0$, if for any subset $S \subseteq V$ with $|S| \leq \gamma_V n_V$, the neighbourhood $\Gamma(S)$ of S in the graph G satisfies $|\Gamma(S)| \geq (1 - \delta_V)d_V|S|$. Similarly, we say that G is (γ_C, δ_C) -right-expanding if for any subset $S \subseteq C$ with $|S| \leq \gamma_C n_C$, we have $|\Gamma(S)| \geq (1 - \delta_C)d_C|S|$. Finally, the graph G is said $(\gamma_V, \delta_V, \gamma_C, \delta_C)$ -expanding if it is both (γ_V, δ_V) -left expanding and (γ_C, δ_C) -right expanding.

Sipser and Spielman introduced *expander codes*, which are the codes associated with (left)-expander graphs. Remarkably these codes come with an efficient decoding algorithm, that can correct *arbitrary* errors of weight $\Omega(n)$ [25].

Theorem 3 (Sipser, Spielman [25]). Let $G = (V \cup C, \mathcal{E})$ be a (γ, δ) -left expander graph with $\delta < 1/4$. There exists an efficient decoding algorithm for the associated code \mathcal{C}_G that corrects all error patterns $E \subseteq V$ such that $|E| \leq \gamma(1 - 2\delta)|V|$.

The decoding algorithm called the “bit-flip” algorithm is very simple: one simply cycles through the bits and flip them if this operation leads to a reduction of the syndrome weight. Sipser and Spielman showed that provided the expansion is sufficient, such an algorithm will always succeed in identifying the error if its weight is below $\gamma(1 - 2\delta)|V|$. In this paper, however, we will be interested in the decoding of *quantum* expander codes, that we will review next.

Before that, let us mention for completeness that although finding explicit constructions of highly-expanding graphs is a hard problem, such graphs can nevertheless be found efficiently by probabilistic techniques. Verifying that a given graph is expanding is a hard task, however.

Theorem 4 (Theorem 8.7 of [24]). Let δ_V, δ_C be positive constants. For integers $d_V > 1/\delta_V$ and $d_C > 1/\delta_C$, a graph $G = (V \cup C, \mathcal{E})$ with left-degree bounded by d_V and right-degree bounded by d_C chosen at random according to some distribution is $(\gamma_V, \delta_V, \gamma_C, \delta_C)$ -expanding for $\gamma_V, \gamma_C = \Omega(1)$ with high probability.

2.2 Quantum error correcting codes

A quantum code encoding k logical qubits into n physical qubits is a subspace of $(\mathbb{C}^2)^{\otimes n}$ of dimension 2^k . A quantum *stabilizer code* is described by a stabilizer, that is an Abelian group of n -qubit Pauli operators (tensor products of single-qubit Pauli operators X, Y, Z and I with an overall phase of ± 1 or $\pm i$) that does not contain $-I$. The code is defined as the eigenspace of the stabilizer with eigenvalue $+1$ ([12]). A stabilizer code of dimension k can be described by a set of $n - k$ generators of its stabilizer group.

A particularly nice construction of stabilizer codes is given by the CSS construction ([4], [26]), where the stabilizer generators are either products of single-qubit X Pauli matrices or products of Z Pauli matrices. The condition that the stabilizer group is Abelian therefore only needs to be enforced between X -type generators (corresponding to products of Pauli X operators) and Z -type generators. More precisely, consider two classical linear codes \mathcal{C}_X and \mathcal{C}_Z of length n satisfying $\mathcal{C}_Z^\perp \subseteq \mathcal{C}_X$, or equivalently, $\mathcal{C}_X^\perp \subseteq \mathcal{C}_Z$. (Here, \mathcal{C}_X^\perp is the dual code to \mathcal{C}_X consisting of the words which are orthogonal to all the words in \mathcal{C}_X .) This condition also reads $H_X \cdot H_Z^T = 0$, if H_X and H_Z denote the respective parity-check matrices of \mathcal{C}_X and \mathcal{C}_Z . The quantum code $CSS(\mathcal{C}_X, \mathcal{C}_Z)$ associated with \mathcal{C}_X (used to correct X -type errors and corresponding to Z -type stabilizer generators) and \mathcal{C}_Z (used to correct Z -type errors and corresponding to X -type stabilizer generators) has length n and is defined as the linear span of $\left\{ \sum_{z \in \mathcal{C}_Z^\perp} |x + z\rangle : x \in \mathcal{C}_X \right\}$, where $\{|x\rangle : x \in \mathbb{F}_2^n\}$ is the canonical basis of $(\mathbb{C}^2)^{\otimes n}$. In particular, two states differing by an element of the stabilizer group are equivalent. The dimension of the CSS code is given by $k =$

$\dim(\mathcal{C}_X/\mathcal{C}_Z^\perp) = \dim(\mathcal{C}_Z/\mathcal{C}_X^\perp) = \dim \mathcal{C}_X + \dim \mathcal{C}_Z - n$. Its minimum distance is defined in analogy with the classical case as the minimum number of single-qubit Pauli operators needed to map a codeword to an orthogonal one. For the code $CSS(\mathcal{C}_X, \mathcal{C}_Z)$, one has $d_{\min} = \min(d_X, d_Z)$ where $d_X = \min\{|E| : E \in \mathcal{C}_X \setminus \mathcal{C}_Z^\perp\}$ and $d_Z = \min\{|E| : E \in \mathcal{C}_Z \setminus \mathcal{C}_X^\perp\}$. We say that $CSS(\mathcal{C}_X, \mathcal{C}_Z)$ is a $[[n, k, d_{\min}]]$ quantum code. In the following, it will be convenient to consider the factor graph $G_X = (V \cup C_X, \mathcal{E}_X)$ (resp. G_Z) of \mathcal{C}_X (resp. of \mathcal{C}_Z). We will denote by Γ_X (resp. Γ_Z) the neighbourhood in G_X (resp. G_Z). For instance, if $g \in C_Z$ is an X -type generator, that is a product of Pauli X operators, then $\Gamma_Z(g)$ is the set of qubits (indexed by V) on which the generator acts non-trivially.

Among stabilizer codes, and CSS codes, the class of quantum LDPC codes stands out for practical reasons: these are the codes for which one can find *sparse* parity-check matrices H_X and H_Z . More precisely, such matrices are assumed to have constant row weight and constant column weight. Physically, this means that each generator of the stabilizer acts at most on a constant number of qubits, and that each qubit is acted upon by a constant number of generators. Note, however, that while surface codes exhibit in addition spatial locality in the sense that interactions only involve spatially close qubits (for an explicit layout of the qubits in Euclidean space), we do not require this for general LDPC codes. This means that generators might involve long-range interactions. This seems necessary in order to find constant rate quantum codes with growing minimum distance [3].

In this work, we will be concerned with Pauli-type noise, mapping a qubit ρ to $p_1\rho + p_X X\rho X + p_Y Y\rho Y + p_Z Z\rho Z$, for some p_1, p_X, p_Y, p_Z . Such a noise model is particularly convenient since one can interpret the action of the noise as applying a given Pauli error with some probability. As usual, it is sufficient to deal with both X and Z -type errors in order to correct Pauli-type errors, and one can therefore define an error by the locations of the Pauli X and Pauli Z errors. An *error pattern* is a pair (e_X, e_Z) of n -bit strings, which describe the locations of the Pauli X errors, and Pauli Z errors respectively. The syndrome associated with (e_X, e_Z) for the code $CSS(\mathcal{C}_X, \mathcal{C}_Z)$ consists of $\sigma_X = \sigma_X(e_X) := H_X e_X$ and $\sigma_Z = \sigma_Z(e_Z) := H_Z e_Z$. A decoder is given the pair (σ_X, σ_Z) of syndromes and should return a pair of errors (\hat{e}_X, \hat{e}_Z) such that $e_X + \hat{e}_X \in \mathcal{C}_Z^\perp$ and $e_Z + \hat{e}_Z \in \mathcal{C}_X^\perp$. In that case, the decoder outputs an error equivalent to (e_X, e_Z) , and we say that it succeeds.

Similarly as in the classical case, it will be convenient to describe X -type error patterns and X -type syndromes as subsets of the vertices of the factor graph $G_X = (V \cup C_X, \mathcal{E}_X)$. The error pattern is then described by a subset $E_X \subseteq V$ whose indicator vector is e_X and the syndrome is the subset $\sigma_X(E_X) \subseteq C_X$ defined by $\sigma_X(E_X) := \bigoplus_{v \in E_X} \Gamma_X(v)$. Z -type error patterns and Z -type syndromes are described in the same fashion using the factor graph G_Z .

In the rest of this paper, we will consider special decoding algorithms that try to recover e_X and e_Z independently. More precisely, a decoding algorithm is given by an X -decoding algorithm that takes as input σ_X and returns \hat{e}_X such that $\sigma_X(\hat{e}_X) = \sigma_X$, and a Z -decoding algorithm that takes as input σ_Z and returns \hat{e}_Z such that $\sigma_Z(\hat{e}_Z) = \sigma_Z$. We note that this special type of decoding algorithm might achieve sub-optimal error probabilities for some error models. In fact, if there are correlations between X and Z errors (for instance in the case of the depolarizing channel where $p_X = p_Y = p_Z$), one can decrease the error probability by trying to recover e_X by using both σ_X and σ_Z . However, for the purpose of this paper, it is sufficient to consider these special decoding algorithms. In addition, as we will consider codes where H_X and H_Z play symmetric roles, so the Z -decoding algorithm is obtained from the X -decoding algorithm by exchanging the roles of X and Z .

Let us conclude this section by mentioning the *maximum likelihood decoding algorithm* which

returns an error (\hat{e}_X, \hat{e}_Z) of minimum Hamming weight with the appropriate syndrome, that is:

$$\hat{e}_X = \arg \min_{\sigma_X(f_X)=\sigma_X} |f_X|, \quad \hat{e}_Z = \arg \min_{\sigma_Z(f_Z)=\sigma_Z} |f_Z|.$$

This inefficient algorithm always succeeds provided that the error weights satisfy $|e_X| \leq \lfloor (d_X - 1)/2 \rfloor$ and $|e_Z| \leq \lfloor (d_Z - 1)/2 \rfloor$.

2.3 Quantum expander codes

In this work, we are particularly interested in a family of LDPC CSS codes that features a constant rate and a minimum distance $\Theta(\sqrt{n})$ obtained by applying the hypergraph product construction of Tillich and Zémor to classical expander codes. If these expander codes have sufficient expansion, the corresponding quantum code is called *quantum expander code* and comes with an efficient decoding algorithm that corrects arbitrary errors of size linear in the minimum distance.

The construction is as follows. Let $G = (A \cup B, \mathcal{E})$ be a biregular $(\gamma_A, \delta_A, \gamma_B, \delta_B)$ -expanding graph with $\delta_A, \delta_B < 1/6$, and constant left and right degrees denoted d_A and d_B . Let us also denote $n_A = |A|$ and $n_B = |B|$ with $n_A < n_B$. According to Theorem 4, such graphs can be found efficiently (in a probabilistic fashion) provided that $d_A, d_B \geq 7$. Let \mathcal{C} be the classical code associated with G , let $d_{\min}(\mathcal{C})$ be the minimal distance of \mathcal{C} and let H be its parity-check matrix (that we assume to be full rank) corresponding to the factor graph G . In particular, the weights of rows and columns of H are d_A and d_B , respectively. The hypergraph product code of \mathcal{C} with itself admits the following parity check matrices:

$$\begin{aligned} H_X &= (I_{n_A} \otimes H, H^T \otimes I_{n_B}) \\ H_Z &= (H \otimes I_{n_A}, I_{n_B} \otimes H^T). \end{aligned}$$

It is immediate to check that this defines a legitimate CSS code since

$$\begin{aligned} H_X H_Z^T &= I_{n_A} \otimes H \cdot (H \otimes I_{n_A})^T + H^T \otimes I_{n_B} (I_{n_B} \otimes H^T)^T \\ &= H^T \otimes H + H^T \otimes H = 0. \end{aligned}$$

Moreover, the code is LDPC with generators of weight $d_A + d_B$ and qubits involved in at most $2 \max(d_A, d_B)$ generators.

We can describe the factor graphs G_X and G_Z as follows: the set of qubits is indexed by $A^2 \cup B^2$, the set of Z -type generators is indexed by $A \times B$ and the set of X -type generators is indexed by $B \times A$. The bipartite graph G_X has left vertices $A^2 \cup B^2$, right vertices $A \times B$ and there is an edge between a vertex $(\alpha, a) \in A^2$ (resp. $(b, \beta) \in B^2$) and a vertex $(\alpha, \beta) \in A \times B$ when a (resp. b) is in the neighbourhood of β (resp. α) in G . The bipartite graph G_Z has left vertices $A^2 \cup B^2$, right vertices $B \times A$ and there is an edge between a vertex $(\alpha, a) \in A^2$ (resp. $(b, \beta) \in B^2$) and a vertex $(b, a) \in B \times A$ when α (resp. β) is in the neighbourhood of b (resp. a) in G .

The following theorem summarizes the main properties of this quantum code.

Theorem 5 (Tillich, Zémor [27]). *The CSS code defined above is LDPC with parameters $[[n, k, d_{\min}]]$, where $n = n_A^2 + n_B^2$, $k \geq (n_A - n_B)^2$ and $d_{\min} = d_{\min}(\mathcal{C})$.*

Since the graph G is sufficiently expanding, we can apply the results from Ref. [20] and show the existence of an efficient decoding algorithm called “small-set-flip” decoding algorithm. Focusing on X -type errors for instance, and assuming that the syndrome $\sigma_X = H_X e_X$ is known, the algorithm cycles through all the X -type generators of the stabilizer group (i.e. the rows of H_Z), and for each one of them, determines whether there is an error pattern contained in the generator that decreases the syndrome weight. Assuming that this is the case, the algorithm applies the error

pattern (maximizing the ratio between the syndrome weight decrease and the pattern weight). The algorithm then proceeds by examining the next generator. Since the generators have (constant) weight $d_A + d_B$, there are $2^{d_A+d_B}$ possible patterns to examine for each generator. If the graph G has sufficient expansion and if the error weight is small enough, there always exists a generator containing an error pattern decreasing the syndrome weight. It can then be proved that if the error weight is below the value w_0 of Eq. (1) below, the decoding algorithm will not stop before reaching a null syndrome, hence corresponding to a codeword. Moreover, because the number of steps of the algorithm is sufficiently low, it is not possible to have reached an incorrect codeword, hence the decoding succeeded.

Let us introduce some additional notations: let \mathcal{X} be the set of subsets of V corresponding to X -type generators: $\mathcal{X} = \{\Gamma_Z(g) : g \in C_Z\} \subseteq \mathcal{P}(V)$. The indicator vectors of the elements of \mathcal{X} span the dual code C_Z^\perp . The condition for successful decoding then asks that there exists a subset $X \subset \mathcal{X}$ such that

$$E \oplus \hat{E} = \bigoplus_{x \in X} x,$$

meaning that the remaining error after decoding is trivial, that is equal to a sum of generators. At each step, the small-set-flip algorithm tries to flip a subset of some $g \in \mathcal{X}$. In other words, it tries to flip some element of $\mathcal{F} := \{F \subseteq x : x \in \mathcal{X}\}$.

Algorithm 1 (Ref. [20]): Small-set-flip decoding algorithm for quantum expander codes

INPUT: $\sigma \subseteq C_X$, a syndrome where $\sigma = \sigma_X(E)$ with $E \subseteq V$ an error

OUTPUT: $\hat{E} \subseteq V$, a guess for the error pattern (alternatively, a set of qubits to correct)

SUCCESS: if $E \oplus \hat{E} = \bigoplus_{x \in X} x$ for $X \subseteq \mathcal{X}$, i.e. E and \hat{E} are equivalent errors

$\hat{E}_0 = 0 ; \sigma_0 = \sigma ; i = 0$

while $(\exists F \in \mathcal{F} : |\sigma_i| - |\sigma_i \oplus \sigma_X(F)| > 0)$ **do**

$F_i = \arg \max_{F \in \mathcal{F}} \frac{|\sigma_i| - |\sigma_i \oplus \sigma_X(F)|}{|F|}$ // pick an arbitrary one if there are several choices

$\hat{E}_{i+1} = \hat{E}_i \oplus F_i$

$\sigma_{i+1} = \sigma_i \oplus \sigma_X(F_i)$ // $\sigma_{i+1} = \sigma_X(E \oplus \hat{E}_{i+1})$

$i = i + 1$

end while

return \hat{E}_i

Theorem 6 (Leverrier, Tillich, Zémor [20]). *Let $G = (A \cup B, \mathcal{E})$ be a (d_A, d_B) -biregular $(\gamma_A, \delta_A, \gamma_B, \delta_B)$ -expanding graph with $\delta_A, \delta_B < 1/6$. Letting d_A and d_B be fixed and allowing n_A, n_B to grow, Algorithm 1 runs in time linear in the code length $n = n_A^2 + n_B^2$, and decodes any quantum error pattern of weight less than*

$$w_0 = \frac{1}{3(1 + d_B)} \min(\gamma_A n_A, \gamma_B n_B). \quad (1)$$

The analysis above applies to arbitrary errors of weight less than w_0 . Unfortunately, $w_0 = O(\sqrt{n})$, corresponding to an error rate of $O(1/\sqrt{n})$, which is often insufficient for applications, where a constant error rate is typically required. The main goal of this work is to determine the performance of the algorithm above against *random noise* with constant error rate. Such an error model leads to typical errors of size linear in n , and Theorem 6 is useless in that regime. The rest of the paper is devoted to show the existence of a threshold for natural models of random noise for Algorithm 1.

3 Efficient decoding algorithms for quantum expander codes

In this section, we first describe two natural noise models for which we will establish the existence of thresholds for the decoding algorithm of quantum expander codes. Then, we will present a variant of the small-set-flip decoding algorithm of Ref. [20]. Finally, we will describe the crucial features of this decoding algorithm, notably its *locality*, and define parameters α and t , that will be particularly useful to establish the existence of thresholds.

3.1 Noise models

We think of an error pattern as a pair (E_X, E_Z) of subsets of the set of qubits V . A noise model is thus described by a distribution on pairs of subsets of V . Letting (E_X, E_Z) be distributed according to such a model, the failure probability of a decoding algorithm that outputs (\hat{E}_X, \hat{E}_Z) is given by

$$\begin{aligned} & \mathbb{P}((E_X, E_Z) \text{ not equivalent to } (\hat{E}_X, \hat{E}_Z)) \\ & \leq \mathbb{P}(E_X \text{ not equivalent to } \hat{E}_X) + \mathbb{P}(E_Z \text{ not equivalent to } \hat{E}_Z). \end{aligned} \quad (2)$$

As the decoders we consider treat X and Z symmetrically, it suffices to focus on X -type errors and the analysis of failure probability for identifying Z -type errors is exactly symmetrical. For this reason, we will now assume that we deal with X -type errors and simply use E for E_X .

The most natural error model is that of independent noise, where each qubit is in error independently, with the same probability.

Definition 7 (Independent noise error model).

Let V be the set of qubits. The error of parameter p is a random variable $E \subseteq V$ such that $\mathbb{P}(E) = p^{|E|}(1-p)^{|V|-|E|}$.

For example, the X -type errors (resp. Z -type errors) produced by a depolarizing channel $\rho \mapsto (1-p)\rho + p/3(X\rho X + Y\rho Y + Z\rho Z)$ are independent with parameter $2p/3$ since $Y = iXZ = -iZX$. More generally, a channel $\rho \mapsto p_I\rho + p_X X\rho X + p_Y Y\rho Y + p_Z Z\rho Z$ satisfies the independent noise condition with parameter $p_X + p_Y$ for the X -type errors and parameter $p_Y + p_Z$ for the Z -type errors.

The independent noise error model is often too restrictive in practice, notably in the context of fault-tolerance since errors will propagate through the circuit and become correlated. Instead of trying to control such correlations, it is handy to simply allow for any possible error set, but put a bound on the probability of observing an error of a given size. A convenient model was considered by Gottesman in Ref. [13] where we ask that such a probability is exponentially small in the error size.

Definition 8 (Local stochastic error model).

Let V be the set of qubits. The error of parameter p is a random variable $E \subseteq V$ such that for all $F \subseteq V$: $\mathbb{P}(F \subseteq E) \leq p^{|F|}$. In other words, the location of the errors is arbitrary but the probability of a given error decays exponentially with its weight.

3.2 Small-set-flip decoding algorithm for quantum expander codes

Let us now turn to the decoding of quantum codes. As mentioned previously, we focus here on decoding X errors. Given a syndrome $\sigma_X = \sigma_X(E_X)$, the decoding algorithm should output a guess \hat{E}_X which is equivalent to E_X , in the sense that $E_X + \hat{E}_X$ belongs to the dual code \mathcal{C}_Z^\perp .

In the description of Algorithm 1, the idea is to go through all the elements of \mathcal{C}_Z (called X -type generators and corresponding to products of Pauli X operators), one at a time, and check

whether applying any error pattern within this generator would decrease the syndrome weight. If this is the case, then apply the “best” such error pattern. Then proceed with the next generator. It is possible to design Algorithm 2, a slight generalization of Algorithm 1 that depends on an extra-parameter $\beta \in (0, 1]$, where a small set F is flipped only if it leads to a decrease of the syndrome weight by at least $\beta d_B |F|$. Note that Algorithm 2 is a “tool” that we use in order to study Algorithm 1 since any error $E \subseteq V$ corrected by Algorithm 2 is corrected by Algorithm 1 (see Remark 9). Requiring a larger value for β reduces the number of qubits incorrectly flipped during the decoding procedure. This property will be useful for studying the performance of the algorithm against various random noise models. The drawback, however, is that larger values of β mean better expansion (smaller value of δ_A and δ_B) and therefore larger degrees d_A and d_B . The value of β should be optimized in order to find the largest possible threshold p_0 under which the small-set-flip decoding algorithm will correct random errors, except with negligible probability.

Algorithm 2 : Small-set-flip decoding algorithm for quantum expander codes, with parameter $\beta \in (0, 1]$

INPUT: $\sigma \subseteq C_X$, a syndrome where $\sigma = \sigma_X(E)$ with $E \subseteq V$ an error

OUTPUT: $\hat{E} \subseteq V$, a guess for the error pattern (alternatively, a set of qubits to correct)

SUCCESS: if $E \oplus \hat{E} = \bigoplus_{x \in X} x$ for $X \subseteq \mathcal{X}$, i.e. E and \hat{E} are equivalent errors

$\hat{E}_0 = 0 ; \sigma_0 = \sigma ; i = 0$

while $(\exists F \in \mathcal{F} : |\sigma_i| - |\sigma_i \oplus \sigma_X(F)| \geq \beta d_B |F|)$ **do**

$F_i = \arg \max_{F \in \mathcal{F}} \frac{|\sigma_i| - |\sigma_i \oplus \sigma_X(F)|}{|F|}$ // pick an arbitrary one if there are several choices

$\hat{E}_{i+1} = \hat{E}_i \oplus F_i$

$\sigma_{i+1} = \sigma_i \oplus \sigma_X(F_i)$ // $\sigma_{i+1} = \sigma_X(E \oplus \hat{E}_{i+1})$

$i = i + 1$

end while

return \hat{E}_i

Remark 9. The while loop condition in Algorithm 2 is stronger than in Algorithm 1 and this is the only difference between the two algorithms. As a consequence if Algorithm 2 corrects an error E , then Algorithm 1 corrects E .

Analyzing this modified algorithm can be done similarly as for the original one. For a given quantum expander code, we prove that there exists $\beta_0 > 0$ such that for $\beta \leq \beta_0$, the small-set-flip decoding algorithm with parameter β corrects any adversarial errors of size up to $\Theta(\sqrt{n})$. The value of β_0 is defined in Definition 10.

Definition 10. Let $G = (A \cup B, \mathcal{E})$ be a (d_A, d_B) -biregular $(\gamma_A, \delta_A, \gamma_B, \delta_B)$ -left-right-expanding graph. We define r and β_0 by:

$$r = \frac{d_A}{d_B}, \quad \beta_0 = \frac{r}{2} [1 - 4(\delta_A + \delta_B + (\delta_B - \delta_A)^2)].$$

Note that $\delta_A, \delta_B < 1/8$ is sufficient to ensure that $\beta_0 > 0$ and there exists such expander graphs as soon as $d_A, d_B \geq 9$ (see Theorem 4). As a reminder, the weight of the stabilizer generators is $d_A + d_B$.

Proposition 11. Let $G = (A \cup B, \mathcal{E})$ be a (d_A, d_B) -biregular $(\gamma_A, \delta_A, \gamma_B, \delta_B)$ -left-right-expanding graph and $r, \beta = \beta_0$ be defined using the associated quantum expander code (Definition 10). The

small-set-flip decoding algorithm of parameter β described in Algorithm 2 can correct any adversarial error of size up to $t_{\text{SSF}(\beta)}$ where:

$$t_{\text{SSF}(\beta)} \geq \frac{r\beta}{1+\beta} \min(\gamma_A n_A, \gamma_B n_B).$$

The proof of Proposition 11 is provided in Appendix B.1. Note that Algorithm 2 is not deterministic since the F_i are not necessarily unique but if the weight of the error is smaller than $t_{\text{SSF}(\beta)}$, then the error is corrected for any non-deterministic choice of the F_i .

3.3 Locality of the small-set-flip decoding algorithm

This section is devoted to explicitly state the properties of the small-set-flip decoding algorithm that we need in order to prove Theorem 1. Recall that our goal is to establish the existence of a threshold: in the random error model of Definition 7, if the probability of error is small enough then we correct the error with high probability. An important property of the algorithm that we consider is that it can correct adversarial errors of size $\leq t$ where $t = \Theta(\sqrt{|V|})$ and V is the set of qubits. The second crucial property we will rely on is that the algorithm is local, meaning in words that errors far away in the *adjacency graph* \mathcal{G} of the code will not interact during the decoding process. The adjacency graph $\mathcal{G} = (V, \mathcal{E})$ associated with the code admits the set of qubits as vertices, and two vertices are adjacent in \mathcal{G} if the corresponding qubits share an X -type or Z -type stabilizer generator.

Let us now define the notion of locality more formally. For an input syndrome $\sigma_X(E) \subseteq C_X$ with $E \subseteq V$ an error, the output of Algorithm 2 is $\hat{E} = F_0 \oplus \dots \oplus F_{f-1}$, where F_i is the small set flipped at the i^{th} round of the algorithm, and we will call $U = E \cup F_0 \cup \dots \cup F_{f-1}$ the support. We have $\hat{E} \subseteq U$ but $\hat{E} \neq U$ in general.

Proposition 12 (Locality of the small-set-flip decoding algorithm). *Let $E \subseteq V$ be an error and run Algorithm 2 on input $\sigma_X(E)$. Consider K , a connected component in \mathcal{G} of U where $U = E \cup F_0 \cup \dots \cup F_{f-1}$ is the support and $\hat{E} = F_0 \oplus \dots \oplus F_{f-1}$ is the output.*

Then there is a valid execution of Algorithm 2 with input $\sigma_X(E \cap K)$, output $\hat{E} \cap K$ and support K .

Let's give the intuition of the proof of Proposition 12 (the formal proof is given in Section 5). The execution of Algorithm 2 on the input syndrome $\sigma_X(E)$ provides some sets F_0, \dots, F_{f-1} that we use to define the support $U = E \cup F_0 \cup \dots \cup F_{f-1}$ and the output $\hat{E} = F_0 \oplus \dots \oplus F_{f-1}$. We would like to prove that there is a valid execution of the algorithm with input $\sigma_X(E \cap K)$, output $\hat{E} \cap K$ and support K where K is some connected component of U in \mathcal{G} . In other words, we would like to prove that on the input syndrome $\sigma_X(E \cap K)$, Algorithm 2 provides some sets $F'_0, \dots, F'_{f'-1}$ (as well as $\sigma'_0, \dots, \sigma'_{f'-1}$) which satisfy $K = (E \cap K) \cup F'_0 \cup \dots \cup F'_{f'-1}$ and $\hat{E} \cap K = F'_0 \oplus \dots \oplus F'_{f'-1}$.

For simplicity, we first assume that $F_i \subseteq K$ for all i and we define $C_K = \Gamma_X(K) \subseteq C_X$ to be the set of Z -type generators which touch at least one qubit of K in the factor graph G_X . Let's prove that a valid choice for f' , F'_i and σ'_i is given by $f' = f$, $F'_i = F_i$ and $\sigma'_i = \sigma_i \cap C_K$. In the factor graph G_X , all the neighbours of C_K are in $K \cup \bar{U}$ and all the neighbours of \bar{C}_K are in \bar{K} . Hence, the syndrome part included in C_K is the same for an error $W \subseteq V$ and for the error $W \cap K$: $\sigma_X(W \cap K) = \sigma_X(W) \cap C_K$ (Lemma 20). If we apply this formula with $W = E$, we get $\sigma'_0 = \sigma_X(E \cap K)$ which matches the input of Algorithm 2 in the conclusion of Proposition 12. With $W = F_i$ and our additional hypothesis $F_i \subseteq K$, we get $\sigma'_{i+1} = \sigma'_i \oplus \sigma_X(F_i)$ which matches the manner in which σ_i is updated in Algorithm 2. On the other hand, the value of the syndrome σ_i determines the algorithm behaviour and, similarly, the value of $\sigma'_i = \sigma_i \cap C_K$

determines the algorithm behaviour on the qubits of K (this is a consequence of Lemma 21). As a conclusion, flipping the sets F_i is a valid execution for Algorithm 2 on input $\sigma_X(E \cap K)$ and thus Proposition 12 holds.

In the general case, we have some indices i satisfying $F_i \not\subseteq K$. Since K is a connected component of U in \mathcal{G} , such i always satisfy $F_i \cap K = \emptyset$ and thus flipping F_i does not change the syndrome part included in C_K . Hence, with the sets F'_i defined to be the sets F_i such that $F_i \subseteq K$, we can prove Proposition 12 as previously.

The concept of locality allows us to treat the errors piece-wise in the sense that if Algorithm 2 corrects $E \cap K$ for any connected component K of U , then the entire error E is corrected. On the other hand, we know that Algorithm 2 corrects any error of size smaller than some $t_{\text{SSF}(\beta)} = \Theta(\sqrt{n})$. We do not have $|E| \leq t_{\text{SSF}(\beta)}$ in general, but using the concept of α -subset, we will prove that with high probability $|E \cap K| \leq |K| \leq t_{\text{SSF}(\beta)}$. Hence Algorithm 2 corrects $E \cap K$ and, by locality, E is corrected.

Definition 13 (α -subsets, $\text{MaxConn}_\alpha(E)$).

An α -subset of a set $E \subseteq V$ is a set $X \subseteq V$ such that $|X \cap E| \geq \alpha|X|$. We denote by $\text{MaxConn}_\alpha(E)$ the maximum size of a connected α -subset of E .

Note that a 1-subset is a subset in the usual sense and $\text{MaxConn}_1(E)$ is the size of the largest connected subset of E . As a relevant example, a corollary of Proposition 12 and of Lemma 14 is that the set K defined in Proposition 12 is a connected $\frac{\beta}{1+\beta}$ -subset of $E \cap K$.

Lemma 14. Let $E \subseteq V$ be an error and run the small-set-flip decoding algorithm of parameter $\beta > 0$ on input $\sigma_X(E)$. Let $\alpha = \frac{\beta}{1+\beta}$ and $U = E \cup F_0 \cup \dots \cup F_{f-1}$ be the support. Then U is an α -subset of E .

Proof. For all i , the decrease in the syndrome weight $|s_i| - |s_{i+1}|$ satisfies $|s_i| - |s_{i+1}| \geq \beta d_B |F_i|$ and thus $d_B |E| \geq |s_0| \geq |s_0| - |s_f| \geq \beta d_B \sum_i |F_i|$. We conclude that:

$$|U| = \left| E \cup \left(\bigcup_{i=0}^{f-1} F_i \right) \right| \leq |E| + \sum_{i=0}^{f-1} |F_i| \leq \frac{1+\beta}{\beta} |E| = \frac{1+\beta}{\beta} |E \cap U|.$$

□

All these notions allow us to prove the following general characterization of correctable errors.

Proposition 15. Suppose that for some quantum expander code, Algorithm 2 with parameter $\beta > 0$ corrects any error $E \subseteq V$ satisfying $|E| \leq t$.

Then any error $E \subseteq V$ satisfying $\text{MaxConn}_\alpha(E) \leq t$ with $\alpha = \frac{\beta}{1+\beta}$ is corrected.

Proof. Our goal is to prove that when we apply Algorithm 2 on the input $\sigma_X(E)$, the error E and the output $\hat{E} = F_0 \oplus \dots \oplus F_{f-1}$ are equivalent. Let $U = E \cup F_0 \cup \dots \cup F_{f-1}$ be the support, then we can decompose E as $E = \biguplus_K (E \cap K)$ and \hat{E} as $\hat{E} = \biguplus_K (\hat{E} \cap K)$, where the index K of these disjoint unions goes through the set of connected components of U . In order to prove that E and \hat{E} are equivalent, we will prove the sufficient condition that $E \cap K$ and $\hat{E} \cap K$ are equivalent for any K . By Proposition 12, there is a valid execution of Algorithm 2 with input $\sigma_X(E \cap K)$, output $\hat{E} \cap K$ and support K . By Lemma 14, K is an α -subset of $E \cap K$ and thus K is a connected α -subset of E . If we suppose $\text{MaxConn}_\alpha(E) \leq t$ then $|E \cap K| \leq |K| \leq t$ and we can conclude that $E \cap K$ and $\hat{E} \cap K$ are equivalent because Algorithm 2 corrects $E \cap K$. □

Remark 16. A similar analysis has been done in [19] and [13] for the maximum likelihood decoding algorithm of quantum LDPC codes and we could also apply the same arguments for the bit-flip algorithm of classical expander codes. However, this is less relevant since this algorithm can already correct adversarial errors of linear size. In fact, the maximum likelihood decoding algorithm and the bit-flip algorithm are both local in the sense of Proposition 12.

4 Properties of random errors: α -percolation

In order to establish Theorem 1 at the end of this section, we first study a specific model of site percolation that is relevant for the analysis of our decoding algorithms. More specifically, for a graph $\mathcal{G} = (V, \mathcal{E})$ with degree upper bounded by some constant d , and a probabilistic model for choosing a random subset $E \subseteq V$, site percolation asks about the size of the largest connected subset of vertices of E . Here, instead of looking for a connected subset of E , we instead ask about the size $\text{MaxConn}_\alpha(E)$ of the largest α -subset of E which is connected.

The simplest model for the choice of E is the i.i.d. model where each site (or vertex) is occupied independently with probability p . In this model, a standard result in percolation theory is that for the infinite d -regular tree, if $p > \frac{1}{d-1}$ there is an infinite connected subset of E and if $p < \frac{1}{d-1}$ all connected subsets are finite [22]. In the setting of finite graphs, for a random d -regular graph the maximum connected subset of E is of size $O(\log n)$ for $p < \frac{1}{d-1}$ and of size $\Omega(n)$ for $p > \frac{1}{d-1}$ [16]. For general graphs with degree bounded by d , [19] showed that for $p < \frac{1}{d-1}$ the size of the largest connected component of E in \mathcal{G} has size $O(\log |V|)$. We will also consider the more general *local stochastic model* where the probability of a subset E is upper bounded by $p^{|E|}$.

Theorem 17 (α -percolation). *Let $\mathcal{G} = (V, \mathcal{E})$ a graph with degree upper bounded by d . Let $\alpha \in (0, 1]$ and let $t \geq 1$ be an integer. Let*

$$p_{\text{ls}} = \left(\frac{2^{-h(\alpha)}}{(d-1)(1 + \frac{1}{d-2})^{d-2}} \right)^{\frac{1}{\alpha}}, \quad (3)$$

where $h(\alpha) = -\alpha \log_2 \alpha - (1-\alpha) \log_2 (1-\alpha)$ is the binary entropy function. Then, for an error random variable E satisfying the local stochastic noise property (Definition 8) with parameter $p < p_{\text{ls}}$, we have

$$\mathbb{P}[\text{MaxConn}_\alpha(E) \geq t] \leq C|V| \left(\frac{p}{p_{\text{ls}}} \right)^{\alpha t} \quad (4)$$

where $1/C = (1 - 2^{h(\alpha)/\alpha} p) \left(1 - \left(\frac{p}{p_{\text{ls}}} \right)^\alpha \right)$. In the special case, where vertices are chosen independently with probability $p < \frac{\alpha}{d-1}$, the result can be slightly improved to

$$\mathbb{P}[\text{MaxConn}_\alpha(E) \geq t] \leq |V| \left(\frac{d-1}{d-2} \right)^2 \frac{q^t}{1-q}, \quad (5)$$

where $q = (1-p)^{d-1-\alpha} p^\alpha 2^{h(\alpha)} (d-1)(1 + \frac{1}{d-2})^{d-2}$.

Note that (5) shows that if p is chosen small enough so that $q < 1$ the size of the largest component will be $O(\log |V|)$ with high probability. It is simple to see that choosing $p < p_{\text{ls}}$ leads to $q < 1$ but one can even take p slightly larger. For example, in the case $\alpha = 1$, we have $q < 1$ as soon as $p < \frac{1}{d-1}$ even though $p_{\text{ls}} = \frac{1}{(d-1)(1 + \frac{1}{d-2})^{d-2}}$. More generally, q as a function of p is increasing in the interval $p \in [0, \frac{\alpha}{d-1}]$ and thus there is a value $p_{\text{iid}} \in [p_{\text{ls}}, \frac{\alpha}{d-1}]$ such that $q = 1$ (see Lemma 29) and any $p < p_{\text{iid}}$ leads to a $q < 1$.

Proof. We focus here on the local stochastic noise bound. The finer analysis of the independent noise model in (5) is left to Appendix B.2.

Let $\mathcal{C}_s(\mathcal{G})$ be the set of connect sets of vertices of size s in \mathcal{G} . Applying a union bound, we obtain

$$\begin{aligned}\mathbb{P}[\text{MaxConn}_\alpha(E) \geq t] &= \mathbb{P}[\exists s \geq t, \exists X \in \mathcal{C}_s(\mathcal{G}) : |X \cap E| \geq \alpha|X|] \\ &\leq \sum_{s \geq t} \sum_{X \in \mathcal{C}_s(\mathcal{G})} \mathbb{P}[|X \cap E| \geq \alpha|X|].\end{aligned}$$

Let us first consider the quantity $\mathbb{P}[|X \cap E| \geq \alpha|X|]$:

$$\begin{aligned}\mathbb{P}[|X \cap E| \geq \alpha|X|] &\leq \sum_{m \geq \alpha s} \sum_{\substack{Y \subseteq X: \\ |Y|=m}} \mathbb{P}[X \cap E = Y] \\ &\leq \sum_{m \geq \alpha s} \sum_{\substack{Y \subseteq X: \\ |Y|=m}} \mathbb{P}[Y \subseteq E] \\ &\leq \sum_{m \geq \alpha s} \sum_{\substack{Y \subseteq X: \\ |Y|=m}} p^m \quad \text{from the definition of the noise model} \\ &\leq \sum_{m \geq \alpha s} \binom{s}{m} p^m.\end{aligned}$$

Using Lemma 25, we get

$$\begin{aligned}\mathbb{P}[|X \cap E| \geq \alpha|X|] &\leq \sum_{m \geq \alpha s} \left(2^{\frac{s}{m} h\left(\frac{m}{s}\right)} p\right)^m \\ &\leq \sum_{m \geq \alpha s} \left(2^{h(\alpha)/\alpha} p\right)^m \quad \text{since } (x \mapsto h(x)/x) \text{ is non increasing on } [\alpha, 1] \\ &\leq \frac{(2^{h(\alpha)/\alpha} p)^{\alpha s}}{1 - 2^{h(\alpha)/\alpha} p}.\end{aligned}$$

We now need an upper bound on the number of connected sets of size s in a graph with degree upper bounded by d . For this we use Corollary 28 saying that $|\mathcal{C}_s(\mathcal{G})| \leq |V|K(d)^s$ with $K(d) = (d-2)\left(1 + \frac{1}{d-2}\right)^{d-2}$. Then, we get

$$\begin{aligned}\mathbb{P}[\text{MaxConn}_\alpha(E) \geq t] &\leq \sum_{s \geq t} \sum_{X \in \mathcal{C}_s(\mathcal{G})} \mathbb{P}[|X \cap E| \geq \alpha|X|] \\ &\leq \frac{1}{1 - 2^{h(\alpha)/\alpha} p} \sum_{s \geq t} |\mathcal{C}_s(\mathcal{G})| 2^{h(\alpha)s} p^{\alpha s} \quad \text{by the union bound} \\ &\leq \frac{|V|}{1 - 2^{h(\alpha)/\alpha} p} \sum_{s \geq t} \left(K(d) 2^{h(\alpha)} p^\alpha\right)^s \\ &\leq \frac{|V|}{1 - 2^{h(\alpha)/\alpha} p} \frac{(K(d) 2^{h(\alpha)} p^\alpha)^t}{1 - K(d) 2^{h(\alpha)} p^\alpha}.\end{aligned}$$

Observing that $p_{\text{ls}} = \left(\frac{1}{K(d) 2^{h(\alpha)}}\right)^{1/\alpha}$, we obtain the desired result. \square

Using Propositions 11 and 15 and Theorem 17, we are now ready to prove Theorem 1:

Proof of Theorem 1. Using Remark 9, it is sufficient to prove Theorem 1 for Algorithm 2 instead of Algorithm 1. Under the condition that the expansion of the initial graph G is good enough, the parameter $\beta = \beta_0$ from Definition 10 is positive. We let $p_0 = p_{ls}$ as defined in (3) where \mathcal{G} is the adjacency graph of the quantum expander code and $d = d_B^2 + 2d_B(d_A - 1)$ is an upper bound on the degree of \mathcal{G} . Then for $p < p_0$, for an error E chosen according to a local stochastic noise, and fixing $t = t_{SSF(\beta)} = \Omega(\sqrt{n})$ as defined in Proposition 11, we obtain

$$\mathbb{P}[\text{MaxConn}_\alpha(E) \geq t] \leq Cn \left(\frac{p}{p_0} \right)^{\alpha t}.$$

As a result, applying Proposition 15, E is corrected except with probability at most $Cn \left(\frac{p}{p_0} \right)^{\alpha t}$ and this gives us the desired results. \square

Remark 18. *It is an interesting question of percolation theory to find the maximum value of p for which the largest connected α -subset is of size $O(\log n)$. This will depend on the graph \mathcal{G} but as mentioned earlier, it is known that for random d -regular graphs and $\alpha = 1$, $p = 1/(d - 1)$ is the threshold for the appearance of large connected sets (see [16], Theorem 3.5). In Proposition 30, we show that for $p > \frac{1}{(d-1)^\alpha}$, the largest connected α -subset is of size $\Omega(n^{1-\epsilon})$ and thus Eq. (3) can be improved by a factor of at most $(\frac{2^{-h(\alpha)}}{e})^{\frac{1}{\alpha}}$.*

5 The small-set-flip decoding algorithm is local (proof of Proposition 12)

In this section we show that the small-set-flip decoding algorithm (Algorithm 2) is local in the sense of Proposition 12 (the intuition of the proof has been given after the statement of the proposition).

Notation 19. *Throughout this section we keep the following notations:*

- The sets $\mathcal{F}, E, K, U, F_0, \dots, F_{f-1}$ and \hat{E} are defined as in Algorithm 2 and proposition 12.
- $C_K = \Gamma_X(K) \subseteq C_X$ is the set of Z -type generators which touch at least one qubit of K in the factor graph G_X .
- For $\sigma \subseteq C_X$ and $F \in \mathcal{F}$: $\Delta(\sigma, F) = \frac{|\sigma| - |\sigma \oplus \sigma_X(F)|}{|F|}$ is the function optimized by Algorithm 2.

Lemma 20. *Let $W \subseteq U$ (later we will take either $W = E$ or $W = F_i$) then $\sigma_X(W \cap K) = \sigma_X(W) \cap C_K$.*

Proof. We decompose $\sigma_X(W) \cap C_K$ using the partition $W = (W \cap K) \uplus (W \cap \overline{K})$ and the distributivity:

$$\begin{aligned} \sigma_X(W) \cap C_K &= [\sigma_X(W \cap K) \oplus \sigma_X(W \cap \overline{K})] \cap C_K \\ &= (\sigma_X(W \cap K) \cap C_K) \oplus (\sigma_X(W \cap \overline{K}) \cap C_K). \end{aligned}$$

The first term of this sum satisfies $\sigma_X(W \cap K) \cap C_K = \sigma_X(W \cap K)$ because $\sigma_X(W \cap K) \subseteq \Gamma_X(K) = C_K$. On the other hand, since K is a connected component of U in \mathcal{G} , it holds that $\Gamma_X(U \setminus K) \cap \Gamma_X(K) = \Gamma_X(U \setminus K) \cap C_K = \emptyset$ and thus the second term of the sum satisfies $\sigma_X(W \cap \overline{K}) \cap C_K = \emptyset$ because $W \cap \overline{K} \subseteq U \setminus K$. \square

Lemma 21. Let $F \in \mathcal{F}$ then:

- (i) $\Delta(\sigma \cap C_K, F) \leq \Delta(\sigma, F)$,
- (ii) If $F \subseteq K$ then $\Delta(\sigma \cap C_K, F) = \Delta(\sigma, F)$.

Proof. Rewrite $\Delta(\sigma, F)$ in a more friendly way using the definition of \oplus :

$$\Delta(\sigma, F) = \frac{2|\sigma_X(F) \cap \sigma| - |\sigma_X(F)|}{|F|}. \quad (6)$$

The proof of item (i) is straightforward using eq. (6): let $F \in \mathcal{F}$, we have:

$$\begin{aligned} \Delta(\sigma \cap C_K, F) &= \frac{2|\sigma_X(F) \cap \sigma \cap C_K| - |\sigma_X(F)|}{|F|} && \text{by eq. (6)} \\ &\leq \frac{2|\sigma_X(F) \cap \sigma| - |\sigma_X(F)|}{|F|} \\ &= \Delta(\sigma, F) && \text{by eq. (6)} \end{aligned}$$

Under the assumption $F \subseteq K$ of item (ii), we have $\sigma_X(F) \cap C_K = \sigma_X(F)$. As a consequence, the inequality in the proof of item (i) is an equality and thus item (ii) holds. \square

We are now ready to prove Proposition 12.

Proof of Prop. 12. We would like to prove that there is a valid execution of Algorithm 2 with input $\sigma_X(E \cap K)$, output $\hat{E} \cap K$ and support K . This means that there exists some sets $F'_0, \dots, F'_{f'-1} \subseteq V$ such that if we define $\sigma'_0 = \sigma_X(E \cap K)$ and $\sigma'_{j+1} = \sigma'_j \oplus \sigma'_X(F'_j)$ for $j \in \llbracket 0; f' - 1 \rrbracket$ then we have the following properties:

- (i) The set $\hat{E} \cap K$ satisfies $\hat{E} \cap K = F'_0 \oplus \dots \oplus F'_{f'-1}$.
- (ii) The set K satisfies $K = (E \cap K) \cup F'_0 \cup \dots \cup F'_{f'-1}$.
- (iii) For $j \in \llbracket 0; f' - 1 \rrbracket$, F'_j maximises $F \mapsto \Delta(\sigma'_j, F)$ over \mathcal{F} .
- (iv) For $j \in \llbracket 0; f' - 1 \rrbracket$, $\Delta(\sigma'_j, F'_j) \geq \beta d_B$.
- (v) For all $F \in \mathcal{F}$, $\Delta(\sigma'_{f'}, F) < \beta d_B$.

In the adjacency graph \mathcal{G} , two qubits in the support of the same generator are connected. As a consequence, since the set F_i is included in the support of a generator and since K is connected, either $F_i \subseteq K$ or $F_i \cap K = \emptyset$.

Consider $i_0 < \dots < i_{f'-1} \in \llbracket 0; f - 1 \rrbracket$ the steps of the algorithm such that $F_{i_k} \subseteq K$ and let's prove that the sets $F'_0 = F_{i_0}, \dots, F'_{f'} = F_{i_{f'-1}}$ are appropriate. By definition we have:

$$F_i \cap K = F_i \quad \text{if } i \in \{i_0, \dots, i_{f'-1}\} \quad (7)$$

$$F_i \cap K = \emptyset \quad \text{if } i \notin \{i_0, \dots, i_{f'-1}\} \quad (8)$$

Using eqs. (7) and (8), we can easily prove items (i) and (ii):

$$(i) \quad \hat{E} \cap K = \left(\bigoplus_{i=0}^{f-1} F_i \right) \cap K = \bigoplus_{i=0}^{f-1} (F_i \cap K) = \bigoplus_{j=0}^{f'-1} F'_j.$$

$$(ii) \quad \bigcup_{j=0}^{f'-1} F'_j = \bigcup_{i=0}^{f-1} (F_i \cap K) \text{ and using the distributivity:}$$

$$K = U \cap K = (E \cap K) \cup F'_0 \cup \dots \cup F'_{f'-1}.$$

By hypothesis, we further have a valid execution of Algorithm 2 with input $\sigma_X(E)$, output \hat{E} and support U . This implies:

(iii') For $i \in \llbracket 0; f-1 \rrbracket$, F_i maximises $F \mapsto \Delta(\sigma_i, F)$ over \mathcal{F} .

(iv') For $i \in \llbracket 0; f-1 \rrbracket$, $\Delta(\sigma_i, F_i) \geq \beta d_B$.

(v') For all $F \in \mathcal{F}$, $\Delta(\sigma_f, F) < \beta d_B$.

We also need the property of eq. (9) where we have defined $i_{f'} = f$:

$$\sigma'_j = \sigma_{i_j} \cap D_K \quad j \in \llbracket 0; f' \rrbracket. \quad (9)$$

Note that item (iii) (resp. item (iv), resp. item (v)) is a direct consequence of Lemma 21, eq. (9) and item (iii') (resp. item (iv'), resp. item (v')).

To complete the proof, it remains to prove eq. (9). By Lemma 20:

$$\begin{aligned} \sigma_0 \cap D_K &= \sigma_X(E) \cap D_K = \sigma_X(E \cap K), \\ \sigma_{i+1} \cap D_K &= (\sigma_i \oplus \sigma_X(F_i)) \cap D_K = (\sigma_i \cap D_K) \oplus \sigma_X(F_i \cap K) \quad i \in \llbracket 0; f-1 \rrbracket. \end{aligned}$$

If $i \in \{i_0, \dots, i_{f'-1}\}$ then by eq. (7): $\sigma_{i+1} \cap D_K = (\sigma_i \cap D_K) \oplus \sigma_X(F_i)$.

If $i \notin \{i_0, \dots, i_{f'-1}\}$ then by eq. (8): $\sigma_{i+1} \cap D_K = \sigma_i \cap D_K$.

In summary:

$$\begin{cases} \sigma'_0 = \sigma_X(E \cap K) \\ \sigma'_{j+1} = \sigma'_j \oplus \sigma_X(F_{i_j}) \quad j \in \llbracket 0; f'-1 \rrbracket \end{cases} \quad \begin{cases} \sigma_0 \cap D_K = \sigma_X(E \cap K) \\ \sigma_{i+1} \cap D_K = (\sigma_i \cap D_K) \oplus \sigma_X(F_i) & i \in \{i_0, \dots, i_{f'-1}\} \\ \sigma_{i+1} \cap D_K = \sigma_i \cap D_K & i \notin \{i_0, \dots, i_{f'-1}\} \end{cases}$$

Equation (9) is a direct corollary of the following property that we can prove by induction:

$$\begin{aligned} \sigma'_0 &= \sigma_0 \cap D_K = \dots = \sigma_{i_0} \cap D_K, \\ \sigma'_{j+1} &= \sigma_{i_{j+1}} \cap D_K = \dots = \sigma_{i_{j+1}} \cap D_K \quad j \in \llbracket 0; f'-1 \rrbracket. \end{aligned}$$

□

A Numerical bounds

In this section, we look at the numerical values of the bounds on the threshold that we derived in the main text. We obtain a threshold $p_{\text{ls}} = 2.70 \cdot 10^{-16}$ for the small-set-flip decoding algorithm of quantum expander codes with errors in the locally stochastic noise model.

By Theorem 4, there exists a family of $(\gamma_A, \delta_A, \gamma_B, \delta_B)$ -expanders for $\gamma_A, \gamma_B = \Omega(1)$ with left degree $d_A = 38$, right degree $d_B = 39$, $\delta_A = 1/d_A + \epsilon$ and $\delta_B = 1/d_B + \epsilon$ for ϵ arbitrarily small. From the construction of Section 2.3 we get a family of quantum expander codes. The degree of the associated adjacency graph is $d = d_B^2 + 2d_B(d_A - 1) = 4407$ and by Definition 10 and propositions 11 and 15 and Theorem 17, the small-set-flip algorithm with parameter $\beta =$

$\frac{d_A}{2d_B} [1 - 4(\delta_A + \delta_B + (\delta_B - \delta_A)^2)] \approx 0.386$ corrects random errors in the locally stochastic noise model of probability $p < p_{\text{ls}}$ with high probability where

$$\alpha = \frac{\beta}{1 + \beta} \approx 0.278, \quad p_{\text{ls}} = \left(\frac{2^{-h(\alpha)}}{(d-1)(1 + \frac{1}{d-2})^{d-2}} \right)^{\frac{1}{\alpha}} \approx 2.70 \cdot 10^{-16}.$$

Note that the threshold p_{iid} in the independent error model is essentially the same: $p_{\text{iid}} - p_{\text{ls}} \approx 1 \cdot 10^{-27}$.

From a practical point of view, this theoretical value of $2.70 \cdot 10^{-16}$ is too small to be relevant. We believe, however, that in practice the threshold will be much better. Take as an example the toric code: our analysis leads to a threshold of $8.1 \cdot 10^{-4}$ in the independent error model with ideal syndrome extraction, whereas it is known that the actual threshold is around 10% (see e.g. Ref. [29]). Notice furthermore that the graph expansion arguments are generally pessimistic since they are based on averaging (see for example the discussion of Ref. [24] example 8.14, about the correction capacity of the bit-flip algorithm for classical expander codes). In addition the threshold for α -percolation given in Theorem 17 is valid for any graph \mathcal{G} of degree bounded by d . In particular, the case $\alpha = 1$ shows that the threshold is the smallest when the graph \mathcal{G} is randomly sampled [16]. The point is that the number of connected sets in that kind of graphs is close to the upper bound of Lemma 27. In contrast, the adjacency graphs \mathcal{G} we considered are not random and seem to contain far less connected sets than the upper bound that we relied on. As a consequence, if one were able to improve the bound of Lemma 27, it would automatically lead to a better threshold as can be seen in the proof of Theorem 17. One last argument is that one can improve the analysis of the “propagation” of errors: suppose that during the bit flip algorithm, a bit v with no error is flipped, then it would mean that at least half of the neighbours of v in the graph \mathcal{G} are in error. This shows that not all the connected sets allow the error to propagate, and thus the bound of Lemma 27 would be improved by only counting this kind of connected sets. Note also the similarity with the setup of “bootstrap percolation” (see Ref. [1]). We believe that some arguments from this field can be used to improve our threshold.

B Proofs

B.1 Proof of Proposition 11

We follow the proof of [20]. An element $g \in C_Z$ is called a generator. An error $E \subseteq V$ is called reduced if $|E|$ is minimal among the errors of $E + C_Z^\perp$. Following Ref. [20], we introduce a notion of *critical* generator which corresponds to a generator g with $\Gamma_Z(g)$ containing a set of qubits that can be flipped to decrease the syndrome weight. More precisely, a critical generator as defined below can always be used to decrease the syndrome weight, but there might exist noncritical generators that also contain a subset of qubits that lead to a decrease of the weight of the syndrome, when flipped.

Definition 22 (Definition 6 of [20]). *A generator $g \in C_Z$ is said to be critical for $E \subseteq V$ if:*

$$\Gamma_Z(g) = x_a \uplus \bar{x}_a \uplus \chi_a \uplus x_b \uplus \bar{x}_b \uplus \chi_b$$

where

- $\Gamma_Z(g) \cap A^2 = x_a \uplus \bar{x}_a \uplus \chi_a$ and $\Gamma_Z(g) \cap B^2 = x_b \uplus \bar{x}_b \uplus \chi_b$;
- $x_a, x_b \subseteq E$ and $\bar{x}_a, \bar{x}_b \subseteq V \setminus E$;
- for all $v_a \in x_a, v_b \in x_b, \bar{v}_a \in \bar{x}_a$ and $\bar{v}_b \in \bar{x}_b$:

- $E \cap \Gamma_X[\Gamma_X(v_a) \cap \Gamma_X(v_b)] = \{v_a, v_b\}$
- $E \cap \Gamma_X[\Gamma_X(\bar{v}_a) \cap \Gamma_X(\bar{v}_b)] = \emptyset$
- $E \cap \Gamma_X[\Gamma_X(v_a) \cap \Gamma_X(\bar{v}_b)] = \{v_a\}$
- $E \cap \Gamma_X[\Gamma_X(\bar{v}_a) \cap \Gamma_X(v_b)] = \{v_b\}$

- $x_a \cup x_b \neq \emptyset, |\chi_a| \leq 2\delta_B d_B$ and $|\chi_b| \leq 2\delta_A d_A$.

Lemma 23 (Lemma 7 of [20]). *For an error $E \subseteq V$ such that $0 < |E| \leq \min(\gamma_A n_A, \gamma_B n_B)$, there exists a critical generator for E .*

Lemma 24 (Modified version of Lemma 8 of [20]). *Let r and $\beta = \beta_0$ be defined as in Definition 10. Let $E \subseteq V$ be a reduced error such that $0 < |E| \leq r \min(\gamma_A n_A, \gamma_B n_B)$ then there exists an error $F \in \mathcal{F}$ with*

$$|\sigma_X(E)| - |\sigma_X(E \oplus F)| \geq \beta d_B |F|.$$

Proof. For convenience, we use a notion of reduced cardinality. Given an error $E \subseteq V = A^2 \uplus B^2$, $\|E\|$ is defined by:

$$\|E\| = \frac{|E \cap A^2|}{d_B} + \frac{|E \cap B^2|}{d_A}.$$

It is straightforward to check that for $E \subseteq V$:

$$d_A \|E\| \leq |E| \leq d_B \|E\|. \quad (10)$$

Let E_R be the error of $E + \mathcal{C}_Z^\perp$ which minimizes $\|E_R\|$. The size of E_R satisfies:

$$0 < |E_R| \leq d_B \|E_R\| \leq d_B \|E\| \leq \frac{d_B}{d_A} |E| \leq \min(\gamma_A n_A, \gamma_B n_B).$$

We exploit Lemma 23 applied to E_R and prove that if we flip $F = x_A \uplus x_B$ in the generator promised by the lemma then $|\sigma_X(E)| - |\sigma_X(E \oplus F)| \geq \beta d_B |F|$.

We use the notations of the proof of Lemma 8 of [20]:

$$\begin{aligned} x = \|x_a\| &= |x_a|/d_B & z = \|\chi_a\| &= |\chi_a|/d_B & \bar{x} &= 1 - x - z = \|\bar{x}_a\| = |\bar{x}_a|/d_B \\ y = \|x_b\| &= |x_b|/d_A & t = \|\chi_b\| &= |\chi_b|/d_A & \bar{y} &= 1 - y - t = \|\bar{x}_b\| = |\bar{x}_b|/d_A \end{aligned}$$

For $F = x_A \uplus x_B$, we have $\|F\| = x + y$ and:

$$\begin{aligned} |\sigma_X(E)| - |\sigma_X(E \oplus F)| &\geq [|x_a| |\bar{x}_b| + |\bar{x}_a| |x_b| - |x_a| |\chi_b| - |x_b| |\chi_a|] \\ &= d_A d_B [x \bar{y} + \bar{x} y - x t - y z] \\ &= d_A d_B [x(1 - y - t) + (1 - x - z)y - x t - y z] \\ &= d_A d_B [x + y - 2xy - 2xt - 2yz] \\ &\geq d_A d_B [x + y - 2xy - 4x\delta_A - 4y\delta_B] \\ &= d_A d_B [x(1 - 4\delta_A) + y(1 - 4\delta_B) - 2xy] \end{aligned}$$

Moreover $0 < x + y$ because $x_A \uplus x_B \neq \emptyset$ and $x + y \leq 1$ because E_R minimizes $\|E_R\|$ over $E + \mathcal{C}_Z^\perp$.

Let $f(x, y) = \frac{1}{x + y} [x(1 - 4\delta_A) + y(1 - 4\delta_B) - 2xy]$ defined for x, y such that:

$$0 \leq x \leq 1, \quad 0 \leq y \leq 1, \quad 0 < x + y \leq 1.$$

The function analysis below shows that $f(x, y) \geq \beta/r$ and thus Lemma 23 holds using eq. (10):

$$|\sigma_X(E)| - |\sigma_X(E \oplus F)| \geq d_A d_B f(x, y) \|F\| \geq \beta d_B |F|.$$

The remaining point is to show that $f(x, y) \geq \beta/r = \frac{1}{2} [1 - 4(\delta_A + \delta_B + (\delta_B - \delta_A)^2)]$:
The partial derivative of f with respect to x is equal to:

$$\frac{\partial f}{\partial x}(x, y) = \frac{y}{(x+y)^2} [4(\delta_B - \delta_A) - 2y].$$

- If $y \leq 2(\delta_B - \delta_A)$ then $x \mapsto f(x, y)$ is non-decreasing and thus:

$$f(x, y) \geq f(0, y) = 1 - 4\delta_B \geq \beta/r.$$

- Otherwise, $x \mapsto f(x, y)$ is non-increasing and thus:

$$f(x, y) \geq f(1 - y, y) = 2y^2 + y(4\delta_A - 4\delta_B - 2) + 1 - 4\delta_A.$$

$f(x, y)$ is then lower bounded by β/r the minimum of $y \mapsto f(1 - y, y)$ on $[0, 1]$ reached for $y = 1/2 + \delta_B - \delta_A$.

□

Proof of Proposition 11. Let $E_0 \subseteq V$ with $|E_0| \leq \frac{r\beta}{1+\beta} \min(\gamma_A n_A, \gamma_B n_B)$ and run Algorithm 2 with parameter β on E_0 . Set $E_i = E_0 \oplus \hat{E}_i = E_0 \oplus \left(\bigoplus_{k=0}^{i-1} F_k \right)$ for $i \in \llbracket 1; f \rrbracket$, where f is the number of steps of the algorithm (the algorithm terminates because the syndrome weight decreases strictly at each step and must remain non negative). Since $|\sigma_X(E_i)| - |\sigma_X(E_{i+1})| \geq \beta d_B |F_i|$, we have:

$$\beta d_B \sum_{i=0}^{f-1} |F_i| \leq \sum_{i=0}^{f-1} |\sigma_X(E_i)| - |\sigma_X(E_{i+1})| = |\sigma_X(E_0)| - |\sigma_X(E_f)| \leq d_B |E_0|,$$

and:

$$|E_f| = \left| E_0 \oplus \left(\bigoplus_{i=0}^{f-1} F_i \right) \right| \leq |E_0| + \sum_{i=0}^{f-1} |F_i| \leq \frac{1+\beta}{\beta} |E_0| \leq r \min(\gamma_A n_A, \gamma_B n_B).$$

Let E_R be the reduced error of $E_f + \mathcal{C}_Z^\perp$ then $|E_R| \leq |E_f| \leq r \min(\gamma_A n_A, \gamma_B n_B)$ and Algorithm 2 stops on $s_f = \sigma_X(E_f) = \sigma_X(E_R)$. We conclude by Lemma 24 that $|E_R| = 0$ and therefore the error is corrected. □

B.2 Results for percolation

We start by stating useful bounds on binomial coefficients.

Lemma 25. *For k, n integers, we have:*

$$\binom{n}{k} \leq 2^{nh(k/n)},$$

where $h(x) = -x \log_2(x) - (1-x) \log_2(1-x)$ is the binary entropy function.

Lemma 26 (Chernoff bound for the binomial distribution). *Let $p \in [0; 1]$, $s \in \mathbb{N}$ and $k \geq sp$, then we have:*

$$\sum_{m \geq k} \binom{s}{m} p^m (1-p)^{s-m} \leq 2^{-sD(\frac{k}{s} \| p)}.$$

Lemma 27 gives an upper bound on the number of connected sets of size s in a graph with an upper bound on the degree. Note that for a rooted tree of degree bounded by d , each vertex which is not the root has at most $d-1$ sons. Using this remark and results from [23], Lemma 27 can be proved following the proof of [28].

Lemma 27 ([23],[28]). *Let $\mathcal{G} = (V, \mathcal{E})$ be a graph of degree bounded by $d \geq 2$. For $s \geq 1$, the number of connected sets of size s satisfies $|\mathcal{C}_s(\mathcal{G})| \leq |V| \frac{d}{s[s(d-2)+2]} \binom{s(d-1)}{s-1}$.*

Proof. Let $\mathcal{T}(s)$ be the set of labelled rooted trees with s vertices such that the maximum out-degree of the root is d and the maximum outdegree of the other nodes is $d-1$. An element of $\mathcal{T}(s)$ is a directed graph where the root has at most d sons and the other vertices have at most $d-1$ sons. The directed edges whose head is some node v are injectively labeled with labels in $\llbracket 1; d \rrbracket$ if v is the root and with labels in $\llbracket 1; d-1 \rrbracket$ if v is not the root. Note that in the underlying undirected graph, a node which is not the root can have d neighbours. Using Ref. [23], we have $|\mathcal{T}(s)| = R(d-1, s-1, d) := \frac{d}{s(d-2)+2} \binom{s(d-1)}{s-1}$ where $R(a, b, c)$ are the Raney numbers.

From the graph \mathcal{G} we construct the oriented graph \mathcal{G}_0 where each non-oriented edge of \mathcal{G} has been replaced by two opposite oriented edges. Similarly than for the trees, we fix some labelling of \mathcal{G}_0 : the directed edges whose head is some node v are injectively labeled with labels in $\llbracket 1; \deg^+(v) \rrbracket$ where $\deg^+(v)$ is the outdegree of v .

Now let $v \in V$ and let $\mathcal{C}_s(v)$ be the set of connected sets $X \in \mathcal{C}_s(\mathcal{G})$ with $v \in X$. For a given $X \in \mathcal{C}_s(v)$, there is at least one spanning tree T_0 of X . We then get a labelled rooted tree $T \in \mathcal{T}(s)$ by fixing v as the root and fixing the labelling of T using the labelling of T_0 in \mathcal{G}_0 . Note that the labelling for T is not exactly the same labelling than T_0 since the labels in T are in $\llbracket 1; d-1 \rrbracket$ (for edges whose head is not the root) whereas the label d is allowed for T_0 . The key point is that when we consider a vertex $v_1 \neq v$ in T_0 then this vertex has a father v_0 . Let i be the label in T_0 of the oriented edge from v_1 to v_0 and j be the label in T_0 of another edge whose head is v_1 . The label in T of this edge is then j if $j < i$ and $j-1$ if $j > i$. Conversely given $T \in \mathcal{T}(s)$, T can be obtained by this procedure using at most one connected set $X \in \mathcal{C}_s(v)$. Thus $|\mathcal{C}_s(v)| \leq |\mathcal{T}(s)|$.

When we consider the sum $\sum_{v \in V} |\mathcal{C}_s(v)|$, each connected set of $\mathcal{C}_s(\mathcal{G})$ is counted s times. Therefore,

$$|\mathcal{C}_s(\mathcal{G})| = \frac{1}{s} \sum_{v \in V} |\mathcal{C}_s(v)| \leq |V| \frac{d}{s[s(d-2)+2]} \binom{s(d-1)}{s-1}.$$

□

Corollary 28. Let $\mathcal{G} = (V, \mathcal{E})$ be a graph of degree bounded by $d \geq 2$. For $s \geq 1$, the number of connected sets of size s satisfies $|\mathcal{C}_s(\mathcal{G})| \leq |V|K(d)^s$ where $K(d) = (d-1) \left(1 + \frac{1}{d-2}\right)^{d-2}$.

Proof. From Lemma 27, we have:

$$\begin{aligned} |\mathcal{C}_s(\mathcal{G})| &\leq |V| \frac{d}{s[s(d-2)+2]} \binom{s(d-1)}{s-1} \\ &= |V| \frac{d}{[s(d-2)+1][s(d-2)+2]} \binom{s(d-1)}{s} \\ &\leq |V| \binom{s(d-1)}{s} \quad \text{since } d \geq 2 \text{ and } s \geq 1 \\ &\leq |V| 2^{s(d-1)h(1/(d-1))} \\ &= |V| \left((d-1) \left(1 + \frac{1}{d-2}\right)^{d-2} \right)^s. \end{aligned}$$

□

We now prove the improved bound (5) on the probability of E to be a connected α -subset of size $\geq t$. The calculation is similar to that in [19, Theorem 3] who considered $\frac{1}{2}$ -percolation in their proof.

Proof of Eq. (5). For $v \in V, X \subseteq V, E \subseteq V$ and $s \in \mathbb{N}$, let us define:

$$\begin{aligned} \mathcal{C}_s(v) &= \{X \in \mathcal{C}_s(\mathcal{G}) : v \in X\}, \\ \partial X &= \Gamma(X) \setminus X, \text{ the boundary of } X, \\ A(E, \alpha, s, v) &= \{X \in \mathcal{C}_s(v) : |X \cap E| \geq \alpha|X|, \partial X \cap E = \emptyset\}. \end{aligned}$$

In other words, the sets $A(E, \alpha, s, v)$ are the connected sets E containing v such that $|X \cap E| \geq \alpha|X|$ (and whose boundary does not intersect E). The reason for the second condition in the definition of $A(E, \alpha, s, v)$ is to capture the idea that the sets $X \in A$ are of maximum size: indeed, increasing the size of a set $X \in A(E, \alpha, s, v)$ would only decrease the fraction of vertices in $X \cap E$.

By induction on the size $|X|$ of a connected set $X \subseteq V$, it is straightforward to show that $|\partial X| \leq (d-2)|X| + 2$.

By definition, we have:

$$\begin{aligned} \mathbb{P}[\text{MaxConn}_\alpha(E) \geq t] &= \mathbb{P}[\exists s \geq t, \exists X \in \mathcal{C}_s(\mathcal{G}) : |X \cap E| \geq \alpha|X|] \\ &= \mathbb{P}[\exists s \geq t, \exists X \in \mathcal{C}_s(\mathcal{G}) : |X \cap E| \geq \alpha|X|, \partial X \cap E = \emptyset] \\ &= \mathbb{P}[\exists s \geq t, \exists v \in V : A(E, \alpha, s, v) \neq \emptyset] \\ &\leq \sum_{s \geq t} \sum_{v \in V} \mathbb{P}[A(E, \alpha, s, v) \neq \emptyset]. \end{aligned}$$

Note that we used the fact that if there exists $X \in \mathcal{C}_s(\mathcal{G})$ with $s \geq t$ such that $|X \cap E| \geq \alpha|X|$, then we can construct X' by repeatedly adding vertices in $\partial X \cap E$ until we cannot do so. Thus, X' satisfied $\partial X' \cap E = \emptyset$. In addition, we have $|X' \cap E| - |X \cap E| = |X'| - |X|$ is the number of vertices added in this process, which implies that $|X' \cap E| \geq \alpha|X'|$.

For $v \in V$ and $s \in \mathbb{N}$, we have:

$$\mathbb{P}[A(E, \alpha, s, v) \neq \emptyset] \leq \sum_{X \in \mathcal{C}_s(v)} \sum_{m \geq \alpha s} \binom{s}{m} p^m (1-p)^{s-m+|\partial X|}.$$

Note furthermore that for any p , we have

$$\mathbb{P}[A(E, 1, s, v) \neq \emptyset] = \sum_{X \in \mathcal{C}_s(v)} p^s (1-p)^{|\partial X|} \leq 1. \quad (11)$$

For $p < \frac{1}{d-1}$, we have:

$$\begin{aligned} & \mathbb{P}[A(E, \alpha, s, v) \neq \emptyset] \\ & \leq \sum_{X \in \mathcal{C}_s(v)} (1-p)^{|\partial X|} \sum_{m \geq \alpha s} \binom{s}{m} p^m (1-p)^{s-m} \\ & \leq \sum_{X \in \mathcal{C}_s(v)} (1-p)^{|\partial X|} 2^{-sD(\alpha \| p)} \quad \text{by Lemma 26} \\ & = \sum_{X \in \mathcal{C}_s(v)} \left(\frac{1-p}{1 - \frac{1}{d-1}} \right)^{|\partial X|} \frac{2^{-sD(\alpha \| p)}}{(d-1)^s} (d-1)^s \left(1 - \frac{1}{d-1}\right)^{|\partial X|} \\ & \leq \left(\frac{1-p}{1 - \frac{1}{d-1}} \right)^{(d-2)s+2} 2^{-sD(\alpha \| p)} (d-1)^s \sum_{X \in \mathcal{C}_s(v)} \left(\frac{1}{d-1} \right)^s \left(1 - \frac{1}{d-1}\right)^{|\partial X|} \quad \text{since } p < \frac{1}{d-1} \\ & \leq \left(\frac{(1-p)(d-1)}{d-2} \right)^2 \left[(1-p)^{d-2} 2^{-D(\alpha \| p)} \frac{(d-1)^{d-1}}{(d-2)^{d-2}} \right]^s \quad \text{from Eq.(11)} \\ & = \left(\frac{d-1}{d-2} \right)^2 q^s \end{aligned}$$

where we defined $q = (1-p)^{d-1-\alpha} p^\alpha 2^{h(\alpha) + (d-1)h(\frac{1}{d-1})}$. The main improvement compared to the local stochastic bound is the factor $(1-p)^{d-1-\alpha}$.

Summing over $s \geq t$ and $v \in V$ yields:

$$\begin{aligned} \mathbb{P}[\text{MaxConn}_\alpha(E) \geq t] & \leq \sum_{s \geq t} \sum_{v \in V} \left(\frac{d-1}{d-2} \right)^2 q^s \\ & \leq |V| \left(\frac{d-1}{d-2} \right)^2 \frac{q^t}{1-q}. \end{aligned}$$

□

Lemma 29. For $\alpha \in (0, 1]$, the function $q : p \mapsto (1-p)^{d-1-\alpha} p^\alpha 2^{h(\alpha)} (d-1) \left(1 + \frac{1}{d-2}\right)^{d-2}$ is increasing in the interval $[0, \frac{\alpha}{d-1}]$ and it satisfies $q(0) = 0$ and $q(\frac{\alpha}{d-1}) \geq 1$. Consequently, there exists p_{iid} in this interval such $q(p_{\text{iid}}) = 1$.

Proof. It is simple to verify that q is increasing. Moreover, we have $q(0) = 0$ and

$$\begin{aligned} q\left(\frac{\alpha}{d-1}\right) & = \left(1 - \frac{\alpha}{d-1}\right)^{d-1-\alpha} \left(\frac{\alpha}{d-1}\right)^\alpha 2^{h(\alpha)} (d-1) \left(1 + \frac{1}{d-2}\right)^{d-2} \\ & = \left(1 - \frac{\alpha}{d-1}\right)^{d-1-\alpha} \left(\frac{\alpha}{d-1}\right)^\alpha 2^{h(\alpha)} \frac{(d-1)^{d-1}}{(d-2)^{d-2}} \\ & = \frac{(d-1-\alpha)^{d-1-\alpha}}{(d-2)^{d-2}} \alpha^\alpha \alpha^{-\alpha} (1-\alpha)^{-(1-\alpha)} \\ & \geq 1, \end{aligned}$$

where we used the fact that $\alpha \in (0, 1]$.

□

The following proposition gives an upper bound on the value of p for which we have only small connected components. This shows basically that the value in Eq. (3) of Theorem 17 can be at most improved by a factor of $(\frac{2^{-h(\alpha)}}{e})^{\frac{1}{\alpha}}$.

Proposition 30. *Let $d \geq 3$ be an integer, $\alpha \in (0, 1]$, ℓ be the smallest integer $\ell > \frac{1}{\alpha}$ and $p > \frac{1}{(d-1)^\ell}$. Then there exists a constant κ (only depending on d, α, p) such that for any $c \in \mathbb{N}^*$, there exists and a family of graphs G_k on N_k vertices (with $N_k \rightarrow \infty$) such that if E contains each vertex independently with probability p , we have*

$$\lim_{k \rightarrow \infty} \mathbb{P}[\text{MaxConn}_\alpha(E) \geq \kappa |N_k|^{1-1/c}] = 1.$$

Proof. We define G_k to be a complete $(d-1)$ -tree of height ck . Note that every vertex has degree at most d and the number of vertices is $N_k = \frac{(d-1)^{ck}-1}{d-2}$. We write V for the set of vertices of the graph G_k and L_i for the set of vertices at depth i , where the root is at depth 0.

For every such vertex $v \in L_{(c-1)k}$, we define the event:

$$E_v = \left\{ \exists w_{(c-1)k}, \dots, w_{ck-1} : w_{(c-1)k} = v, w_{i+1} \text{ is a child of } w_i \text{ and } |E \cap \{w_{(c-1)k}, \dots, w_{ck-1}\}| \geq \frac{1}{\ell}k \right\}. \quad (12)$$

We define $P_v = \emptyset$ if E_v does not hold and if it does hold, then $P_v = \{w_{(c-1)k}, \dots, w_{ck-1}\}$, where w_i are as in (12). We then define the (random) subset S of vertices as

$$S = \bigcup_{i=0}^{(c-1)k-1} L_i \cup \bigcup_{v \in L_{(c-1)k}} P_v.$$

The claim is then that, provided k is large enough, with high probability, S is an α -subset of E and it is of size $\Omega(|N_k|^{1-1/c})$. To see this, observe that

$$\begin{aligned} |E \cap S| &\geq |\{v \in L_{(c-1)k} : E_v \text{ holds}\}| \frac{1}{\ell}k \\ &= \alpha k |\{v \in L_{(c-1)k} : E_v \text{ holds}\}| + \left(\frac{1}{\ell} - \alpha\right) k |\{v \in L_{(c-1)k} : E_v \text{ holds}\}| \end{aligned}$$

On the other hand

$$|S| = k |\{v \in L_{(c-1)k} : E_v \text{ holds}\}| + \frac{(d-1)^{(c-1)k} - 1}{d-2}.$$

Thus, in order to show that S is an α -subset of E , it suffices to show that for large enough k , we have with high probability

$$\left(\frac{1}{\ell} - \alpha\right) k |\{v \in L_{(c-1)k} : E_v \text{ holds}\}| \geq \alpha \frac{(d-1)^{(c-1)k} - 1}{d-2}.$$

Now to prove this, it suffices to show that for any $v \in L_{(c-1)k}$, $\mathbb{P}[E_v] \geq \tau$ with $\tau > 0$ a constant independent of k and c . In fact, once we have that, $|\{v \in L_{(c-1)k} : E_v \text{ holds}\}|$ has a binomial distribution with parameters $((d-1)^{(c-1)k}, \geq \tau)$ and the inequality will follow easily for large enough k . Note also the inequality $\mathbb{P}[E_v] \geq \tau$ also implies the claimed lower bound on the size of S .

Now to lower bound the probability of E_v , we look at an even more restricted event. Namely, we consider the event that $w_{(c-1)k} = v \in E$, and there exists a descendent $w_{(c-1)k+\ell} \in L_{(c-1)k+\ell}$ of $w_{(c-1)k}$ that is in E , and there is a descendent $w_{(c-1)k+2\ell} \in L_{(c-1)k+2\ell}$ of $w_{(c-1)k+\ell}$ that is in E , and all the way until we reach the bottom of the tree. Note that the probability that such a path exists is lower bounded by the probability of survival of the branching process with offspring distribution binomial with parameters $((d-1)^\ell, p)$. But as $p(d-1)^\ell > 1$, the probability of survival is > 0 and we obtain the desired result. \square

Acknowledgments

We would like to thank Benjamin Audoux, Alain Couvreur, Anirudh Krishna, Vivien Londe, Jean-Pierre Tillich and Gilles Zémor for many fruitful discussions on quantum codes as well as Daniel Gottesman for answering our questions about [13]. AG and AL acknowledge support from the ANR through the QuantERA project QCDA.

References

- [1] Joan Adler. Bootstrap percolation. *Physica A: Statistical Mechanics and its Applications*, 171(3):453–470, 1991.
- [2] Dorit Aharonov and Michael Ben-Or. Fault-tolerant quantum computation with constant error. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 176–188. ACM, 1997.
- [3] Sergey Bravyi and Barbara Terhal. A no-go theorem for a two-dimensional self-correcting quantum memory based on stabilizer codes. *New Journal of Physics*, 11(4):043029, 2009.
- [4] A Robert Calderbank and Peter W Shor. Good quantum error-correcting codes exist. *Physical Review A*, 54(2):1098, 1996.
- [5] Nicolas Delfosse and Naomi H. Nickerson. Almost-linear time decoding algorithm for topological codes. *arXiv preprint arXiv:1709.06218*, 2017.
- [6] Nicolas Delfosse and Gilles Zémor. Quantum erasure-correcting codes and percolation on regular tilings of the hyperbolic plane. In *Information Theory Workshop (ITW), 2010 IEEE*, pages 1–5. IEEE, 2010.
- [7] Nicolas Delfosse and Gilles Zémor. Upper bounds on the rate of low density stabilizer codes for the quantum erasure channel. *Quantum Information & Computation*, 13(9&10):0793–0826, 2013.
- [8] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Journal of Mathematical Physics*, 43(9):4452–4505, 2002.
- [9] Jack Edmonds. Maximum matching and a polyhedron with 0, 1-vertices. *Journal of Research of the National Bureau of Standards B*, 69(125-130):55–56, 1965.
- [10] Michael H Freedman, David A Meyer, and Feng Luo. Z_2 -systolic freedom and quantum codes. *Mathematics of quantum computation*, Chapman & Hall/CRC, pages 287–320, 2002.
- [11] Robert Gallager. Low-density parity-check codes. *IRE Transactions on information theory*, 8(1):21–28, 1962.
- [12] Daniel Gottesman. Stabilizer codes and quantum error correction. *arXiv preprint quant-ph/9705052*, 1997.
- [13] Daniel Gottesman. Fault-tolerant quantum computation with constant overhead. *arXiv preprint arXiv:1310.2984*, 2013.
- [14] Larry Guth and Alexander Lubotzky. Quantum error correcting codes and 4-dimensional arithmetic hyperbolic manifolds. *Journal of Mathematical Physics*, 55(8):082202, 2014.

- [15] Matthew B Hastings. Decoding in hyperbolic spaces: Ldpc codes with linear rate and efficient error correction. *arXiv preprint arXiv:1312.2546*, 2013.
- [16] Svante Janson. On percolation in random graphs with given vertex degrees. *Electronic Journal of Probability*, 14:86–118, 2009.
- [17] Isaac Hyun Kim. *Quantum codes on Hurwitz surfaces*. PhD thesis, Massachusetts Institute of Technology, 2007.
- [18] A Yu Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, 2003.
- [19] Alexey A Kovalev and Leonid P Pryadko. Fault tolerance of quantum low-density parity check codes with sublinear distance scaling. *Physical Review A*, 87(2):020304, 2013.
- [20] Anthony Leverrier, Jean-Pierre Tillich, and Gilles Zémor. Quantum expander codes. In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pages 810–824. IEEE, 2015.
- [21] Vivien Londe and Anthony Leverrier. Golden codes: quantum LDPC codes built from regular tessellations of hyperbolic 4-manifolds. *arXiv preprint arXiv:1712.08578*, 2017.
- [22] Russell Lyons. Random walks, capacity and percolation on trees. *The Annals of Probability*, pages 2043–2088, 1992.
- [23] Chin Hee Pah and Mohamed Ridza Wahiddin. Combinatorial interpretation of raney numbers and tree enumerations. *Open Journal of Discrete Mathematics*, 5(01):1, 2015.
- [24] Tom Richardson and Ruediger Urbanke. *Modern coding theory*. Cambridge University Press, 2008.
- [25] Michael Sipser and Daniel A Spielman. Expander codes. *IEEE Transactions on Information Theory*, 42(6):1710–1722, 1996.
- [26] Andrew M Steane. Error correcting codes in quantum theory. *Physical Review Letters*, 77(5):793, 1996.
- [27] Jean-Pierre Tillich and Gilles Zémor. Quantum ldpc codes with positive rate and minimum distance proportional to the square root of the blocklength. *IEEE Transactions on Information Theory*, 60(2):1193–1202, 2014.
- [28] Ryuhei Uehara et al. The number of connected components in graphs and its applications. *Manuscript*. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary>, 1999.
- [29] David S Wang, Austin G Fowler, Ashley M Stephens, and Lloyd Christopher L Hollenberg. Threshold error rates for the toric and surface codes. *arXiv preprint arXiv:0905.0531*, 2009.
- [30] Gilles Zémor. On Cayley Graphs, Surface Codes, and the Limits of Homological Coding for Quantum Error Correction. In *IWCC*, pages 259–273. Springer, 2009.