arXiv:1808.03821v1 [quant-ph] 11 Aug 2018

Constant overhead quantum fault-tolerance with quantum expander codes

Omar Fawzi* Antoine Grospellier† Anthony Leverrier†

Abstract

We prove that quantum expander codes can be combined with quantum fault-tolerance techniques to achieve constant overhead: the ratio between the total number of physical qubits required for a quantum computation with faulty hardware and the number of logical qubits involved in the ideal computation is asymptotically constant, and can even be taken arbitrarily close to 1 in the limit of small physical error rate. This improves on the polylogarithmic overhead promised by the standard threshold theorem.

To achieve this, we exploit a framework introduced by Gottesman together with a family of constant rate quantum codes, *quantum expander codes*. Our main technical contribution is to analyze an efficient decoding algorithm for these codes and prove that it remains robust in the presence of noisy syndrome measurements, a property which is crucial for fault-tolerant circuits. We also establish two additional features of the decoding algorithm that make it attractive for quantum computation: it can be parallelized to run in logarithmic depth, and is single-shot, meaning that it only requires a single round of noisy syndrome measurement.

1 Introduction

Quantum computers are expected to offer significant, sometimes exponential, speedups compared to classical computers. For this reason, building a large, universal computer, is a central objective of modern science. Despite two decades of effort, experimental progress has been somewhat slow and the largest computers available at the moment reach a few tens of physical qubits, still quite far from the numbers necessary to run "interesting" algorithms. A major source of difficulty is the inherent fragility of quantum information: storing a qubit is quite challenging, processing quantum information even more so.

Any physical implementation of a quantum computer is necessarily imperfect because qubits are subject to decoherence and physical gates can only be approximately realized. In order to perform a correct computation on a faulty computer, techniques of fault-tolerant computation must be developed. One of the crowning achievements of the early days of quantum computing is the *threshold theorem* which states that upon encoding the logical qubits within the appropriate quantum error correcting code, it is possible to perform arbitrary long computations on a faulty quantum computer, provided that the noise level is below some *constant* threshold value [1]. This solution comes at a cost, however, since the fault-tolerant version of the circuit to be performed is usually larger that the initial version. In particular, a number of ancilla qubits is required and the space overhead, *i.e.*, the ratio between the total number of physical qubits and the number of logical qubits, scales polylogarithmically with the number of gates involved in the original computation. Indeed, the main technique to achieve fault-tolerance is to protect the logical qubits with concatenated codes. In order to guarantee a final failure probability ε for a circuit C acting

*Univ Lyon, ENS de Lyon, CNRS, UCBL, LIP UMR 5668, F-69007 Lyon, France. omar.fawzi@ens-lyon.fr
†Inria, France. antoine.grospellier@inria.fr, anthony.leverrier@inria.fr

on k qubits with $|\mathrm{C}|$ locations[1], $\mathcal{O}(\log\log(|\mathrm{C}|/\varepsilon))$ levels of encoding are needed, which translates into a $\mathrm{polylog}(|\mathrm{C}|/\varepsilon)$ space overhead. While this might seem like a reasonably small overhead, this remains rather prohibitive in practice, and more importantly, it raises the question of whether this value is optimal. In this paper, we consider a realistic model for quantum computing where the quantum gates are noisy, but all classical computation is assumed to be fast and error-free. Note that if classical gates are also noisy, then it is known that classical fault-tolerance cannot be obtained with constant overhead [20, 10].

In a breakthrough paper, Gottesman has shown that the polylogarithmic overhead was maybe not necessary after all, and that polynomial-time computations could be performed with a noisy circuit with only a *constant* overhead [14]. In fact, the constant can even be taken arbitrarily close to 1 provided that the physical error is sufficiently rate small. In order to overcome the polylogarithmic barrier, Gottesman suggested to use quantum error correcting codes with *constant rate*. More precisely, the idea is to encode the logical qubits in large blocks, but still of size sub-linear in k. The encoding can still be made fault-tolerant thanks to concatenation, but this only yields an overhead polylogarithmic in the block size, and choosing a sufficiently small block size to a sub-linear overhead for encoding. Gates are then performed with Knill's technique by teleporting the appropriate encoded states. Overall, apart from the initial preparation and final measurement, the encoded circuit alternates between steps applying a gate of the original circuit on the encoded state with Knill's technique, and steps of error correction for the quantum code consisting of a measurement of the syndrome, running a (sufficiently fast) classical decoding algorithm and applying the necessary correction. For this to work out properly, and to keep a constant overhead, the syndrome measurement should be efficient and this will be the case if the quantum code is a low-density parity-check (LDPC) code. Indeed, in that case, the syndrome measurement circuit will be of constant depth and won't require any additional overhead. The last property needed for the scheme to work is the existence of an efficient classical decoding algorithm for the quantum code working even in the presence of noise on the syndrome measurement.

The main result of Gottesman is as follows: provided that the right family of quantum codes exists, it is possible to perform fault-tolerant quantum computing with constant overhead. By right family, we mean a family of constant rate LDPC quantum codes with an efficient decoding algorithm robust against noisy syndrome measurements. More precisely, the decoding algorithm should correct stochastic errors of linear weight and fail with negligible probability. Ideally, we also want the algorithm to be sufficiently fast to avoid errors building up during the decoding. At the time, no such family of codes was known to exist. In fact, families of constant rate LDPC codes with unbounded minimum distance are quite difficult to construct, even forgetting about the decoding problem. Possible candidate families include surface codes [9], 4-dimensional topological codes [15] and hypergraph product codes [24]. While surface codes come with an efficient decoding algorithm based on minimum weight matching [7], they only display a logarithmic minimum distance if they have constant rate [6]. Topological 4-D codes come with a much larger minimum distance, but the available efficient decoding algorithms are only known to perform well for errors of logarithmic weight [16, 19]. In both cases, this is insufficient to provide a universal threshold for the error rate, independent of the size of the quantum circuit to be performed. Finally, the family of hypergraph product codes yields the best minimum distance to date for constant rate LDPC codes: the minimum distance scales like the square-root of the block length. In general, however, we don't know of any efficient decoding algorithm for hypergraph product codes.

The hypergraph product construction takes a classical code $[n, k, d_{\min}]$ as input an yields a quantum code $[[N, K, D_{\min}]]$ of length $N = \Theta(n^2)$, dimension $K = \Theta(k^2)$ and a minimum distance equal to that of the classical code. When applying this construction with a classical

[1] A location refers either to a quantum gate, the preparation of a qubit in a given state, a qubit measurement or a wait location if the qubit is not acted upon at a given time step.

expander code [22], it yields a so-called *quantum expander code* [18]. In that case, one has $K = \Theta(N)$ and $D_{\min} = \Theta(\sqrt{N})$, and interestingly, one can take inspiration of the efficient bit-flip decoding algorithm for classical expander codes [22] to design an efficient decoding algorithm for the quantum expander codes. Such an algorithm, the *small-set-flip decoding algorithm*, was introduced in [18] where it was proved that it corrects arbitrary (adversarial) errors of weight $O(D_{\min})$ in linear time. More recently in [8], we studied the behavior of this algorithm against stochastic noise and proved that it corrects random errors of linear weight, except with negligible probability.

In the present work, we extend the analysis significantly and show that the algorithm still works in the presence of a noisy syndrome. This was the missing condition to satisfy all the criteria required by Gottesman's construction. In other words, quantum expander codes can be exploited to obtain quantum fault-tolerance with a constant overhead. In addition, we establish two remarkable features of the decoding algorithm: first, it is *single-shot* meaning that the syndrome measurement need not be repeated a polynomial number of times as in typical constructions [2]: one measurement suffices; second, the algorithm can be parallelized to run in logarithmic time instead of linear time. This second point is important since storage errors will always affect the qubits during the classical decoding step, meaning that it is crucial to reduce the necessary time as much as possible. We note, however, that for our main result below to hold, we need to assume that the error rate affecting the qubits during the decoding step is constant and doesn't depend on the size of the computation. Without this extra assumption in our model, it is implausible that true constant space overhead quantum fault-tolerance can be achieved.

We obtain the following general result by using our analysis of quantum expander codes in Gottesman's generic construction [14]. More details on the definition of circuits and the noise model are given later in Section 2.5.

Theorem 1. *For any $\eta > 1$ and $\varepsilon > 0$, there exists $p_T(\eta) > 0$ such that the following holds for sufficiently large k. Let C be a quantum circuit acting on k qubits, and consisting of $f(k)$ locations for f an arbitrary polynomial. There exists a circuit $\tilde{\mathrm{C}}$ using ηk physical qubits, depth $\mathcal{O}(f(k))$ and number of locations $\mathcal{O}(kf(k))$ that outputs a distribution which has total variation distance at most ε from the output distribution of C, even if the components of $\tilde{\mathrm{C}}$ are noisy with an error rate $p < p_T$.*

Before moving to the proof techniques, let us mention some limitations of the present work. For our analysis to apply, we need bipartite expander graphs with a large (vertex) expansion. A first issue is that there is no known efficient algorithm that can deterministically construct such graphs[2]. While random graphs will display the right expansion (provided their degree is large enough) with high probability, it is not known how to check efficiently that a given graph is indeed sufficiently expanding. The second issue is that we need graphs with a large (constant) degree, which will translate into significantly large quantum codes. In other words, one shouldn't expect the present analysis to be applicable to small size quantum codes that might be built in the near future. We note that Gottesman's analysis also required the initial circuit size to be large enough: this is necessary in order to make the contribution of additive terms sub-linear and therefore obtain a constant overhead. Another limitation of our work is the very small threshold value that it yields. While the threshold is usually expected to lie between 10^{-3} and 10^{-2} for the best constructions based on code concatenation, we expect our value to be several orders of magnitude smaller, as this was already the case in Gottesman's paper [14] and in our previous work with perfect syndrome measurement [8]. Part of the explanation is due to the very crude bounds that we obtain *via* percolation theory arguments. In this work, we haven't tried to optimize the value of the

[2]While algorithms to construct graphs with large *spectral* expansion are known, they do not imply a sufficient vertex expansion for our purpose.

threshold and have instead tried to simplify the general scheme as much as possible. We expect that future work, in particular based on simulations, will help to better understand the true value of the threshold for fault-tolerance schemes with constant overhead. Finally, as already pointed out, we consider a model with error-free classical computation, and assume that the logarithmic-depth decoding algorithm can be performed in constant time.

Main result and proof techniques

In this section, we provide an informal overview of the main result and the techniques used for the proofs. More formal definitions and proofs can be found in the following sections. When decoding a quantum error correcting code, two types of errors need to be taken into account: the X-type (or bit flip) errors and the Z-type (or phase flip) errors. They play a symmetric role for the codes we consider and it is therefore sufficient to focus on bit flips for instance. An X-type error is described by a subset E of the qubits to which the bit flip operator X was applied. To decode a quantum error correcting code, we start by performing a measurement that returns the syndrome $\sigma = \sigma(E)$ which only depends on the error. The objective of the decoding algorithm is given σ to output an error $\hat{E}$ which is the same as E. More precisely, it is sufficient for the errors E and $\hat{E}$ to be *equivalent* in the sense that the error $E \oplus \hat{E}$ acts trivially on every codeword (for a stabilizer code, this simply means that $E \oplus \hat{E}$ belongs to the stabilizer group). As previously mentioned, our main contribution is to analyze the small-set-flip decoding algorithm in the setting where the syndrome measurement is noisy, *i.e.*, the decoding algorithm takes as input $(\sigma(E) \oplus D)$ instead of just $\sigma(E)$, where D represents the syndrome measurement error. The objective of the decoding algorithm is then not to recover the error exactly (which will not be possible) but rather to control the size of remaining error $E \oplus \hat{E}$. In the context of quantum fault-tolerance, the relevant error model for the pair (E, D) is the *local stochastic noise model* with parameters (p, q) defined by requiring that for any F and G, the probability that F and G are part of the qubit and syndrome errors, respectively, is bounded as follows, $\mathbb{P}[F \subseteq E, G \subseteq D] \leq p^{|F|} q^{|G|}$.

Theorem 2 (Informal)**.** *There exist constants $p_0 > 0, p_1 > 0$ such that the following holds. Consider a bipartite graph with sufficiently good expansion and the corresponding quantum expander code. Consider random errors (E, D) satisfying a local stochastic noise model with parameter $(p_{\mathrm{phys}}, p_{\mathrm{synd}})$ with $p_{\mathrm{phys}} < p_0$ and $p_{\mathrm{synd}} < p_1$. Let $\hat{E}$ be the output of the small-set-flip decoding algorithm on the observed syndrome. Then, except for a failure probability of $e^{-\Omega(\sqrt{n})}$, $E \oplus \hat{E}$ is equivalent to E_{ls} that has a local stochastic distribution with parameter $p_{\mathrm{synd}}^{\Omega(1)}$. In addition, the small-set-flip algorithm can be parallelized to run in $\mathcal{O}(\log n)$ depth.*

In the special case where the syndrome measurements are perfect, *i.e.*, $p_{\mathrm{synd}} = 0$, the statement guarantees that for a typical error of size at most $p_0 n$, the small-set-flip algorithm finds an error that is equivalent to the error that occurred. If the syndrome measurements are noisy, then we cannot hope to recover an equivalent error exactly, but instead we can control the size of the remaining error $E \oplus \hat{E}$ by the amount of noise in the syndrome measurements. In particular, for any qubit error rate below p_0, the decoding operation reduces this error rate to be $p_{\mathrm{synd}}^{\Omega(1)}$ (our choice of p_0 will be such that $p_{\mathrm{synd}}^{\Omega(1)} \ll p_0$). This criterion is sufficient for fault-tolerant schemes as it ensures that the size of the qubit errors stay bounded throughout the execution of the circuit. The proof of this theorem consists of two main parts: analyzing arbitrary low weight errors below the minimum distance (Proposition 14) and exploiting percolation theory to analyze stochastic errors of linear weight (Theorem 13).

The small-set-flip decoding algorithm proceeds by trying to flip small sets of qubits so as to decrease the weight of the syndrome, and the main challenge in its analysis is to prove the existence of such a small set F. In the case where the observed syndrome is error free, Refs [18]

and [8] relied on the existence of a "critical generator" to exhibit such a set of qubits. This approach, however, only yields a *single* such set F, and when the syndrome becomes noisy, nothing guarantees anymore that flipping the qubits in F will result in a decrease of the syndrome weight and it becomes unclear whether the decoding algorithm can continue. Instead, in order to take into account the errors on the syndrome measurements, we would like to show that there are *many* possible sets of qubits F that decrease the syndrome weight. In order to establish this point, we consider an error E of size below the minimum distance and we imagine running the (sequential) decoding algorithm [18] without errors on the syndrome. The algorithm gives a sequence of small sets $\{F_i\}$ to flip successively in order to correct the error. In other words, we obtain the following decomposition of the error, $E = \oplus_i F_i$ (note that the sets F_i might overlap). The expansion properties of the graph guarantee that there are very few intersections between the syndromes $\sigma(F_i)$ (see the proof of Proposition 14). In particular in the case of noiseless syndrome, a linear number of these F_i can be flipped to decrease the syndrome weight. There are two consequences to this result. First, it is possible to parallelize the decoding algorithm by flipping multiple F_i in each round (Section 4) and this decreases the syndrome weight by a constant factor, thereby correcting the error after a logarithmic number of rounds. Second, even when the syndrome is noisy there will remain some F_i that can be flipped in order to decrease the syndrome weight and finally, the size of the error $E \oplus \hat{E}$ can be upper bounded with a linear function of the syndrome error size (Proposition 14).

In order to analyze random errors of linear weight, we show using percolation theory that, with high probability, the error forms clusters in the sense of connected α-subsets (Definition 20 and Lemma 27). This is similar to the analysis in [17, 8], except that we use the syndrome adjacency graph of the code (as in [14]) to establish the "locality" of the decoding algorithm, implying that each cluster of the error is corrected independently of the other ones (Lemma 19). Using the fact that clusters are of size bounded by the minimum distance of the code, the result on low weight errors shows that the size of $E \oplus \hat{E}$ is controlled by the syndrome error size. In order to show that the error after correction is local stochastic, we introduce the notion of *witness* (Definition 24). The basic idea is to find a syndrome error in the neighborhood of a given qubit error S. However, a qubit error can be the consequence of a distant syndrome error. This is why a witness is defined as a set of qubit errors W but is potentially larger than S. The previously mentioned results show that witnesses exist for $E \oplus \hat{E}$ and we conclude our proof using an upper bound on the probability that a witness exists.

A remarkable feature of our analysis is that it shows that the small-set-flip decoding algorithm only uses a single noisy syndrome measurement and outputs an error with controlled weight. Note that this is in contrast to decoding algorithms for many other codes such as the toric code for which such a repetition is necessary. This property is called *single-shot* in the fault-tolerant quantum computation literature [2, 5].

Organization

We start in Section 2 with notations and preliminaries to recall the construction and main properties of quantum expander codes, their efficient decoding algorithm. We also introduce the relevant noise model and give an overview of the fault-tolerant scheme of [14] applied to quantum expander codes. In Section 3, we establish our main technical result showing that the small-set-flip decoding algorithm for quantum expander codes is robust against local stochastic noise in the syndrome measurement. In Section 4, we prove that the algorithm can be efficiently parallelized and works in logarithmic time.

2 Preliminaries

In this section, we first review the construction of classical and quantum expander codes. We then discuss models of noise which are relevant in the context of quantum fault-tolerance. Finally, we describe the fault-tolerant construction of Gottesman applied to 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 sometimes 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:

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

where $\Gamma(v) \subseteq C$ is the set of neighbors of v. 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 [22].

Definition 3 (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, δ)-left-expanding *for some constants $\gamma, \delta > 0$, if for any subset $S \subseteq V$ with $|S| \leq \gamma n_V$, the neighborhood $\Gamma(S)$ of S in the graph G satisfies $|\Gamma(S)| \geq (1-\delta) d_V |S|$. Similarly, we say that G is* (γ, δ)-right-expanding *if for any subset $S \subseteq C$ with $|S| \leq \gamma n_C$, we have $|\Gamma(S)| \geq (1-\delta) d_C |S|$. Finally, the graph G is said* (γ, δ)-expanding *if it is both (γ, δ)-left expanding and (γ, δ)-right expanding.*

Sipser and Spielman introduced *expander codes*, which are the linear codes associated with (left-)expander graphs. Remarkably these codes come with an efficient decoding algorithm that can correct *arbitrary* errors of weight $\Omega(n)$ [22].

Theorem 4 (Sipser, Spielman [22])**.** *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 $|E| \leq \gamma(1-2\delta)|V|$.*

Note that the relevant notion of expansion here is that of *vertex expansion*, which differs from *spectral expansion* (defined as a property of the adjacency matrix of the graph). While the two notions are related, we emphasize that even optimal spectral expansion (as in Ramanujan graphs) doesn't imply values of δ below $1/2$ and are therefore insufficient for our purpose.

The decoding algorithm called the "bit-flip" algorithm is very simple: one simply cycles through the bits in V 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 5 (Theorem 8.7 of [21])**.** *Let δ be a positive constant. For integers $d_A > 1/\delta$ and $d_B > 1/\delta$, a graph $G = (A \cup B, \mathcal{E})$ with left-degree bounded by d_A and right-degree bounded by d_B chosen at random according to some distribution is (γ, δ)-expanding for $\gamma = \Omega(1)$ 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 $X = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right), Y = ZX, Z = \left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$ 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], [23], 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, $\mathcal{C}_X^\perp \subseteq \mathcal{C}_Z$), where the dual code $\mathcal{C}_X^\perp$ to $\mathcal{C}_X$ consists 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 neighborhood 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].

A natural noise model is the so-called *Pauli-type noise*, mapping a qubit ρ to $p_{\mathbb{1}}\rho + p_X X\rho X + p_Y Y\rho Y + p_Z Z\rho Z$, for some $p_{\mathbb{1}}, p_X, p_Y, p_Z$. 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 syndrome is the subset $\sigma_X(E_X) \subseteq C_X$ defined by $\sigma_X(E_X) := \bigoplus_{v \in E_X} \Gamma_X(v)$. One describes Z-type error patterns and Z-type syndromes in the same fashion using the factor graph G_Z.

In this paper, we consider Algorithm 2 which tries to recover E_X and E_Z independently. More precisely, the 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 $\sigma_Z(\hat{E}_Z) = \sigma_Z$. Here the two algorithms are identical upon exchanging the roles of X and Z. We note that this kind 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.

Let us conclude this section by mentioning the generic (but inefficient) algorithm which returns an error $(\hat{E}_X, \hat{E}_Z)$ of minimum Hamming weight with the appropriate syndrome [3], that is:

$$\hat{E}_X = \underset{\sigma_X(F_X)=\sigma_X}{\arg\min} |F_X|, \quad \hat{E}_Z = \underset{\sigma_Z(F_Z)=\sigma_Z}{\arg\min} |F_Z|.$$

This 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 [18].

[3] For degenerate quantum codes, this might differ from the output of *maximum likelihood decoding algorithm* which performs an optimization over the probability of equivalent errors.

The construction is as follows. Let $G = (A \cup B, \mathcal{E})$ be a biregular (γ, δ)-expanding graph with δ sufficiently small [4], and constant left and right degrees denoted d_A and d_B with $d_A \leq d_B$. Let us also denote $n_A = |A|$ and $n_B = |B|$ with $n_B \leq n_A$. According to Theorem 5, such graphs can be found in a probabilistic fashion provided that $d_A \geq \lceil \delta^{-1} \rceil$. 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:

$$H_X = \left(I_{n_A} \otimes H, H^T \otimes I_{n_B}\right),$$
$$H_Z = \left(H \otimes I_{n_A}, I_{n_B} \otimes H^T\right).$$

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

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

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

We can describe the factor graphs G_X and G_Z as follows: the set of qubits is indexed by $V := A^2 \cup B^2$, the set of Z-type generators is indexed by $C_X := A \times B$ and the set of X-type generators is indexed by $C_Z := B \times A$. The bipartite graph G_X has left vertices V, right vertices C_X 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 neighborhood of β (resp. α) in G. The bipartite graph G_Z has left vertices V, right vertices C_Z 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 neighborhood of b (resp. a) in G.

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

Theorem 6 (Tillich, Zémor [24])**.** *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* $d_{\min} = d_{\min}(\mathcal{C})$.

A natural approach to perform error correction would be to directly mimic the classical bit-flip decoding algorithm analyzed by Sipser and Spielman, that is try to apply X-type (or Z-type) correction to qubits when it leads to a decrease of the syndrome weight. Unfortunately, in that case, there are error configurations of constant weight that couldn't be corrected: these correspond to error patterns consisting of half a generator where exactly half of its qubits in $A \times A$ and half of its qubits in $B \times B$ are in error. It is easy to check that flipping any single qubit of that generator leaves the syndrome weight invariant, equal to $d_A d_B/2$. On the other hand, removing all the errors from that generator decreases the syndrome weight by $d_A d_B/2$. This suggests the "small-set-flip" strategy that we describe next.

Focusing on X-type errors for instance, and assuming that the syndrome $\sigma = H_X E$ 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 (choosing the one maximizing the ratio between the syndrome weight decrease and the pattern weight), if there are several). The algorithm then proceeds by examining the next generator. Since the generators have (constant) weight $d_A + d_B$, there are $2^{d_A + d_B} = \mathcal{O}(1)$ possible patterns to examine for each generator.

[4] The existence of an efficient algorithm that corrects arbitrary errors of size $O(\sqrt{n})$ is guaranteed as soon as $\delta < 1/6$ [18]. The same algorithm corrects random errors of linear weight except with negligible probability as soon as $\delta < 1/8$ [8] and in the present paper, we will require the more stringent condition $\delta < 1/16$.

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

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

where $\hat{E}$ is the output of the decoding algorithm. This means 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 $\Gamma_Z(g)$ for some generator $g \in C_Z$ which decreases the syndrome weight $|\sigma|$. In other words, it tries to flip some element $F \in \mathcal{F}_0$ such that $\Delta(\sigma, F) > 0$ where:

$$\mathcal{F}_0 := \{F \subseteq \Gamma_Z(g) : g \in C_Z\}, \qquad \Delta(\sigma, F) := |\sigma| - |\sigma \oplus \sigma_X(F)|. \tag{1}$$

The decoding algorithm consists of two iterations of Algorithm 1 below: it first tries to correct X-type errors, then it is applied a second time (exchanging the roles of X and Z) to correct Z-type errors.

Algorithm 1 (Ref. [18]): 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 $(\exists F \in \mathcal{F}_0 : \Delta(\sigma_i, F) > 0)$ **do**

$F_i = \arg\max_{F \in \mathcal{F}_0} \dfrac{\Delta(\sigma_i, F)}{|F|}$ // 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 $\hat{E}_i$

It was proven in Ref. [18] that this algorithm corrects arbitrary errors of size $O(\sqrt{n})$ provided that the expansion of the graph satisfies $\delta < 1/6$.

Theorem 7 (Leverrier, Tillich, Zémor [18]). *Let $G = (A \cup B, \mathcal{E})$ be a (d_A, d_B)-biregular (γ, δ)-expanding graph with $\delta < 1/6$. Letting d_A and d_B be fixed and allowing n_A, n_B to grow, the small-set-flip decoding algorithm (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*

$$w_0 = \frac{\gamma n_B}{3(1 + d_B)}. \tag{2}$$

In a recent work, the analysis was extended to the case of random errors (either independent and identically distributed, or local stochastic) provided that the syndrome extraction is performed perfectly and under a stricter condition on the expansion of the graph [8].

Theorem 8 (Fawzi, Grospellier, Leverrier [8]). *Let $G = (A \cup B, \mathcal{E})$ be a (d_A, d_B)-biregular (γ, δ)-expanding graph with $\delta < 1/8$. Then there exists a probability $p_0 > 0$ and constants*

C, C' such that if the noise parameter on the qubits satisfies $p < p_0$, the small-set-flip decoding algorithm (Algorithm 1) runs in time linear in the code length and corrects a random error with probability at least $1 - Cn\left(\frac{p}{p_0}\right)^{C'\sqrt{n}}$.

The caveat of this result is that it only applies in absence of errors for the syndrome extraction. The main technical contributions of this paper are to establish that the same algorithm still works in presence of noise on the syndrome, and to show that the decoding algorithm can be parallelized to run in logarithmic time.

2.4 Noise models

In the context of quantum fault-tolerance, we are interested in modeling noise occurring during a quantum computation. We refer the reader to the introduction on the topic by Gottesman for a thorough description of noise models for fault-tolerance [13]. In the circuit model of quantum computation, the effect of noise is to cause faults occurring at different locations of the circuit: either on the initial state and ancillas, on gates (either active gates or storage gates) or on measurement gates. We refer to this model as *basic model* for fault-tolerance. The main idea to perform a computation in a fault-tolerant manner is then to encode the logical qubits with a quantum error correcting code, replace the locations of the original circuit by gadgets applying the corresponding gate on the encoded qubits, and interleave the steps of the computation with error correction steps. In general, it is convenient to abstract away the details of the implementation and consider a *simplified model* of fault-tolerance where one is concerned with only two types of errors: errors occurring at each time step on the physical qubits, and errors on the results of the syndrome measurement. The link between the basic and the simplified models for fault-tolerance can be made once a specific choice of gate set and gadgets for each gate is made. This is done for instance in Section 7 of Ref. [14]. In other words, the simplified model of fault-tolerance allows us to work with quantum error correcting codes where both the physical qubits and the check nodes are affected by errors.

As usual in the context of quantum error correction, we restrict our attention to Pauli-type errors since the ability to correct all Pauli errors of weight t implies that arbitrary errors of weight t can be corrected. In particular, one only needs to address X and Z-type errors since a Y-error corresponds to simultaneous X and Z-errors. Therefore, we think of an error pattern on the qubits as a pair (E_X, E_Z) of subsets of the set of qubits V. This should be interpreted as Pauli error X on all qubits in $E_X \setminus E_Z$, error Y on $E_X \cap E_Z$ and error Z on $E_Z \setminus E_X$. Similarly, the error on the syndrome consists of two classical strings (D_X, D_Z) which are subsets of the sets C_X and C_Z of check nodes, whose values have been flipped. This means that the syndromes that are provided as the input of the decoding algorithm are

$$\sigma_X := \sigma_X(E_X) \oplus D_X, \quad \sigma_Z := \sigma_Z(E_Z) \oplus D_Z. \tag{3}$$

The algorithm we will consider in this work treat X errors and Z errors in a symmetric fashion. More precisely, the decoding algorithm first tries to recover E_X from σ_X, then proceeds in a similar way to try to recover E_Z from σ_Z, without exploiting any information about E_X or σ_X. Said otherwise, the algorithm tries to recover both E_X and E_Z independently. For this reason, it will be convenient to restrict our attention to X-type errors in the following since Z-type error would be treated in the same way. In particular, an error will correspond to two sets: a subset E of the qubits and a subset D of the check nodes.

While considering independent errors is natural in the context of quantum error correction, assuming independence isn't really justified for fault-tolerance since errors will tend to propagate through the circuit for instance, and will therefore likely be correlated. For this reason, the usage

is to consider a weaker model of stochastic, but locally decaying noise when studying quantum fault-tolerance: this is the local stochastic error model.

Definition 9 (Local stochastic error model)**.**
Let V be the set of qubits and C be the set of check nodes. A random error (E, D) with $E \subseteq V$ and $D \subseteq C$ satisfies the local stochastic error model with parameters (p, q) if for all $S \subseteq V$ and $T \subseteq C$, we have

$$\mathbb{P}[S \subseteq E, T \subseteq D] \leq p^{|S|} q^{|T|}. \tag{4}$$

If $q = 0$, i.e., there are no errors on the syndrome, then we talk of a local stochastic model of parameter p. In other words, the location of the errors is arbitrary but the probability of a given error decays exponentially with its weight.

In this paper, we study a variant of Algorithm 1 that allows us to deal with syndrome errors. This is Algorithm 2 below (see eqs. (1) and (5) for notations). The three differences with Algorithm 1 are the input $\sigma = \sigma_X(E) \oplus D$ instead of $\sigma = \sigma_X(E)$, the while loop condition $\Delta(\sigma_i, F_i) \geq \beta |\sigma_X(F_i)|$ instead of $\Delta(\sigma_i, F_i) > 0$ and the use of $\mathcal{F}$ instead of $\mathcal{F}_0$ as set of possible flips (see Remark 10 for a discussion about these changes):

$$\mathcal{F} := \left\{ F \subseteq \Gamma_Z(g) : g \in C_Z \text{ and } |\sigma_X(F)| \geq \frac{d_A}{2} |F| \right\}. \tag{5}$$

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

INPUT: $\sigma \subseteq C_X$ a syndrome such that $\sigma = \sigma_X(E) \oplus D$ for some (unknown) $E \subseteq V$ and $D \subseteq C_X$
OUTPUT: $\hat{E} \subseteq V$, a guess for the error pattern (alternatively, a set of qubits to correct)

```
Ê_0 = 0 ; σ_0 = σ ; i = 0
while ∃F_i ∈ F : Δ(σ_i, F_i) ≥ β|σ_X(F_i)| do
    Ê_{i+1} = Ê_i ⊕ F_i
    σ_{i+1} = σ_i ⊕ σ_X(F_i)        // σ_{i+1} = σ_X(E ⊕ Ê_{i+1}) ⊕ D
    i = i + 1
end while
return Ê_i
```

Remark 10. *In order to simplify our discussion in the paper, we will say that the input of Algorithm 2 is (E, D) when its input is $\sigma_X(E) \oplus D$, we will call $\hat{E}$ the output, we will call $E \oplus \hat{E}$ the remaining error, we will denote by f the number of steps and we will call $U = E \cup F_0 \cup \ldots \cup F_{f-1}$ the execution support.*

Using $\mathcal{F}$ instead of $\mathcal{F}_0$ in Algorithm 2 is not restrictive because if the condition $|\sigma_X(F)| \geq \frac{d_A}{2}|F|$ is not satisfied for some $F \subseteq \Gamma_Z(g)$ then this condition is satisfied by $F' = \Gamma_Z(g) \setminus F$ (see the proof of Lemma 18).

In Algorithm 1, the weaker "while loop condition" $\Delta(\sigma_i, F_i) > 0$ was used, but it turns out that if $D = \varnothing$ then with high probability on the choice of E, the condition $\Delta(\sigma_i, F_i) \geq (1 - 8\delta)|\sigma_X(F_i)|$ is automatically satisfied at each step of Algorithm 1 (this property was used in the proof of Theorem 8). On the other hand when $D \neq \varnothing$, requiring $\Delta(\sigma_i, F_i) \geq \beta |\sigma_X(F_i)|$ with β close to 1 makes Algorithm 2 more robust against syndrome errors.

The behavior of Algorithm 2 in the particular case where $D = \varnothing$ could be studied following the proof of [8]: given $\mathcal{Q}_G$ a quantum expander code constructed using G some bipartite (γ, δ)-expander graph with $\delta < 1/8$, it is possible to prove that a random error E is corrected with high probability by the small-set-flip algorithm of parameter β_0 (β_0 as defined in Section 3.1). In the noisy case $D \neq \varnothing$, we cannot hope to entirely correct the error because any single qubit error cannot be distinguished from a well-chosen constant weight syndrome bit error. But we will prove in Theorem 13 that when $\delta < 1/16$, the correction provided by the small-set-flip algorithm of parameter $\beta < \beta_1$ (β_1 as defined in Section 3.1) leads to a residual error that is local stochastic with controlled parameter.

2.5 Fault-tolerant protocol with quantum expander codes

The objective of this section is to provide a brief description of Gottesman's framework applied to quantum expander codes. A fault-tolerant protocol is a procedure that transforms an ideal circuit C into a fault-tolerant one $\tilde{\mathrm{C}}$ in a way that ensures that even if the individual components of $\tilde{\mathrm{C}}$ are noisy, the output of the $\tilde{\mathrm{C}}$ can be used to reproduce the output of C (in an approximate sense we will specify shortly).

Definition of a circuit We start by defining a circuit in this context. A circuit C can be described by local operations acting on a collection of (at most) k wires, each containing a qubit. More precisely, a circuit of depth d has d time steps and in each time step an operation may be applied to each one of the wires. Each such operation is called a *location* and it is even useful to also consider the operation of not doing anything on a wire as a "wait" location (this is used to model memory errors). As such, our circuit C will have the following types of locations acting on one of the k wires: wait, preparation of $|0\rangle$ (for an inactive wire), Hadamard gate, $R_{\pi/4}$ gate, $R_{\pi/8}$ gate where $R_\phi = \begin{pmatrix} e^{-i\phi} & 0 \\ 0 & e^{i\phi} \end{pmatrix}$, a measurement in the computational eigenbasis of Z (the corresponding wire then becomes inactive and the outcome is stored in a classical wire). In order to do useful computations, a two-qubit gate is needed, the controlled-not (CNOT) gate is a location acting on two of the wires. Another item is needed to describe the behavior of the circuit: classical computations on the classical wires of the circuit. As we are going to assume that the classical computation is not subject to faults, we will not describe explicitly the classical computations in terms of a circuit but rather it will be considered as a computation that takes place in between time steps. For this reason, it is important that the classical computations are fast, and this will be the case for us as all the classical computations can be done in parallel in $\mathcal{O}(\log k)$ time. The location present at any point in the circuit can be controlled by the classical wires. Note that circuit C does not take any input and all quantum wires are initialized with a state preparation $|0\rangle$, and the output is composed of some classical wires and some quantum wires. As we are most often interested in a classical result, we will assume the circuits ends with a measurement of all the qubits, and hence the output of the circuit is described a random variable $Y \in \{0,1\}^m$, with a probability distribution P_Y.

Noise model An ideal circuit will behave exactly as prescribed. We now would like to define a noise model, called *local stochastic model*, for a circuit. The subset of locations that are faulty is a random variable F with a distribution restricted by a parameter p. Namely, the set F of faulty locations has an arbitrary distribution that satisfies the property for any subset S of locations $\mathbb{P}[S \subseteq F] \leq p^{|S|}$. Then, the faulty locations are replaced by arbitrary quantum operations with the same input and output space as the original location. We also highlight some important assumptions that are implicit in our definition of circuit. Namely, it is possible to act on all the qubits in parallel. In

addition, it is possible to introduce fresh ancilla qubits at any point during the computation, *via* a prepare $|0\rangle$ location. Another assumption that was already mentioned is that classical computation is not subject to faults and is fast enough so that it can be done between the time steps of the main circuit.

If the noise model we just described is applied to a circuit C, in general, its output will have little to do with the ideal output P_Y of C. Our objective is to build a new circuit $\tilde{\mathrm{C}}$ which acts as a robust version of C. In particular, the output of the circuit $\tilde{\mathrm{C}}$ should have as output a random variable $\tilde{Y} \in \{0,1\}^{m'}$ that should behave as an *encoded* version of the ideal outcome Y in some error correcting code. More precisely, we want that there exists a function D (which should be efficiently computable) such that the distributions $P_{\mathrm{D}(\tilde{Y})}$ and P_Y are ε-close in total variation distance. For example, if we want to solve a decision problem and $Y \in \{0,1\}$, then the output $\tilde{Y}$ of the robust circuit will correspond to the ideal result except with probability at most ε.

Circuit transformation overview We start with a circuit C on k qubits with $f(k)$ locations, and we will assume that the circuit C is sequential, *i.e.*, at any time step, there is only one location which is not a wait location. Note that this can always be done at the cost of increasing the number of locations to at most $kf(k)$. The main error correcting code we use to construct $\tilde{\mathrm{C}}$ is a quantum expander code $\mathcal{Q}$ with parameters $[[n', k']]$ with k' chosen later and $n' = \frac{k'}{R}$, where R is the rate of the code (in the notation of Theorem 1, R should be chosen to be basically $1/\eta$). The general structure of the circuit $\tilde{\mathrm{C}}$ is as follows. The k qubits of C are partitioned into blocks, $B_1, \ldots, B_\ell$ with each block containing at most $k' = \frac{k}{\ell}$ with $\ell = \log^c k$ and c a well-chosen constant (for simplicity, we assume that k is divisible by ℓ). The circuit $\tilde{\mathrm{C}}$ is going to contain blocks $\tilde{B}_j$ each having n' physical qubits and corresponding to the block B_j encoded using the code $\mathcal{Q}$. The circuit $\tilde{\mathrm{C}}$ is going to alternate between two types of cycles: a simulation cycle and an error correction cycle. In an error correction cycle, we perform a measurement of the syndromes for the blocks $\tilde{B}_j$ for all j in parallel, then use the small-set-flip decoding algorithm to determine the error and apply the corresponding correction to each block. In a simulation cycle, we simulate an action of the circuit C. Recall that we assumed that at each time step of C, there is only one location which is not a wait location. The objective will be to simulate this location. Note that in a simulation cycle, only one block $\tilde{B}_j$ will be involved if there are no locations that are across different blocks or two blocks $\tilde{B}_{j_1}$ and $\tilde{B}_{j_2}$ if the location is a CNOT between two qubits, one in the block B_{j_1} and one in B_{j_2}. Recall that our objective is to obtain that under the noise model described above, the output is ε-close to an ideal output. For this reason, we define $\varepsilon_0 = \frac{\varepsilon}{f(k)}$ which can be interpreted as an allowed failure rate per location, for a logical operation.

In the following paragraphs, we give a bit more details on how these cycles are performed. We refer the reader to [14] for the analysis (and other variants that could be used).

Error correction cycle

For each block, we perform a measurement of the syndrome. For this, we use $n' - k'$ new qubits, one for each generator, prepared using a "prepare $|0\rangle$" location. For a given generator (which has a constant weight at most r), we start by applying a Hadamard gate, then we apply controlled-X (*i.e.*, CNOT) and controlled-Z (*i.e.*, CNOT conjugated by Hadamard on the target qubit) gates for each non-identity element of the generator, then a Hadamard gate and a measurement. If the gates and measurements are perfect, this is a measurement of the generator. In the presence of noise, one can still bound the propagation of errors because each generator measurement acts on a constant number of qubits and each qubit is involved in a constant number of generators, thanks to the LDPC character of the quantum code. This measurement is performed for each generator of the code $\mathcal{Q}$. We will perform in parallel measurements for generators that act on disjoint qubits (for

example using the partition described in Section 4). And all the blocks $\tilde{B}_j$ are treated in parallel.

Once we obtain the results of the measurements, we then apply the small-set-flip decoding algorithm (which is a classical algorithm) to determine the error pattern on the physical qubits and then apply the corresponding corrections.

Cost of an error correction cycle In terms of memory, note that the number of new qubits used for this step is $(n' - k')\ell = (\frac{1}{R} - 1)k$. We remark that we could perform the error correction less often in order to reduce the memory overhead. For example, for an integer s, at the p-th error correction cycle we could only perform error correction for the blocks $\tilde{B}_j$ where $j = p \mod s$. In this case, the number of qubits used for the measurement is at most $\lceil(\frac{1}{R} - 1)\frac{k}{s}\rceil$. This will be at the cost of decreasing the allowed probability of error per location, but for any constant s, the allowed error probability per location will remain constant.

In terms of time, the number of time steps (or depth) in the quantum circuit $\tilde{\mathrm{C}}$ is constant. In addition, the classical computation time for the small-set-flip decoding algorithm can be done in time $\mathcal{O}(\log k)$ (see Section 4).

Simulation cycle

For a simulation cycle, the structure of the operations we will apply is as follows. We use new qubits to create a well-chosen ancilla state (depending on the type of location we want to simulate), then perform a Bell measurement between the relevant block(s) and the ancilla state and then perform some corrections. In order to construct the well-chosen ancilla state $|\Psi\rangle$, we use the following lemma, which is proved in [14].

Lemma 11. *Given $\mathcal{D}$ a quantum circuit with output given by the m-qubit state $|\Psi\rangle$ and any $\delta > 0$, there exists a quantum circuit $\mathcal{D}'$ satisfying the following properties.*

- *If the circuit undergoes local stochastic noise with sufficiently small parameter p, then there exists a failure event that has probability at most δ and conditioned on the failure event not happening, the output of the circuit is described by $|\Psi\rangle$ to which a local stochastic error is applied.*
- *The number of qubits in the circuit $\mathcal{D}'$ is at most $c_0 m \log^{c_1}(|\mathcal{D}|/\delta))$ where $|\mathcal{D}|$ is the number of locations in $\mathcal{D}$, for some constants c_0, c_1.*
- *When $\mathcal{D}$ is of depth d and the classical computation takes t time steps, the depth of $\mathcal{D}'$ is $\mathcal{O}(d + \log(\log(|\mathcal{D}|/\delta)))$ and the classical computation time is $\mathcal{O}(t + \log(\log(|\mathcal{D}|/\delta)))$.*

For each time step in C, we simulate a single location which is not a wait location. If this location affects one qubit, we assume for simplicity of notation that it acts on the first qubit of block B_j. Note that we always take $\delta = \varepsilon_0 := \frac{\varepsilon}{f(k)}$ when applying Lemma 11. We consider the different types of location case by case.

- To simulate a preparation of $|0\rangle$, if the block $\tilde{B}_j$ was already created, then there is nothing we need to do. If the block $\tilde{B}_j$ does not exist yet, we will create it prepared in the state $|0\rangle^{\otimes k'}$ encoded in the code $\mathcal{Q}$. In order to do this, we apply Lemma 11 with the circuit $\mathcal{D}$ being an encoding circuit for the code $\mathcal{Q}$ initialized in the state $|0\rangle^{\otimes k'}$.
- To simulate a Hadamard (an $R_{\pi/4}$ gate is similar), we will apply gate teleportation by preparing an ancilla state as an encoded entangled pair to which H is applied. More precisely, we apply Lemma 11 with the circuit $\mathcal{D}$ being two parallel encoding circuits for the code $\mathcal{Q}$ with input initialized to $(I \otimes H)\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \otimes (\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle))^{\otimes(k'-1)}$. As a result, we

obtain a state $|\Psi\rangle \in (\mathbb{C}^2)^{\otimes 2n'}$. We call D_1 and D_2 the two blocks of n' qubits. We then perform n' Bell measurements in parallel on physical qubits of blocks $\tilde{B}_j$ and D_1. This allows us to deduce the outcome of the Bell measurement on the first logical qubit encoded in $\tilde{B}_j$ together with the first logical qubit encoded in D_1. Depending on the outcome, a Pauli correction should be on the first logical qubit of D_2, which can be done transversally by applying some Pauli operators on each qubit of D_2. The block D_2 then takes the place of the block $\tilde{B}_j$.

- To simulate a $R_{\pi/8}$-gate, we similarly apply Lemma 11 with the circuit $\mathcal{D}$ being two parallel encoding circuits for the code $\mathcal{Q}$ with input initialized to $(I \otimes R_{\pi/8})\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \otimes (\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle))^{\otimes(k'-1)}$. We similarly apply the Bell measurement between the blocks $\tilde{B}_j$ and D_1, but the difficulty now is that the possible correction operations are Z and $XR_{\pi/4}$. In order to apply a logical $R_{\pi/4}$ gate, we prepare another pair of ancilla blocks D_3 and D_4 and use the same procedure as described above to prepare it and apply it.

- If the location is a CNOT gate between qubits in blocks B_{j_1} and B_{j_2}, assuming for simplicity of notation that the CNOT gate is between the first qubit of B_{j_1} and the first qubit of B_{j_2}, we will create four blocks D_1, D_2, D_3, D_4, each containing n' physical qubits and use Lemma 11 to prepare the following state: $(I_{D_1D_2} \otimes \mathrm{CNOT}_{D_3D_4})\frac{1}{2}(|00\rangle_{D_1D_3} + |11\rangle_{D_1D_3}) \otimes (|00\rangle_{D_2D_4} + |11\rangle_{D_2D_4}) \otimes (\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle))^{\otimes(k'-2)}$. Then, as before, we perform a measurement in the Bell basis corresponding to logical qubit 1 of B_{j_1} with logical qubit 1 of D_1 and another in the Bell basis corresponding to logical qubit 1 of B_{j_2} with logical qubit 1 of D_2 and apply the Pauli corrections to the blocks D_3 and D_4. The blocks D_3 and D_4 then play the role of $\tilde{D}_{j_1}$ and $\tilde{D}_{j_2}$. For a CNOT gate between qubits within the same block B_j, the construction is similar but slightly simpler as we only need two ancillary blocks.

- To simulate a measurement, we prepare an ancilla state in the encoded $|0\rangle^{\otimes k'}$ state, apply a logical CNOT between the first logical qubit of $\tilde{B}_j$ and the first logical qubit of the ancilla block (using the method described above), then measure all the physical qubits of the ancilla block getting a bit-string of length n', and finally use the small-set-flip decoding algorithm on the Z-type generators (this corresponds to a classical parity check matrix) on the classical outcome to get an encoded bit-string. The first logical bit encoded in this bit-string corresponds to the outcome of the measurement we are simulating.

Cost of a simulation cycle In terms of memory, for all the location types, we prepare a state $|\Psi\rangle$ on $\mathcal{O}(n')$ qubits, and so $\mathcal{O}(n' \log^{c_1}(n'/\varepsilon_0))$ qubits. By the choice of k', this quantity is sub-linear in the number of qubits k.

In terms of time, the number of time steps added to the quantum circuit $\tilde{\mathrm{C}}$ is $\mathcal{O}(\log\log f(k))$ and the classical time is $\mathcal{O}(\log k)$.

3 Analysis of Algorithm 2

3.1 Notations

The algorithms of Section 3 and Section 4 depend on three parameters $\delta, \beta \in (0;1)$ and $c \in \mathbb{R}_+^*$. Note that the parameter c and the constants χ, c_3, η and f_0 defined below are used in Section 4 for the analysis of the parallel version of the algorithm but they are not used in Section 3.
We consider $G = (A \cup B, \mathcal{E})$ a (d_A, d_B)-biregular (γ, δ)-expander graph with $\gamma > 0$, we denote by $\mathcal{Q}$ the quantum expander code associated to G (see Section 2.3) and by V, C_X, C_Z and $n := |V|$ respectively the set of qubits, the set of Z-type stabilizer generators, the set of X-type stabilizer

generators and the number of physical qubits of $\mathcal{Q}$. We will also use Γ_X and Γ_Z the neighborhoods in the graphs G_X ans G_Z as defined in Section 2.3.
We run the small-set-flip decoding algorithm (Algorithm 2) of parameter β on input (E, D) where $E \subseteq V$ represents a qubit error and $D \subseteq C_X$ represents a syndrome error, we denote by $\hat{E}$ the output of the algorithm, by f the number of steps and by $U = E \cup F_0 \cup \ldots \cup F_{f-1}$ the execution support.

We also define the constants:

$$r := d_A/d_B, \qquad \gamma_0 = \frac{r^2}{\sqrt{1+r^2}}\gamma, \qquad \beta_0 = \beta_0(\delta) := 1 - 8\delta, \qquad \beta_1 = \beta_1(\delta) := 1 - 16\delta,$$

$$c_0 = c_0(\delta, \beta) := \frac{4}{d_A(\beta_1 - \beta)}, \qquad c_1 = c_1(\delta, \beta) := \frac{\beta_1 - \beta}{\beta_0(1-\beta)},$$

$$c_2 = c_2(\delta, \beta) := \frac{2\beta_0}{\beta_1 - \beta}, \qquad c_3 = c_3(\beta, c) := \frac{2(1+c)}{\beta d_A},$$

$$\chi := (d_B(d_A - 1) + 1)(d_A(d_B - 1) + 1),$$

$$\alpha_0 = \alpha_0(\beta) := \frac{r\beta}{4 + 2r\beta}, \qquad \eta = \eta(\delta, \beta, c) := 1 - \frac{\beta c_1(c - 1 - c_2)}{d_A d_B \chi c}.$$

We also define the function (s will represents the size of the input syndrome for the parallel algorithm Algorithm 3):

$$f_0 = f_0(\delta, \beta, c) : s \in \mathbb{N} \mapsto \left\lceil \chi \log_{1/\eta}(s) \right\rceil.$$

Note that:

- If $\delta < 1/8$ then $\beta_0 > 0$.
- If $\delta < 1/16$ then $\beta_1 > 0$.
- If $\delta < 1/16$ and $0 < \beta < \beta_1$ then $c_0, c_1, c_2, c_3 > 0$ and $0 < \alpha_0 \leq 1$.
- If $\delta < 1/16$, $0 < \beta < \beta_1$ and $c > c_2 + 1$ then $\eta < 1$ and $f_0(s) = \Theta(\log(s))$.

3.2 Statements of the theorems

As discussed previously, the noiseless case $D = \varnothing$ was studied in [8]:

Theorem 12 ([8])**.** *We use the notations of Section 3.1 with $\delta < 1/8$.*
With probability at least $1 - e^{-\Omega(\sqrt{n})}$ on the choice of the local stochastic error E, the small-set-flip algorithm (Algorithm 2) with parameter β_0 outputs some error $\hat{E}$ equivalent to E under the assumption $D = \varnothing$.

In this section, we are going to prove Theorem 13, a generalized version of Theorem 12 that we can apply in the case where the syndrome error $D \subseteq C_X$ is not empty.

Theorem 13. *We use the notations of Section 3.1 with $\delta < 1/16$ and $\beta < \beta_1$.*
There exist constants $p_0 > 0, p_1 > 0$ such that the following holds. Suppose the pair (E, D) satisfies a local stochastic noise model with parameter $(p_{\text{phys}}, p_{\text{synd}})$ where $p_{\text{phys}} < p_0$ and $p_{\text{synd}} < p_1$. Then there exists an event succ *that has probability $1 - e^{-\Omega(\sqrt{n})}$ and a random variable E_{ls} that is equivalent to $E \oplus \hat{E}$ such that conditioned on* succ*, E_{ls} has a local stochastic distribution with parameter $K p_{\text{synd}}^{1/c_0}$ where K is a constant independent of p_{synd}.*

3.3 Small adversarial errors

The first step to prove Theorem 13 is to study the case where the qubit error E can be adversarial but where $E \oplus \hat{E}$ is supposed to be reduced with $|E \oplus \hat{E}| \leq \gamma_0\sqrt{n}$. Here "reduced" means that $E \oplus \hat{E}$ has the smallest Hamming weight among all errors equivalent to E. The result in that case is summarized in Corollary 16: it is possible to use expansion-based arguments to find an upper bound on $|E \oplus \hat{E}|$ which grows linearly with $|D \cap \sigma_X(E \oplus \hat{E})|$. Corollary 16 is a consequence of Proposition 14 and Lemma 15 that we state now but only prove at the end of this section.

Proposition 14. *We use the notations of Section 3.1 with $\delta < 1/16$ and $\beta \in (0; \beta_1)$.*
If $|E| \leq \gamma_0\sqrt{n}$ and $|\sigma_X(E)| > c_2|D \cap \sigma_X(E)|$ then there exists at least one valid $F \in \mathcal{F}$ for Algorithm 2 with parameter β.
More precisely, let $\sigma = \sigma_X(E) \oplus D$ then the set $G := \{F \in \mathcal{F} : \Delta(\sigma, F) \geq \beta|\sigma_X(F)|\}$ satisfies:

$$\sum_{F \in G} |\sigma_X(F)| \geq c_1 \left[|\sigma_X(E)| - c_2|D \cap \sigma_X(E)|\right] > 0.$$

Lemma 15 (Robustness)**.** *We use the notations of Section 3.1 with $\delta < 1/8$.*
If $E_R \subseteq V$ is a reduced error with $|E_R| \leq \gamma_0\sqrt{n}$ then:

$$|\sigma_X(E_R)| \geq \frac{\beta_0 d_A}{2}|E_R|.$$

Together, Proposition 14 and Lemma 15 imply the following:

Corollary 16. *We use the notations of Section 3.1 with $\delta < 1/16$ and $\beta \in (0; \beta_1)$.*
Suppose that $E \oplus \hat{E}$ is reduced with $|E \oplus \hat{E}| \leq \gamma_0\sqrt{n}$ then

$$|E \oplus \hat{E}| \leq c_0|D \cap \sigma_X(E \oplus \hat{E})|.$$

Proof. Using the notations σ_i from the body of Algorithm 2, the value of the syndrome σ_f at the end of the algorithm is $\sigma_f = \sigma_X(E \oplus \hat{E}) \oplus D$. Since the while loop condition is not satisfied for σ_f, the contraposition of Proposition 14 ensures that $|\sigma_X(E \oplus \hat{E})| \leq c_2|D \cap \sigma_X(E \oplus \hat{E})|$. By Lemma 15, $|\sigma_X(E \oplus \hat{E})| \geq \frac{\beta_0 d_A}{2}|E \oplus \hat{E}|$ which concludes the proof. □

The rest of Section 3.3 is devoted to prove Proposition 14 and Lemma 15.
We will study the sets $F \in G$ (G is defined in Proposition 14) which would have been flipped during the small-set-flip algorithm with input E and $D = \varnothing$, *i.e.*, without syndrome error. For a given set $E \subseteq V = A^2 \uplus B^2$ (where $\uplus$ stands for disjoint union), we introduce a notation for a normalized Hamming weight:

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

$\|\cdot\|$ shares a couple of properties with the usual the cardinality $|\cdot|$. In particular it is straightforward to check that for $E, E_1, E_2 \subseteq V$:

$$\begin{aligned}
&\|E\| = 0 \Leftrightarrow E = \varnothing, && d_A\|E\| \leq |E| \leq d_B\|E\|,\\
&|\sigma_X(E)| \leq d_A d_B\|E\|, && \|E_1 \cup E_2\| \leq \|E_1\| + \|E_2\|,\\
&\|E_1 \uplus E_2\| = \|E_1\| + \|E_2\|, && \|E_1 \oplus E_2\| = \|E_1\| + \|E_2\| - 2\|E_1 \cap E_2\|.
\end{aligned}$$

We will say that a qubit error $E \subseteq V$ is $\|\ \|$-reduced when $\|E\|$ is minimal over $E + \mathcal{C}_Z{}^{\perp}$. All along this section we will use the handy property of Lemma 17:

Lemma 17. *Let $E_1 \subseteq E_2 \subseteq V$ be two errors. If E_2 is reduced (resp. $\|\ \|$-reduced) then E_1 is reduced (resp. $\|\ \|$-reduced).*

Proof. Let's prove that for any $E \in \mathcal{C}_Z^\perp, |E_1| \leq |E_1 \oplus E|$. Using the fact that $|\cdot|$ is additive for disjoint unions and non-negative, we have

$$\begin{aligned} 0 \leq |E_2 \oplus E| - |E_2| = |E - E_2| + |E_2 - E| - |E_2 \cap E| - |E_2 - E| = |E| - 2|E_2 \cap E| \\ \leq |E| - 2|E_1 \cap E| = |E_1 \oplus E| - |E_1|. \end{aligned}$$

As $\|\cdot\|$ is also additive for disjoint unions and non-negative, the same proof also works when replacing $\|\cdot\|$ with $|\cdot|$. □

First of all, we need to study the case where the syndrome is noiseless ($D = \varnothing$). In that case and when the initial graph G is sufficiently expanding, there exists at least one X-type stabilizer generator called a "critical generator" (this notion was introduced in [18]) whose support contains some $F \in \mathcal{F}$ that decreases the syndrome weight when flipped.

Lemma 18 (Lemma 8 of [18] revisited)**.** *Let $E \subseteq V$ be a $\|\ \|$-reduced error such that $0 < \|E\| \leq \gamma_0\sqrt{n}/d_A$, then there exists $F \in \mathcal{F}$ with $F \subseteq E$ and:*

(i) $|\sigma_X(F)| \geq \frac{1}{2} d_A d_B \|F\|$,

(ii) $\Delta(\sigma_X(E), F) \geq |\sigma_X(F)| - 4\delta d_A d_B \|F\|$.

Proof. We have:

$$|E| \leq d_B\|E\| \leq \gamma_0\sqrt{n}/r = \gamma\frac{r}{\sqrt{1+r^2}}\sqrt{n} = \gamma\frac{r}{\sqrt{1+r^2}}\sqrt{n_A^2 + n_B^2} = \gamma n_B = \min(\gamma n_A, \gamma n_B).$$

Since $|E| \leq \min(\gamma n_A, \gamma n_B)$, Lemma 7 of [18] states that there exists $g \in C_Z$ called a "critical generator" whose neighborhood satisfies $\Gamma_Z(g) = x_a \uplus \overline{x}_a \uplus \chi_a \uplus x_b \uplus \overline{x}_b \uplus \chi_b$ with:

$$\begin{array}{llll} x_a \subseteq A^2 \cap E, & x_b \subseteq B^2 \cap E, & x_a \cup x_b \neq \varnothing, & \\ \overline{x}_a \subseteq A^2 \setminus E, & \overline{x}_b \subseteq B^2 \setminus E, & & \\ \chi_a \subseteq A^2, & \chi_b \subseteq B^2, & \|\chi_a\| \leq 2\delta, & \|\chi_b\| \leq 2\delta, \end{array}$$

and such that the Z-type generators of C_X involving qubits from $x_a \uplus \overline{x}_a \uplus x_b \uplus \overline{x}_b$ do not involve other qubits from E. Formally, for $v_a \in x_a, v_b \in x_b, \overline{v}_a \in \overline{x}_a$ and $\overline{v}_b \in \overline{x}_b$:

$$\begin{array}{ll} E \cap \Gamma_X[\Gamma_X(v_a) \cap \Gamma_X(v_b)] = \{v_a, v_b\}, & E \cap \Gamma_X[\Gamma_X(\overline{v}_a) \cap \Gamma_X(\overline{v}_b)] = \varnothing, \\ E \cap \Gamma_X[\Gamma_X(v_a) \cap \Gamma_X(\overline{v}_b)] = \{v_a\}, & E \cap \Gamma_X[\Gamma_X(\overline{v}_a) \cap \Gamma_X(v_b)] = \{v_b\}. \end{array}$$

Take $F = x_a \uplus x_b$ with the purpose to get $F \subseteq E$. We have:

$$\begin{aligned} |\sigma_X(F)| &= |x_a|(|\overline{x}_b| + |\chi_b|) + |x_b|(|\overline{x}_a| + |\chi_a|) \\ &= d_A d_B(\|x_a\|(\|\overline{x}_b\| + \|\chi_b\|) + \|x_b\|(\|\overline{x}_a\| + \|\chi_a\|)) \\ &= d_A d_B(\|x_a\|(1 - \|x_b\|) + \|x_b\|(1 - \|x_a\|)). \end{aligned}$$

But $\|F\| = \|x_a\| + \|x_b\|$, thus:

$$\frac{|\sigma_X(F)|}{d_A d_B \|F\|} = 1 - \frac{2\|x_a\|\|x_b\|}{\|x_a\| + \|x_b\|}. \tag{6}$$

Note that we have $0 \leq \|x_a\|, \|x_b\| \leq 1$ by definition, we have $0 < \|x_a\| + \|x_b\|$ because $x_a \cup x_b \neq \varnothing$ and we have $\|x_a\| + \|x_b\| \leq 1$ because E is $\|\ \|$-reduced. With these constraints on $\|x_a\|$ and $\|x_b\|$, a function analysis of the right hand side of eq. (6) gives item (i) and $F \in \mathcal{F}$.
For item (ii), we lower bound $\Delta(\sigma_X(E), F)$ using the inequalities $\|\chi_a\| \leq 2\delta$ and $\|\chi_b\| \leq 2\delta$:

$$\begin{aligned}\Delta(\sigma_X(E), F) &\geq |x_a||\overline{x}_b| + |x_b||\overline{x}_a| - |x_a||\chi_b| - |x_b||\chi_a| \\ &= |\sigma_X(F)| - 2d_A d_B(\|x_a\|\|\chi_b\| + \|x_b\|\|\chi_a\|) \\ &\geq |\sigma_X(F)| - 4\delta d_A d_B(\|x_a\| + \|x_b\|) \\ &= |\sigma_X(F)| - 4\delta d_A d_B\|F\|.\end{aligned}$$

□

Proof of Proposition 14 and Lemma 15. Both proofs begin in the same way: we set E_0 to be the $\|\ \|$-reduced error equivalent to E (or equivalent to E_{R} in the case of Lemma 15), we apply Lemma 18 to E_0 which provides some $F_0 \subseteq E_0$ and we define $E_1 := E_0 \oplus F_0 = E_0 \setminus F_0$. More generally, we set by induction $E_{i+1} = E_i \oplus F_i = E_i \setminus F_i$ where F_i is obtained by applying Lemma 18 to E_i. This construction is licit (*i.e.*, we can apply Lemma 18 to E_i) because E_i is $\|\ \|$-reduced as a subset of the $\|\ \|$-reduced error E_0 (see Lemma 17), and $\|E_i\| \leq \|E_0\| \leq \|E\| \leq |E|/d_A \leq \gamma_0\sqrt{n}/d_A$. Let f' be the last step of this procedure then $\|E_{f'}\| = 0$ and thus:

$$E_0 = \biguplus_{i=0}^{f'-1} F_i. \tag{7}$$

Since the sets F_i are those given in Lemma 18, we can use Lemma 18 items (i) and (ii) to lower bound $|\sigma_X(E_0)|$:

$$|\sigma_X(E_0)| = \sum_{i=0}^{f'-1} \Delta(\sigma_X(E_i), F_i) \geq \sum_{i=0}^{f'-1} |\sigma_X(F_i)| - \sum_{i=0}^{f'-1} 4\delta d_A d_B\|F_i\| \geq \frac{\beta_0 d_A d_B}{2} \sum_{i=0}^{f'-1} \|F_i\|. \tag{8}$$

Because $E_0 = \biguplus_i F_i$ (eq. (7)), we have:

$$|\sigma_X(E_0)| \geq \frac{\beta_0 d_A d_B}{2}\|E_0\|. \tag{9}$$

The above arguments hold for Proposition 14 as well as for Lemma 15. Proving Lemma 15 is now direct:

$$|\sigma_X(E_{\mathrm{R}})| = |\sigma_X(E_0)| \geq \frac{\beta_0 d_A d_B}{2}\|E_0\| \geq \frac{\beta_0 d_A}{2}|E_0| \geq \frac{\beta_0 d_A}{2}|E_{\mathrm{R}}|.$$

For Proposition 14, let us start by providing an overview of how we will proceed. Note that a union bound yields $|\sigma_X(E)| = |\sigma_X(E_0)| \leq \sum_{i=0}^{f'-1} |\sigma_X(F_i)|$. In fact, we will prove in eq. (10) below that that this upper bound is nearly tight: $|\sigma_X(E)| \geq \beta_0 \sum_{i=0}^{f'-1} |\sigma_X(F_i)|$ and β_0 is arbitrarily close to 1 when δ is small. Intuitively, this means that the intersection of the sets $\sigma_X(F_i)$ is small and thus $|\sigma_X(E) \cap \sigma_X(F_i)|$ is generally large. This is still true if the size of the syndrome error D is small, *i.e.*, it holds that $|\sigma \cap \sigma_X(F_i)|$ is generally large. Hence we will obtain Proposition 14 by

computing the average of the quantity $|\sigma \cap \sigma_X(F_i)|$ over the sets F_i. We now provide the details:

$$\begin{aligned}
|\sigma_X(E)| = |\sigma_X(E_0)| &\geq \sum_{i=0}^{f'-1} |\sigma_X(F_i)| - \sum_{i=0}^{f'-1} 4\delta d_A d_B \|F_i\| && \text{by eq. (8)} \\
&= \sum_{i=0}^{f'-1} |\sigma_X(F_i)| - 4\delta d_A d_B \|E_0\| && \text{by eq. (7)} \\
&\geq \sum_{i=0}^{f'-1} |\sigma_X(F_i)| - \frac{8\delta}{\beta_0} |\sigma_X(E)| && \text{by eq. (9).}
\end{aligned}$$

Hence we have:

$$\left(1 + \frac{8\delta}{\beta_0}\right) |\sigma_X(E)| \geq \sum_{i=0}^{f'-1} |\sigma_X(F_i)|. \tag{10}$$

The relation between $\Delta(\sigma, F)$ and $|\sigma \cap \sigma_X(F)|$ is given by:

$$\Delta(\sigma, F) = |\sigma| - |\sigma \oplus \sigma_X(F)| = 2\,|\sigma \cap \sigma_X(F)| - |\sigma_X(F)|$$

where we have used the equality $|A_1 \oplus A_2| = |A_1| + |A_2| - 2|A_1 \cap A_2|$.
In particular, when $F \notin G$:

$$|\sigma \cap \sigma_X(F)| \leq \frac{1+\beta}{2} |\sigma_X(F)|. \tag{11}$$

On the one hand, eqs. (10) and (11) give an upper bound on the sum $S := \sum_{i=0}^{f'-1} |\sigma \cap \sigma_X(F_i)|$:

$$\begin{aligned}
S &= \sum_{F_i \in G} |\sigma \cap \sigma_X(F_i)| + \sum_{F_i \notin G} |\sigma \cap \sigma_X(F_i)| \\
&\leq \sum_{F_i \in G} |\sigma_X(F_i)| + \frac{1+\beta}{2} \sum_{F_i \notin G} |\sigma_X(F_i)| && \text{by eq. (11)} \\
&= \frac{1-\beta}{2} \sum_{F_i \in G} |\sigma_X(F_i)| + \frac{1+\beta}{2} \sum_{i=0}^{f'-1} |\sigma_X(F_i)| \\
&\leq \frac{1-\beta}{2} \sum_{F_i \in G} |\sigma_X(F_i)| + \frac{1+\beta}{2\beta_0} |\sigma_X(E)| && \text{by eq. (10).}
\end{aligned}$$

On the other hand, $E_0 = \biguplus_i F_i$ (eq. (7)) implies that $\sigma_X(E) = \sigma_X(E_0) = \bigoplus_{i=0}^{f'-1} \sigma_X(F_i)$ and thus S is lower bounded by:

$$\begin{aligned}
S &\geq |\sigma \cap \sigma_X(E)| \\
&= |(\sigma_X(E) \oplus D) \cap \sigma_X(E)| \\
&= |\sigma_X(E) \oplus (D \cap \sigma_X(E))| \\
&= |\sigma_X(E)| - |D \cap \sigma_X(E)|.
\end{aligned}$$

Combining both inequalities we get Proposition 14. □

3.4 Random errors of linear size

The upper bound given by Proposition 14 can be applied for qubit errors of size up to $t = \mathcal{O}(\sqrt{n})$. In the case of a local stochastic noise, the errors have a typical size of $\Theta(n)$. The relationship between the two frameworks is given by percolation arguments close to the arguments used in [8].

Percolation arguments will allow us to decompose a random error E as a disjoint union of small error sets, each of which has size upper bounded by t. The study of these small errors has been done in Section 3.3 and a consequence of Corollary 16 is that when we use Algorithm 2 to correct it, the remaining error is local stochastic. Moreover, Algorithm 2 is intuitively local in the sense that two qubit errors far away in the factor graph of the code will not interact during the decoding procedure. Under the assumption that a local stochastic error is produced for each small error set, the locality property will allow us to conclude that when we correct the initial error E with Algorithm 2, the remaining error has a local stochastic distribution.
In order to formalize the notion of locality, we define $\mathcal{G}$ called the syndrome adjacency graph of the code in the following way: $\mathcal{G}$ is equal to G_X (as defined in Section 2.3) with additional edges between the qubits which share an X-type or a Z-type generator. In other words, the set of vertices of $\mathcal{G}$ is indexed by $\mathcal{V} := V \cup C_X$ the set of qubits and the set of Z-type generators, a Z-type generator is incident to the qubits in its support and two qubits are linked when they are both in the support of the same generator. Note that the degree of $\mathcal{G}$ is upper bounded by $d := d_B(d_B + 2d_A - 1)$. Using the execution support $U = E \cup F_0 \cup \ldots \cup F_{f-1}$, it is easy to decompose the error into small error sets: each connected component K of U provides one error set $K \cap E$.

Lemma 19 (Locality of Algorithm 2, [8])**.** *We use the notations of Section 3.1.*
For any set $K \subseteq V$ with $\Gamma(K) \cap \Gamma(U \setminus K) = \varnothing$ in $\mathcal{G}$, there is a valid execution of Algorithm 2 on the input $(E \cap K, D \cap \Gamma_X(K))$ whose output is $\hat{E} \cap K$ and whose support is $U \cap K$.

What is the size of the remaining error $(E \oplus \hat{E}) \cap K$? We will show that this size is small enough to apply Corollary 16. The key point is to note that among the vertices of $X := K \cup (D \cap \Gamma_X(K)) \subseteq \mathcal{V}$, there is at least a fraction $2\alpha_0$ of these vertices which belong to $E \cup D$ (α_0 is defined in Section 3.1). We will say that X is a $2\alpha_0$-subset of $E \cup D$ (see Definition 20) and percolation arguments (see Lemma 27) will show that with high probability, any connected α-subset of a random error $E \cup D$ must be small enough to apply Corollary 16.

Definition 20 ([8])**.** *Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph, let $\alpha \in (0; 1]$ and let $E, X \subseteq \mathcal{V}$. X is said to be an* α-subset of E *if $|X \cap E| \geq \alpha |X|$. We also define the integer $\mathit{MaxConn}_\alpha(E)$ by:*

$$\mathit{MaxConn}_\alpha(E) = \max\{|X| : X \text{ is a connected in } \mathcal{G} \text{ and is an } \alpha\text{-subset of } E\}$$

This notion of α-subset is relevant because if we run the small-set-flip decoding algorithm of parameter $\beta > 0$ and set $U = E \cup F_0 \cup \ldots \cup F_{f-1}$ to be the execution support then $U \cup D$ is a $2\alpha_0$-subset of $E \cup D$.

Later we will need the following technical lemma in order to reduce to the case where the remaining error $E \oplus \hat{E}$ is reduced:

Lemma 21. *Let $E, X_1, X_2 \subseteq \mathcal{V}$ with $|X_2| \leq |X_1|$. If X_1 is an α-subset of E then $X_1 \cup X_2$ is an $\frac{\alpha}{2}$-subset of E.*

Proof. By assumption $|X_1| \leq \frac{1}{\alpha}|X_1 \cap E|$ thus:

$$|X_1 \cup X_2| \leq 2|X_1| \leq \frac{2}{\alpha}|X_1 \cap E| \leq \frac{2}{\alpha}|(X_1 \cup X_2) \cap E|.$$

□

Proposition 22. *Using the notations of Section 3.1, $U \cup D$ is a $2\alpha_0$-subset of $E \cup D$.*

Proof. Using the notations σ_i and F_i from the body of Algorithm 2, we have $|\sigma_i| - |\sigma_{i+1}| \geq \beta|\sigma_X(F_i)| \geq \frac{\beta d_A}{2}|F_i|$ thus we can lower bound $|\sigma_X(E) \oplus D| = |\sigma_0|$ by:

$$|\sigma_X(E) \oplus D| \geq |\sigma_0| - |\sigma_f| = \sum_{i=0}^{f-1} |\sigma_i| - |\sigma_{i+1}| \geq \frac{\beta d_A}{2} \sum_{i=0}^{f-1} |F_i|.$$

But $|\sigma_X(E) \oplus D| \leq d_B|E| + |D| \leq d_B|E \cup D|$ thus $\sum_i |F_i| \leq \frac{2}{\beta r}|E \cup D|$ and:

$$|U \cup D| \leq |E \cup D| + \sum_{i=0}^{f-1} |F_i| \leq \frac{1}{2\alpha_0}|E \cup D| = \frac{1}{2\alpha_0}|(U \cup D) \cap (E \cup D)|.$$

□

In order to prove Theorem 13, we would like to show that with probability at least $1 - e^{-\Omega(\sqrt{n})}$, there is a reduced error E_{ls} equivalent to $E \oplus \hat{E}$ which is local stochastic.

Recall from Definition 9 that an error E_{ls} is local stochastic with parameter p if we can upper bound the probability $\mathbb{P}[S \subseteq E_{\text{ls}}]$ by $p^{|S|}$ for all $S \subseteq V$. The reason why the probability p provided by Theorem 13 depends on p_{synd} is that we will use the hypothesis that D is local stochastic: $\mathbb{P}[T \subseteq D] \leq p_{\text{synd}}^{|T|}$ for all $T \subseteq C_X$. In order to establish Theorem 13, it would be sufficient to prove that for all $S \subseteq E_{\text{ls}}$, the size of $D \cap \Gamma_X(S)$ is lower bounded by a linear function of $|S|$. The particular case where $S = E_{\text{ls}} = E \oplus \hat{E}$ has already been proven in Corollary 16 but unfortunately it is not possible to prove this property for all $S \subseteq E_{\text{ls}}$. Indeed, for an initial error $E = \varnothing$ and $S = \hat{E} \setminus \Gamma_X(D)$, such a property would imply that $S = \varnothing$, *i.e.*, $\hat{E} \subseteq \Gamma_X(D)$. The last statement is clearly false because on input (E, D) with $E = \varnothing$, Algorithm 2 could flip some qubit which is not in $\Gamma_X(D)$ (note, however, that such a flip would necessarily happen after several steps).

Therefore, instead of lower bounding $|D \cap \Gamma_X(S)|$ linearly in $|S|$, we will lower bound $|D \cap \Gamma_X(W)|$ linearly in $|W|$ where W contains S and is such that we can apply the locality property of Lemma 19 to prove that Algorithm 2 cannot flip any set on input $(W, D \cap \Gamma_X(W))$. Applying Corollary 16 for the execution of Algorithm 2 on input $(W, D \cap \Gamma_X(W))$ will provide the desired lower bound. We will call such a W a *witness* for S (see Definition 24).

In the proof of Theorem 13, we will show that if we have witnesses for any $S \subseteq E_{\text{ls}}$ then E_{ls} is local stochastic. This will require to be able to control the number of possible witnesses for a given S and that is why we will need an additional condition $W \in \mathcal{M}(S)$ in the definition of a witness (see Definition 23 for $\mathcal{M}(S)$ and Definition 24 for witness).

Definition 23 ([14])**.** *Let $\mathcal{G} = (V \cup C_X, \mathcal{E})$ be the syndrome adjacency graph and $d := d_B(d_B + 2d_A - 1)$ an upper bound on the degrees in $\mathcal{G}$ (as defined above Lemma 19). For $S \subseteq V$, we define $\mathcal{M}(S) \subseteq \mathcal{P}(V)$ as the set of all subsets $W \subseteq V$ such that $S \subseteq W$ and such that any connected component W' of W in $\mathcal{G}$ satisfies $W' \cap S \neq \varnothing$.*

In this definition, a set $W \in \mathcal{M}(S)$ is any superset of S such that S intersects every connected component of W. Lemma 2 of Ref. [14] provides an upper bound on the number of sets $W \in \mathcal{M}(S)$ of a given size:

$$|\{W \in \mathcal{M}(S) : |W| = t\}| \leq \frac{(ed)^t}{ed^{|S|}}. \tag{12}$$

Definition 24. *Let $D \subseteq C_X$ be a syndrome error and c be a constant. For $S \subseteq V$, we say that $W \subseteq V$ is a* c-witness for (S, D) *if $W \in \mathcal{M}(S)$ and $|W| \leq c|D \cap \Gamma_X(W)|$.*

The proofs are organized as follow: first we show that we can easily find a witness for any $S \subseteq E \oplus \hat{E}$ under the assumptions that $|E \oplus \hat{E}| \leq \gamma_0\sqrt{n}$ and that $E \oplus \hat{E}$ is reduced (Lemma 25), second we use Lemma 25 in order to construct witnesses for $S \subseteq E_{\text{ls}}$ under the assumption that $|\text{MaxConn}_{\alpha_0}(E)| \leq \gamma_0\sqrt{n}$ (Lemma 26), third we show using percolation arguments that $|\text{MaxConn}_{\alpha_0}(E)| \leq \gamma_0\sqrt{n}$ holds with probability $1 - e^{-\Omega(\sqrt{n})}$ when E and D are local stochastic (Lemma 27) and finally we conclude the proof of Theorem 13 by showing that if there exist witnesses for all $S \subseteq E_{\text{ls}}$ then E_{ls} is local stochastic.

Lemma 25. *We use the notations of Section 3.1 with $\delta < 1/16$ and $\beta \in (0; \beta_1)$.*
If the remaining error $E \oplus \hat{E}$ is reduced and $|E \oplus \hat{E}| \leq \gamma_0\sqrt{n}$ then for all $S \subseteq E \oplus \hat{E}$, there is a c_0-witness W for (S, D) with the additional constraint $W \subseteq E \oplus \hat{E}$.

Proof. We set $E_{\text{R}} = E \oplus \hat{E}$ and define W to be all the connected components of E_{R} in $\mathcal{G}$ that contain at least one element of S. It is clear that $W \in \mathcal{M}(S)$ and that $W \subseteq E \oplus \hat{E}$ hold, and it remains to prove $|W| \leq c_0|D \cap \Gamma_X(W)|$.
By locality (Lemma 19 applied with $K = W$), no flip is done by Algorithm 2 on the input $E_{\text{R}} \cap W, D \cap \Gamma_X(W)$. Moreover, the remaining error $(E \oplus \hat{E}) \cap W$ is reduced as a subset of the reduced error $E \oplus \hat{E}$ (Lemma 17) and satisfies $|(E \oplus \hat{E}) \cap W| \leq \gamma_0\sqrt{n}$. Hence Corollary 16 states that:

$$|W| = |(E \oplus \hat{E}) \cap W| \leq c_0|D \cap \Gamma_X(W) \cap \sigma_X((E \oplus \hat{E}) \cap W)| \leq c_0|D \cap \Gamma_X(W)|.$$

□

Lemma 26. *We use the notations of Section 3.1 with $\delta < 1/16$ and $\beta \in (0; \beta_1)$.*
If $|\text{MaxConn}_{\alpha_0}(E \cup D)| \leq \gamma_0\sqrt{n}$ then there is a reduced error E_{ls} equivalent to the remaining error $E \oplus \hat{E}$ such that for all $S \subseteq E_{ls}$ there is a c_0-witness for (S, D).

Proof. We define E_{ls} as one of the minimal weight errors whose syndrome is the same than the syndrome of the error $E \oplus \hat{E}$. We will show that E_{ls} and $E \oplus \hat{E}$ are equivalent.
Let K be a connected component of $U \cup E_{\text{ls}}$. By locality (Lemma 19), there is a valid execution of Algorithm 2 on input $(E \cap K, D \cap \Gamma_X(K))$ whose output is $\hat{E} \cap K$ and whose support is $U \cap K$. Using Proposition 22 we know that $(U \cap K) \cup (D \cap \Gamma_X(K))$ is a $2\alpha_0$-subset of $(E \cap K) \cup (D \cap \Gamma_X(K))$. By Lemma 21 and the fact that $|E_{\text{ls}} \cap K| \leq |(E \oplus \hat{E}) \cap K| \leq |U \cap K|$, we know that $K \cup (D \cap \Gamma_X(K))$ is an α_0-subset of $(E \cap K) \cup (D \cap \Gamma_X(K))$. Hence $K \cup (D \cap \Gamma_X(K))$ is an α_0-subset of $E \cup D$. From the hypothesis $|\text{MaxConn}_{\alpha_0}(E \cup D)| \leq \gamma_0\sqrt{n}$, we conclude that $|K| \leq \gamma_0\sqrt{n}$. In particular, $E \oplus \hat{E}$ and E_{ls} have the same syndrome and the weight of $E \oplus \hat{E} \oplus E_{\text{ls}}$ is smaller than the minimal distance thus $E_{\text{ls}} \cap K$ is equivalent to $(E \oplus \hat{E}) \cap K$. Since this is true for all K then E_{ls} is equivalent to $E \oplus \hat{E}$.
Let $S \subseteq E_{\text{ls}}$, keeping the same notations we will prove that for each K, there is a witness W_K for $S_K := S \cap K$ with the additional constraint $W_K \subseteq E_{\text{ls}} \cap K$:
Let $E' = E_{\text{ls}} \oplus \hat{E}$ and let's prove that if we run Algorithm 2 on input $(E' \cap K, D \cap \Gamma_X(K))$ then the output $\hat{E}'$ satisfies the hypothesis of Lemma 25. The errors E and E' have same syndrome thus the behavior of Algorithm 2 is the same on input (E, D) and on input (E', D). In other words there is a valid execution of Algorithm 2 on input (E', D) whose output is $\hat{E}' = \hat{E}$ and whose support is $E' \cup F_0 \cup \ldots \cup F_{f-1}$. By locality (Lemma 19), there is also a valid execution of Algorithm 2 on input $(E' \cap K, D \cap \Gamma_X(K))$ whose output is $\hat{E} \cap K$ and whose support is $(E' \cup F_0 \cup \ldots \cup F_{f-1}) \cap K$. The remaining error of the last execution is $(E' \oplus \hat{E}) \cap K = E_{\text{ls}} \cap K$ which is reduced and is of weight smaller than $\gamma_0\sqrt{n}$. By Lemma 25, there is a c_0-witness W_K for S_K with $W_K \subseteq E_{\text{ls}} \cap K$. Finally, $W = \biguplus_K W_K$ is a c_0-witness for S.

□

The last ingredient before proving Theorem 13 is provided by Lemma 27 below: the condition $\text{MaxConn}_{\alpha_0}(E \cup D) \leq \gamma_0\sqrt{n}$ needed in Lemma 26 is verified with high probability for local stochastic errors.

Lemma 27 (α-percolation, [8])**.** *Let $\mathcal{G} = (V \cup C_X, \mathcal{E})$ be the syndrome adjacency graph and $d := d_B(d_B + 2d_A - 1)$ an upper bound on the degrees in $\mathcal{G}$ (as defined above Lemma 19). Then for any $\alpha \in (0, 1]$, there exists a threshold $p_{th} = p_{th}(\alpha, d) > 0$ such that for any $t \in \mathbb{N}^*$ and $p < p_{th}$:*

$$\mathbb{P}[\mathit{MaxConn}_\alpha(E) \geq t] \leq C|\mathcal{V}| \left(\frac{p}{p_{th}}\right)^{\alpha t},$$

where $C = C(\alpha, p, p_{th})$ is a constant and $E \subseteq \mathcal{V}$ is a random set chosen accordingly to a local stochastic noise of parameter p.

Proof of Theorem 13. We define the event succ to be $\text{MaxConn}_{\alpha_0}(E \cup D) \leq \gamma_0\sqrt{n}$. When we choose the error E and D according to a local stochastic noise, Lemma 27 ensures that succ holds with probability at least $1 - e^{-\Omega(\sqrt{n})}$. We define E_{ls} as in Lemma 26. We know that provided $\text{MaxConn}_{\alpha_0}(E \cup D) \leq \gamma_0\sqrt{n}$, there is a c_0-witness W for (S, D) for any $S \subseteq E_{\text{ls}}$ and to conclude we will upper bound the probability that such a witness exists.
First of all, for a fixed $W \subseteq V$, W is a witness for (S, D) implies that there is a set $T \subseteq \Gamma_X(W)$ such that $|W| \leq c_0|T|$ and $T \subseteq D$. Thus the probability that W is a witness for (S, D) is upper bounded by:

$$\mathbb{P}[W \text{ is a witness for } (S, D)] \leq \sum_{\substack{T \subseteq \Gamma_X(W): \\ |W| \leq c_0|T|}} \mathbb{P}[T \subseteq D] \leq \sum_{\substack{T \subseteq \Gamma_X(W): \\ |W| \leq c_0|T|}} p_{\text{synd}}^{|T|}.$$

Since the cardinality of $\Gamma_X(W)$ is upper bounded by $d_B|W|$, we have:

$$\mathbb{P}[W \text{ is a witness for } (S, D)] \leq \sum_{|T| \geq |W|/c_0} \binom{d_B|W|}{|T|} p_{\text{synd}}^{|T|}.$$

For k, n integers, we have the well-known upper bound on the binomial coefficient:

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

where $h(x) = -x\log_2(x) - (1-x)\log_2(1-x)$ is the binary entropy function. Since $n \mapsto nh(k/n)$ is decreasing, with $k = |T|$, $n = d_B|W|$ and under the condition $|T| \geq |W|/c_0$, we have:

$$\binom{d_B|W|}{|T|} \leq \left(\frac{p_1^{c_0}}{p_{\text{synd}}}\right)^{|T|} \text{ where } p_1 := 2^{d_B h(1/(d_B c_0))} p_{\text{synd}}^{1/c_0}.$$

Under the condition that p_{synd} is sufficiently small to guarantee $p_1 < 1$, we get:

$$\mathbb{P}[W \text{ is a witness for } (S, D)] \leq \sum_{|T| \geq |W|/c_0} p_1^{c_0|T|} \leq \frac{p_1^{|W|}}{1 - p_1^{c_0}}.$$

By conditioning on the event $\text{MaxConn}_{\alpha_0}(E \cup D) \leq \gamma_0\sqrt{n}$, we can now bound the probability of $S \subseteq E_{\text{ls}}$ as follows:

$$\begin{aligned}\mathbb{P}[S \subseteq E_{\text{ls}} | \text{MaxConn}_{\alpha_0}(E \cup D) \leq \gamma_0\sqrt{n}] &= \frac{\mathbb{P}[S \subseteq E_{\text{ls}}, \text{MaxConn}_{\alpha_0}(E \cup D) \leq \gamma_0\sqrt{n}]}{\mathbb{P}[\text{MaxConn}_{\alpha_0}(E \cup D) \leq \gamma_0\sqrt{n}]} \\ &= 2 \cdot \mathbb{P}[S \subseteq E_{\text{ls}}, \text{MaxConn}_{\alpha_0}(E \cup D) \leq \gamma_0\sqrt{n}],\end{aligned}$$

where we used $\mathbb{P}[\text{MaxConn}_{\alpha_0}(E \cup D) \leq \gamma_0\sqrt{n}] \geq 1 - e^{-\Omega(\sqrt{n})} \geq \frac{1}{2}$ (Lemma 27). Now using Lemma 26 and under the condition that p_{synd} is sufficiently small to guarantee $p_1 ed < 1$ we get

$$\begin{aligned}
\mathbb{P}[S \subseteq E_{\text{ls}} | \text{MaxConn}_{\alpha_0}(E \cup D) \leq \gamma_0\sqrt{n}] &\leq 2 \cdot \mathbb{P}[\exists W : W \text{ is a witness for } (S, D)] \\
&\leq 2 \sum_{\substack{W \in \mathcal{M}(S) \\ |W| \geq |S|}} \mathbb{P}[W \text{ is a witness for } (S, D)] \\
&\leq 2 \sum_{\substack{W \in \mathcal{M}(S) \\ |W| \geq |S|}} \frac{p_1^{|W|}}{1 - p_1^{c_0}} \\
&\leq 2 \sum_{|W| \geq |S|} \frac{(p_1 ed)^{|W|}}{d^{|S|} e \left(1 - p_1^{c_0}\right)} \\
&\leq 2 \frac{(p_1 e)^{|S|}}{e \left(1 - p_1 ed\right) \left(1 - p_1^{c_0}\right)}.
\end{aligned}$$

If p_{synd} is sufficiently small, then we can make this $\leq \left(K p_{\text{synd}}^{1/c_0}\right)^{|S|}$.

□

4 Parallel version for Algorithm 2

At each step in Algorithm 2, we flip a subset of $F_g \subseteq \Gamma_X(g)$ for some X-type generator g. We can parallelize this procedure by flipping at each step the subsets of F_g for a large number of X-type generators. However, we have to pay attention to the fact that the $\sigma_X(F_g)$ could intersect. In order to avoid that, we introduce a coloring of the X-type generators: if g_1 and g_2 have the same color then for any $F_1 \subseteq \Gamma_X(g_1)$ and $F_2 \subseteq \Gamma_X(g_2)$: $\sigma_X(F_1) \cap \sigma_X(F_2) = \varnothing$.

Lemma 28. *The set of X-type generators can be partitioned in less than χ sets of generators (χ from Section 3.1) such that each subset can be decoded in parallel.*
In other words, there is a partition $C_Z = \biguplus_{k=1}^{\chi} C_Z^k$ of the X-type generators such that for any $v_1, v_2 \in V$ and for any $c_1, c_2 \in C_Z$, if v_1 and v_2 are in the support of the same Z-type generator, v_1 is in the support of c_1 and v_2 is in the support of c_2 then c_1 and c_2 cannot be in same subset C_Z^k.

Proof. We explicitly construct a partition of the X-type generators consisting of a constant number of sets, each containing generators with non-intersecting syndromes. From the two factor graphs G_X and G_Z, we construct the graph G_0 whose vertex set is C_Z the set of X-type generators and whose incidence relation is given for $c_1 \in C_Z$ by:

$$\Gamma(c_1) = \{c_2 \in C_Z : \Gamma_X(\Gamma_Z(c_1)) \cap \Gamma_X(\Gamma_Z(c_2)) \neq \varnothing\}.$$

In words, two X-type generators c_1 and c_2 are adjacent in G_0 when we can find a qubit v_1 in the support of c_1 and a qubit v_2 in the support of c_2 which are in the support of the same Z-type generator. The sets $C_Z^1, \ldots, C_Z^\chi$ are defined as a coloring of the graph G_0. The chromatic number is upper bounded by the degree of the graph (plus 1) thus it is upper bonded by χ from Section 3.1. □

This leads to Algorithm 3 below which is a parallelized version of Algorithm 2. It is important to note a difference with Algorithm 2 though: instead of running until no flips reduce the syndrome

weight, Algorithm 3 runs for a fixed number of steps and may have some (final) steps that do not reduce the syndrome weight.

Algorithm 3 : Parallel small-set-flip decoding algorithm for quantum expander codes of parameter $\beta \in (0; 1]$ and with $f \in \mathbb{N}$ steps.

INPUT: $\sigma \subseteq C_X$ a syndrome where $\sigma = \sigma_X(E) \oplus D$ with $E \subseteq V$ an error on qubits and $D \subseteq C_X$ an error on the syndrome

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

$\hat{E}_0 = 0$; $\sigma_0 = \sigma$
for $i \in [\![0; f-1]\!]$ **do**
  $k = i \mod \chi$ // Current color
  in parallel for $g \in C_Z^k$ **do**
    if $\mathcal{F}_g := \{F \in \mathcal{F} : F \subseteq \Gamma_Z(g) \ \& \ \Delta(\sigma_i, F) \geq \beta|\sigma_X(F)|\} \neq \varnothing$ **then**
      $F_g =$ any element of $\mathcal{F}_g$
    else
      $F_g = \varnothing$
    end if
  end parallel for
  $F_i = \bigoplus_{g \in C_Z^k} F_g$
  $\hat{E}_{i+1} = \hat{E}_i \oplus F_i$
  $\sigma_{i+1} = \sigma_i \oplus \sigma_X(F_i)$ // $\sigma_{i+1} = \sigma_X(E \oplus \hat{E}_{i+1}) \oplus D$
end for

Remark 29. *The sequence of flips performed by Algorithm 3 could have been performed by Algorithm 2. As a consequence, if we define $U = E \cup F_0 \cup \ldots \cup F_{f-1}$ the execution support then Proposition 22 implies that the set $U \cup D$ is $2\alpha_0$-subset of $E \cup D$.*

If we fix the number of steps of Algorithm 3 to be $f = f_0(|\sigma|)$ where f_0 is defined in Section 3.1, then it is clear that the complexity is logarithmic in the size of the input. The tricky part is to show that the output is local stochastic and the proof will follow the same scheme than the proof for Algorithm 2: Lemma 33 is the counterpart of Lemma 25, Lemma 34 is the counterpart of Lemma 26 and Theorem 30 is the counterpart of Theorem 13.

Theorem 30. *We use the notations of Section 3.1 with $\delta < 1/16$, $\beta \in (0; \beta_1)$ and $c > c_2 + 1$ running Algorithm 3 with $f = f_0(|\sigma|)$ steps instead of Algorithm 2.*
There exist constants $p_0 > 0, p_1 > 0$ such that the following holds. Suppose the pair (E, D) satisfies a local stochastic noise model with parameter $(p_{\text{phys}}, p_{\text{synd}})$ where $p_{\text{phys}} < p_0$ and $p_{\text{synd}} < p_1$. Then there exists an event succ *that has probability $1 - e^{-\Omega(\sqrt{n})}$ and a random variable E_{ls} that is equivalent to $E \oplus \hat{E}$ such that conditioned on* succ*, E_{ls} has a local stochastic distribution with parameter $p_{\text{ls}} = K p_{\text{synd}}^{1/c_3}$ where K is a constant independent of p_{synd}.*

About the parameter c, Algorithm 3 depends on it because of the number of steps $f = f_0(|\sigma|)$. When c increases, η decreases and the number of steps $f_0(|\sigma|)$ decreases but the drawback is that c_3 increases and thus the local stochastic parameter p_{ls} associated to the error E_{ls} gets worse. On the other hand, when c is chosen close to $c_2 + 1$, p_{ls} gets better but the number of steps grows.

The proof of Theorem 30 proceeds in the same way as the proof of Theorem 13. Indeed, it is sufficient to find a reduced error E_{ls} equivalent to $E \oplus \hat{E}$ and such that for all $S \subseteq E_{\text{ls}}$ there is a c_3-witness for (S, D) in the sense of Definition 24. As for the sequential algorithm, we start by establishing the existence of a witness under the assumptions that the remaining error $E \oplus \hat{E}$

is reduced with weight smaller than $\gamma_0\sqrt{n}$. This will be done in Lemma 33 below, but we will first establish two useful properties of Algorithm 3: it is local (Lemma 31) and at any step i of the algorithm, we either have $|\sigma_i| < c|D|$ or the syndrome weight will decrease rapidly in the following steps (Lemma 32).

Lemma 31 (Locality for Algorithm 3)**.** *We use the notations of Section 3.1 running Algorithm 3 with f steps instead of Algorithm 2.*
For any $K \subseteq U$ with $\Gamma(K) \cap \Gamma(U \setminus K) = \varnothing$ in $\mathcal{G}$ (for the same $\mathcal{G}$ as Lemma 19), there is a valid execution of Algorithm 3 with f steps on input $(E \cap K, D \cap \Gamma(K))$ which flips $F_0 \cap K, \ldots, F_{f-1} \cap K$.

Lemma 32. *We use the notations of Section 3.1 with $\delta < 1/16$, $\beta \in (0; \beta_1)$ and $c > c_2 + 1$ running Algorithm 3 with f steps instead of Algorithm 2.*
Suppose $|U| \leq \gamma_0\sqrt{n}$ and that for some $i \in [\![0; f - \chi]\!]$ we have $|\sigma_i| \geq c|D|$ then $|\sigma_{i+\chi}| \leq \eta|\sigma_i|$ where η is defined in Section 3.1.

Proof. We define the error $E_i := E \oplus F_0 \oplus \ldots \oplus F_{i-1}$. We have:

$$|\sigma_X(E_i)| \geq |\sigma_i| - |D| \geq (c-1)|D| > c_2|D \cap \sigma_X(E_i)|. \tag{13}$$

Since $|E_i| \leq |U| \leq \gamma_0\sqrt{n}$, we can apply Proposition 14 to E_i: there exists a color $k \in [\![0; \chi - 1]\!]$, there exists $G \subseteq \mathcal{F}_k := \mathcal{F} \cap \{\Gamma_X(g) : g \in C_Z^k\}$ such that $\Delta(\sigma_i, F) \geq \beta|\sigma_X(F)|$ for all $F \in G$ and:

$$\sum_{F \in G} |\sigma_X(F)| \geq \frac{c_1}{\chi} \left[|\sigma_X(E_i)| - c_2|D \cap \sigma_X(E_i)| \right]. \tag{14}$$

We set j as the integer in $[\![i+1; i+\chi]\!]$ such that $j-1$ is a step of the algorithm where we flip the color k. In other words, $k = (j-1) \mod \chi$.
We define $G_1 \subseteq \mathcal{F}$ the set of all flips done during the steps $i, \ldots, j-2$ and $G_2 \subseteq \mathcal{F}_k$ the set of all flips done at step $j-1$. By the coloring property, $\sigma_X(F) \cap \sigma_X(F') = \varnothing$ for all $F, F' \in G$ thus for all $F_1 \in G_1$, each check-node of $\sigma_X(F_1)$ is in the syndrome of at most one element of G. In other words, for a given syndrome σ and for $F_1 \in G_1$, the number of $F \in G$ such that $\Delta(\sigma, F) \neq \Delta(\sigma \oplus \sigma_X(F_1), F)$ is at most $|\sigma_X(F_1)|$. As a consequence:

$$|G_2| \geq |G| - \sum_{F_1 \in G_1} |\sigma_X(F_1)|. \tag{15}$$

Moreover:

$$|\sigma_i| - |\sigma_{j-1}| \geq \sum_{F_1 \in G_1} \beta|\sigma_X(F_1)|, \tag{16}$$

and

$$|\sigma_{j-1}| - |\sigma_j| \geq \sum_{F_2 \in G_2} \beta|\sigma_X(F_2)| \geq \beta|G_2|. \tag{17}$$

Combining eqs. (15) to (17) we get:

$$\sum_{F \in G} |\sigma_X(F)| \leq \sum_{F \in G} d_A d_B = d_A d_B |G| \leq d_A d_B \left(|G_2| + \sum_{F_1 \in G_1} |\sigma_X(F_1)| \right) \leq \frac{d_A d_B}{\beta} (|\sigma_i| - |\sigma_j|).$$

By eq. (14):

$$|\sigma_i| - |\sigma_j| \geq \frac{\beta}{d_A d_B} \sum_{F \in G} |\sigma_X(F)| \geq \frac{\beta c_1}{d_A d_B \chi} \left[|\sigma_X(E_i)| - c_2 |D \cap \sigma_X(E_i)| \right].$$

Using $|\sigma_X(E_i)| \geq (c-1)|D| \geq (c-1)|D \cap \sigma_X(E_i)|$ from eq. (13) and the hypothesis $|\sigma_i| \geq c|D|$:

$$|\sigma_i| - |\sigma_j| \geq \frac{c(1-\eta)}{c-1} |\sigma_X(E_i)| \geq \frac{c(1-\eta)}{c-1} (|\sigma_i| - |D|) \geq (1-\eta)|\sigma_i|.$$

Thus $|\sigma_{i+\chi}| \leq |\sigma_j| \leq \eta |\sigma_i|$. □

Lemma 33. *We use the notations of Section 3.1 with $\delta < 1/16$, $\beta \in (0; \beta_1)$ and $c > c_2 + 1$ running Algorithm 3 with $f = f_0(|\sigma|)$ steps instead of Algorithm 2.*
Suppose $|U \cup D| \leq \gamma_0 \sqrt{n}$ and the remaining error $E \oplus \hat{E}$ is reduced then for all $S \subseteq E \oplus \hat{E}$, there is a c_3-witness W for (S, D) with $W \subseteq U$.

Proof. For any $S \subseteq E_{\mathrm{R}} := E \oplus \hat{E}$ and all $i \in [\![0; f]\!]$ we define:

- $U_i := (E \oplus F_0 \oplus \ldots \oplus F_{i-1}) \cup F_i \cup \ldots \cup F_{f-1} = E_{\mathrm{R}} \cup F_i \cup \ldots \cup F_{f-1}$,
- W_i, the union of the connected components of U_i which contains at least one element of S,
- $E_i := (E \oplus F_0 \oplus \ldots \oplus F_{i-1}) \cap W_i$,
- $X_i := \Gamma(W_i)$,
- $s_i := \sigma_i \cap X_i = \sigma_X(E_i) \oplus (D \cap X_i)$.

Note that $U_{i+1} \subseteq U_i, W_{i+1} \subseteq W_i$ and $X_{i+1} \subseteq X_i$.
We are going to show that there is at least one $i \in [\![0; f]\!]$ such that W_i is a witness for S. Suppose that for some $i \in [\![0; f - \chi]\!]$, we have $|s_i| \geq c|D \cap X_i|$. By locality (Lemma 31 applied with $f - i$ steps, for $K = W_i$ and errors $E \oplus F_0 \oplus \ldots \oplus F_{i-1}, D$), there is a valid execution of Algorithm 3 with $f - i$ steps on input $(E_i, D \cap X_i)$ which flips $F_i \cap W_i, \ldots, F_{f-1} \cap W_i$. Note that the remaining error is $E_i \oplus (F_i \cap W_i) \oplus \ldots \oplus (F_{f-1} \cap W_i) = E_{\mathrm{R}} \cap W_i$. Hence for $k \in [\![i; f-1]\!]$:

$$\Delta(\sigma_k \cap X_i, F_k \cap W_i) \geq \frac{\beta d_A}{2} |F_k \cap W_i|. \tag{18}$$

And by Lemma 32:

$$|\sigma_{i+\chi} \cap X_i| \leq \eta |\sigma_i \cap X_i|.$$

Hence using $X_{i+\chi} \subseteq X_i$:

$$|s_{i+\chi}| = |\sigma_{i+\chi} \cap X_{i+\chi}| \leq |\sigma_{i+\chi} \cap X_i| \leq \eta |\sigma_i \cap X_i| = \eta |s_i|.$$

The last inequality is true as soon as $|s_i| \geq c|D \cap X_i|$ but we run Algorithm 3 with $f = f_0(|\sigma|) \geq f_0(|s_0|)$ steps hence there is at least one $i \in [\![0; f]\!]$ such that:

$$|s_i| \leq c|D \cap X_i|. \tag{19}$$

$E_{\mathrm{R}} \cap W_i$ is reduced as a subset of the reduced error E_{R} (Lemma 17) thus we apply Lemma 15 to get:

$$|\sigma_X(E_{\mathrm{R}} \cap W_i)| \geq \frac{\beta_0 d_A}{2} |E_{\mathrm{R}} \cap W_i| \tag{20}$$

As a conclusion, eqs. (18) to (20) imply that W_i is a c_3-witness for S:

$$\begin{aligned}
|W_i| &= |W_i \cap U_i| \\
&\leq |E_{\mathrm{R}} \cap W_i| + \sum_{k=i}^{f-1} |F_k \cap W_i| \\
&\leq \frac{2}{\beta_0 d_A} |\sigma_X(E_{\mathrm{R}} \cap W_i)| + \frac{2}{\beta d_A} \sum_{k=i}^{f-1} \Delta(\sigma_k \cap X_i, F_k \cap W_i) \qquad \text{by eqs. (18) and (20)} \\
&\leq \frac{2}{\beta d_A} \left(|\sigma_X(E_{\mathrm{R}} \cap W_i)| + |\sigma_i \cap X_i| - |\sigma_f \cap X_i|\right) \\
&= \frac{2}{\beta d_A} \left(|\sigma_X(E_{\mathrm{R}} \cap W_i)| + |s_i| - |\sigma_X(E_{\mathrm{R}} \cap W_i) \oplus (D \cap X_i)|\right) \\
&\leq \frac{2}{\beta d_A} \left(|D \cap X_i| + |s_i|\right) \\
&\leq c_3 |D \cap X_i| \qquad \text{by eq. (19)}
\end{aligned}$$

□

Lemma 34. *We use the notations of Section 3.1 with $\delta < 1/16$, $\beta \in (0; \beta_1)$ and $c > c_2 + 1$ running Algorithm 3 with $f = f_0(|\sigma|)$ steps instead of Algorithm 2.*
If $|\mathit{MaxConn}_{\alpha_0}(E \cup D)| \leq \gamma_0 \sqrt{n}$ then there is a reduced error E_{ls} equivalent to the remaining error $E \oplus \hat{E}$ such that for all $S \subseteq E_{ls}$ there is a c_3-witness for (S, D).

Proof. Using the locality property (Lemma 31), the proof is identical to the proof of Lemma 26. □

With Lemma 34, the proof of Theorem 30 is the same as the proof Theorem 13.

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 [14]. We would also like to thank the anonymous reviewers for their useful comments. AG and AL acknowledge support from the ANR through the QuantERA project QCDA.

References

[1] Dorit Aharonov and Michael Ben-Or. Fault-tolerant quantum computation with constant error rate. *SIAM Journal on Computing*, 38(4):1207–1282, 2008.

[2] Héctor Bombín. Single-shot fault-tolerant quantum error correction. *Physical Review X*, 5(3):031043, 2015.

[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] Earl T Campbell. A theory of single-shot error correction for adversarial noise. *arXiv preprint arXiv:1805.09271*, 2018.

[6] Nicolas Delfosse. Tradeoffs for reliable quantum information storage in surface codes and color codes. In *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on*, pages 917–921. IEEE, 2013.

[7] 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.

[8] Omar Fawzi, Antoine Grospellier, and Anthony Leverrier. Efficient decoding of random errors for quantum expander codes. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 521–534. ACM, 2018.

[9] Michael H Freedman, David A Meyer, and Feng Luo. Z2-systolic freedom and quantum codes. *Mathematics of quantum computation, Chapman & Hall/CRC*, pages 287–320, 2002.

[10] Péter Gács and Anna Gál. Lower bounds for the complexity of reliable boolean circuits with noisy gates. *IEEE Transactions on Information Theory*, 40(2):579–583, 1994.

[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*. PhD thesis, California Institute of Technology, 1997.

[13] Daniel Gottesman. An introduction to quantum error correction and fault-tolerant quantum computation. *arXiv preprint arXiv:0904.2557*, 2009.

[14] Daniel Gottesman. Fault-tolerant quantum computation with constant overhead. *Quantum Information & Computation*, 14(15-16):1338–1372, 2014.

[15] Larry Guth and Alexander Lubotzky. Quantum error correcting codes and 4-dimensional arithmetic hyperbolic manifolds. *Journal of Mathematical Physics*, 55(8):082202, 2014.

[16] Matthew B Hastings. Decoding in hyperbolic spaces: quantum LDPC codes with linear rate and efficient error correction. *Quantum Information & Computation*, 14(13-14):1187–1202, 2014.

[17] 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.

[18] 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.

[19] Vivien Londe and Anthony Leverrier. Golden codes: quantum LDPC codes built from regular tessellations of hyperbolic 4-manifolds. *arXiv preprint arXiv:1712.08578*, 2017.

[20] Rüdiger Reischuk and Bernd Schmeltz. Reliable computation with noisy circuits and decision trees-a general n log n lower bound. In *Foundations of Computer Science, 1991. Proceedings., 32nd Annual Symposium on*, pages 602–611. IEEE, 1991.

[21] Tom Richardson and Ruediger Urbanke. *Modern coding theory*. Cambridge University Press, 2008.

[22] Michael Sipser and Daniel A Spielman. Expander codes. *IEEE Transactions on Information Theory*, 42(6):1710–1722, 1996.

[23] Andrew M Steane. Error correcting codes in quantum theory. *Physical Review Letters*, 77(5):793, 1996.

[24] 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.