2015 IEEE 56th Annual Symposium on Foundations of Computer Science

Quantum Expander Codes

Anthony Leverrier, Jean-Pierre Tillich
Inria, EPI SECRET
B.P. 105, 78153 Le Chesnay Cedex, France
Email:anthony.leverrier@inria.fr, jean-pierre.tillich@inria.fr

Gilles Zémor
Institut de Mathématiques de Bordeaux
UMR 5251, Université de Bordeaux
France
Email: zemor@math.u-bordeaux.fr

Abstract

We present an efficient decoding algorithm for constant rate quantum hypergraph-product LDPC codes which provably corrects adversarial errors of weight proportional to the code minimum distance, or equivalently to the square-root of the blocklength. The algorithm runs in time linear in the number of qubits, which makes its performance the strongest to date for linear-time decoding of quantum codes. The algorithm relies on expanding properties, not of the quantum code's factor graph directly, but of the factor graph of the original classical code it is constructed from.

Keywords

Expander codes; LDPC codes; quantum codes; CSS codes

I. Introduction

A quantum CSS code is a particular instance of a quantum stabilizer code, and can be defined by two classical binary linear codes $\mathcal{C}_X$ and $\mathcal{C}_Z$ in the ambient space $\mathbb{F}_2^n$, with the property that $\mathcal{C}_X^\perp \subset \mathcal{C}_Z$ and $\mathcal{C}_Z^\perp \subset \mathcal{C}_X$. In other words, the classical codes $\mathcal{C}_X$ and $\mathcal{C}_Z$ come together with respective parity-check matrices H_X and H_Z such that the linear space $R_X = \mathcal{C}_X^\perp$ generated by the rows of H_X is orthogonal to the row space $R_Z = \mathcal{C}_Z^\perp$ of H_Z, where orthogonality is with respect to the standard inner product. An *error pattern* is defined as a couple (e_X, e_Z), where e_X and e_Z are both binary vectors. The decoder is given the pair of *syndromes* $\sigma_X = H_X e_X^T$ and $\sigma_Z = H_Z e_Z^T$ and decoding succeeds if it outputs, not necessarily the initial error pattern (e_X, e_Z), but a couple of the form $(e_X + f_X, e_Z + f_Z)$ where $f_X \in R_Z$ and $f_Z \in R_X$. See [14] for the equivalence with the stabilizer formalism and a detailed introduction to quantum coding.

If efficient quantum computing is to be achieved, it will come with a strong error-correcting component, that will involve very fast decoders, probably in not much more than linear time in the blocklength n. The likeliest candidates for this task are quantum LDPC codes: in the CSS case, an LDPC code is simply a code whose above parity-check matrices H_X and H_Z have row and column weights bounded from above by a constant. Among recent developments, the recent paper [15] has shown how fault tolerant quantum computation with constant multiple overhead can be obtained, and quantum LDPC codes are an essential component of the scheme, making them possibly even more appealing.

It is natural to hope that the success of classical LDPC codes, both in terms of performance and of decoding efficiency, can eventually be matched in the quantum setting. This agenda involves two major difficulties, however. The first one is that coming up with intrinsically good constructions of quantum LDPC codes is in itself a challenge. In particular the random constructions that can be so effective in the classical case do not work at all in the quantum case. Indeed if one chooses randomly a sparse parity-check matrix H_X, then, precisely since this gives a good classical code, there are no low-weight codewords in the dual of the row-space of H_X and therefore an appropriate matrix H_Z does not exist. A testament to this difficulty is given in the introduction of [23] by way of a list of proposed constructions of quantum LDPC codes from within the classical coding community that all yield constant minimum distances. Presently, the known constructions of families of quantum LDPC codes that

0272-5428/15 $31.00 © 2015 IEEE
DOI 10.1109/FOCS.2015.55

come with constant rates and minimum distances that grow with the qubit length can be reduced to essentially three constructions. The first consists of quantum codes based on tilings of two-dimensional hyperbolic manifolds (surfaces) that generalise Kitaev's toric code and originate in [11]. The minimum distance of these codes grows as $\log n$, where n is the qubit length. A recent generalisation of this approach to 4-dimensional hyperbolic geometry [16] yields minimum distances that behave as n^{ϵ} where ϵ is larger than some unknown constant and not more than 0.3. Finally, the construction [23] yields codes of constant rate with minimum distances that grow as $n^{1/2}$. These codes are perhaps the closest to classical LDPC codes in spirit, since they are constructed by taking a properly defined product of a classical LDPC code with itself. We note that presently all known constructions of quantum codes, even if they are allowed to have vanishing rate, fail to significantly break the $n^{1/2}$ barrier for the minimum distance and it is an intriguing open question as to whether there exist asymptotically good quantum LDPC codes (i.e. with constant rate and minimum distance linear in n). We also make the side remark that Gottesman [15] requires, for the purpose of fault-tolerant quantum computation, constant rate LDPC codes with good minimum distance properties that should behave well under some sort of adversarial error setting.

The second difficulty in attempting to match the achievements of classical LDPC coding, is to devise efficient decoding algorithms. The vast majority of decoding algorithms developed for classical LDPC codes rely on iterative techniques whose ultimate goal is to take decisions on individual bits. A straightforward transposition to the quantum setting of this strategy would consist in trying to recover the bits of e_X and e_Z by decoding the LDPC codes $\mathcal{C}_X$ and $\mathcal{C}_Z$. The problem with this approach is that the classical LDPC codes $\mathcal{C}_X$ and $\mathcal{C}_Z$ are somewhat non-standard in that they have by construction bounded minimum distance. Indeed, $\mathcal{C}_X$ has minimum distance at most equal to the smallest row weight of H_Z since $\mathcal{C}_Z^{\perp} \subset \mathcal{C}_X$ and by definition $\mathcal{C}_Z^{\perp}$ is generated by the rows of H_Z. Such a decoder for $\mathcal{C}_X$ is doomed to fail because it is fooled by any error that spans only half the weight of a row of H_Z [19]. In other words, such decoding algorithms can not have vanishing error probability.

The way to overcome this problem is really to look for the most likely error modulo the stabilizer group. However this also implies that bit-oriented decoding strategies are mostly pointless, there is here no "correct value" for a single bit of e_X or e_Z which can always be just as well 0 or 1. Decoding quantum LDPC codes requires therefore additional elements to the classical toolkit.

Decoding quantum LDPC codes seems to be easier in one case, namely for the surface codes mentioned above. This is due to the fact that the underlying classical codes $\mathcal{C}_X$ and $\mathcal{C}_Z$ are cycle codes of graphs: full decoding, which is NP-hard in general for linear codes, can be achieved for cycle codes of graphs in polynomial time (with the help of Edmonds' weighted matching algorithm [9]), and this strategy (which does not really qualify as a local technique) yields a decoding scheme for the quantum code that achieves vanishing error-probability for random errors.

Unfortunately, this technique does not extend to other classes of LDPC codes, and in an adversarial setting is limited to correcting at most $\log n$ errors, since the minimum distance of surface codes of constant rate can never surpass a logarithm of the qubit length [6]. Recently, a number of alternative decoders have been proposed such as for example the decoders which utilize renormalization group ideas, proposed by Duclos-Cianci and Poulin [7], [8], by Sarvepalli and Raussendorf [21] and by Bravyi and Haah [4]. In some sense, they can be viewed as multilevel decoding strategies. While the first two decoders [7], [8] can be applied to other codes than just surface codes, they really apply to quantum LDPC codes with a very specific multilevel cell structure such as the toric code, higher dimensional versions of this code, or color codes based on lattices – but the generalisation of this approach to more general versions of LDPC codes is an open problem. Moreover, analysing this kind of decoder seems to be quite challenging due to the correlations that appear during the decoding process and has yet to be achieved even for simple structures such as the toric code. On the other hand, the hard decision renormalization decoder of [4] is much more amenable to analysis but requires in a crucial way that the frontier of the clusters of growing size formed during the decoding process should be small compared to the size of the clusters. This happens for instance when the Tanner graphs associated to H_X and H_Z are very poorly expanding (which is precisely what happens in the lattice structures considered in [4]) but this will not be the case for the quantum codes we will consider here.

Even more recently, an alternative decoding algorithm was proposed [17] for the 4-dimensional hyperbolic codes of Guth and Lubotzky that is devised to work in a probabilistic setting and for which its adversarial performance is unclear. The third class of constant rate quantum codes with growing minimum distance, namely the codes [23], had no known decoding algorithm to go with it until the present paper whose prime objective is to tackle this very problem.

In this paper we devise a decoding algorithm for the product codes [23] that runs in linear time and decodes an arbitrary pattern of errors of any weight up to a constant fraction of the minimum distance, i.e. $cn^{1/2}$ for some constant $c > 0$. Our inspiration is the decoding algorithm of Sipser and Spielman [22] which applies to classical LDPC codes whose Tanner graph is an expander graph. The quantum codes under consideration here are products (in a sense to be given precisely below) of a classical LDPC code $\mathcal{C}$ with itself, and we take the original code $\mathcal{C}$ to be an expander code. The resulting Tanner graphs of the two classical codes $\mathcal{C}_X$ and $\mathcal{C}_Z$ that make up the quantum code are not strictly speaking expander graphs, but they retain enough expanding structure from the original code for a decoding algorithm to work. Arguably, this is the first time that an import from classical LDPC coding theory succeeds in decoding a quantum LDPC code from a non-constant number of errors in an adversarial setting. There are some twists to the original Sipser-Spielman decoding scheme however, since it guesses values of individual bits and we have pointed out that this strategy cannot carry through to the quantum setting. The solution is to work with *generators* rather than qubits: the generators are the row vectors of H_X and H_Z (thus called because they correspond to generators of the stabilizer group of the code). At each iteration, the decoding algorithm looks for a pattern of qubits inside the support of a single generator that will decrease the syndrome weight.

Our results also have some significance in the area of local testability. Locally Testable Codes (LTC) play a fundamental role in complexity theory: they have the property that code membership can be verified by querying only a few bits of a word [13]. More precisely, the number of constraints not satisfied by a word should be proportional to the distance of the word from the code. Given their importance, it is natural to ask whether a quantum version of LTC exists, and to investigate their consequences for the burgeoning field of quantum Hamiltonian complexity, which studies quantum satisfaction problems [12].

Quantum LT codes were recently defined in [2], and these hypothetical objects are mainly characterized by their *soundness* (or *robustness*) $R(\delta)$, i.e. the probability that a word at relative distance δ from the code violates a randomly chosen constraint.

While we do not exhibit such quantum LT codes here, we construct codes which are robust for errors of reasonably low weight, up to a constant fraction of the minimum distance: see Corollary 9 below. Reaching beyond the regime of low weight errors appears to be much harder since it is well-known that the random expander codes at the heart of our construction are not locally testable [3]. Interestingly, for our construction, better expansion translates into greater robustness. This should be seen in contrast to results in Ref. [1], [2], where good expansion (admittedly not of the same graph) appears to hurt the local testability of the quantum codes. We also remark that in the very recent result of [10], quantum codes are constructed by applying [23] to classical LT codes, which leads to an alternative form of robustness where errors with small syndrome weight correspond to highly entangled states.

The remainder of the paper is organized as follows: Section II describes the code construction with its expanding properties and states the main result. Section III describes the basic decoding algorithm. Section IV gives the main points of the analysis of the algorithm, and states the robustness result. Sections V and VI are dedicated to the proofs of the more technical lemmas. Finally Section VII gives some concluding comments.

II. Expansion, code construction and main result

Let $G = (A \cup B, \mathcal{E})$ be a biregular bipartite graph with left (resp. right) degree equal to Δ_A (resp. Δ_B). Let $|A| = n_A$, $|B| = n_B$, and suppose $n_B \leq n_A$, so that $\Delta_A \leq \Delta_B$. We shall write $a \sim_G b$ (or more concisely, $a \sim b$ when G is clear from context) to mean that the vertices a and b are adjacent in the graph G. If S is a subset of vertices, denote by $\Gamma(S)$ the set of all neighbors of vertices of S. Let us say that G is (γ_A, δ_A)-*left-expanding*,

for some constants $\gamma_A, \delta_A > 0$, if for any subset $S \subseteq A$ with $|S| \leq \gamma_A n_A$ we have $|\Gamma(S)| \geq (1-\delta_A)\Delta_A|S|$. Similarly, we shall say that G is (γ_B, δ_B)-*right-expanding* if for any subset $S \subseteq B$ with $|S| \leq \gamma_B n_B$ we have $|\Gamma(S)| \geq (1-\delta_B)\Delta_B|S|$. Finally we shall say that G is $(\gamma_A, \delta_A, \gamma_B, \delta_B)$ left-right expanding (or simply expanding) is it is both (γ_A, δ_A)-left-expanding and (γ_B, δ_B)-right-expanding.

To any bipartite graph G we may associate the $n_B \times n_A$ matrix H, whose rows are indexed by the vertices of B, whose columns are indexed by the vertices of A, and such that $H_{ij} = 1$ if i and j are adjacent in G and $H_{ij} = 0$ otherwise. A binary linear code $\mathcal{C}_G$ can be thus defined as the set of vectors x of $\mathbb{F}_2^{n_A}$ such that $Hx^T = 0$, i.e. $\mathcal{C}_G$ is the code with parity-check matrix H. Conversely, any code $\mathcal{C}$ has a parity-check matrix H and the binary matrix H can in turn be viewed as an incidence relation between its rows and columns, i.e. a bipartite graph G: such a graph is usually called "the" Tanner graph or the factor graph of the code $\mathcal{C}$. A code has several factor graphs, because it has several parity-check matrices, but it is usually clear which one we are talking about. "Left" and "Right" vertices of the factor graph are traditionally associated to bits (or variables) and equations respectively.

Expansion and classical error-correction are connected through the following result of Sipser and Spielman [22].

Theorem 1. *Let $G = (A \cup B, \mathcal{E})$ be a (Δ_A, Δ_B)-biregular (γ_A, δ_A)-left-expanding graph. Letting Δ_A and Δ_B be fixed and allowing n_A to grow, there exists a decoding algorithm for the associated code $\mathcal{C}_G$ that runs in time linear in the code length $n_A = |A|$, and that, under the condition $\delta_A < 1/4$, corrects any pattern of at most $\frac{1}{2}\gamma_A n_A$ errors.*

Our objective is to derive a quantum analogue of Theorem 1. The codes we shall work with are the codes of [23], whose construction we briefly recall. As mentioned in the introduction, a CSS code is defined by two classical codes $\mathcal{C}_X$ and $\mathcal{C}_Z$: in our case, both these classical codes are constructed from a fixed bipartite graph $G = (A \cup B, \mathcal{E})$. Let us describe $\mathcal{C}_X$ and $\mathcal{C}_Z$ through their factor graphs $\mathcal{G}_X$ and $\mathcal{G}_Z$.

The bipartite graph $\mathcal{G}_X$ has left set of vertices $A^2 \cup B^2$, and its right set of vertices is $A \times B$. The bipartite graph $\mathcal{G}_Z$ has the same left vertices but its set of right vertices is $B \times A$. We will find it convenient to denote vertices of $\mathcal{G}_X$ and $\mathcal{G}_Z$ by pairs of letters, omitting parentheses to lighten notation, and to use Greek letters for right vertices of $\mathcal{G}_X$, Latin letters for right vertices of $\mathcal{G}_Z$, and denote elements of A^2 by a Greek letter followed by a Latin letter, and elements of B^2 by a Latin letter followed by a Greek letter. Typical elements of $A^2, B^2, A \times B$, and $B \times A$ will therefore be written respectively, αa, $b\beta$, $\alpha\beta$, and ba. The incidence structure of the two graphs is defined as follows:

- In $\mathcal{G}_X$: the set of neighbors of left vertex $\alpha a \in A^2$ is defined as
$$\Gamma(\alpha a) = \{\alpha\beta \in A \times B, a \sim_G \beta\}.$$
The set of neighbors of left vertex $b\beta \in B^2$ is defined as
$$\Gamma(b\beta) = \{\alpha\beta \in A \times B, \alpha \sim_G b\}.$$
- In $\mathcal{G}_Z$: the set of neighbors of left vertex $\alpha a \in A^2$ is defined as
$$\Gamma(\alpha a) = \{ba \in B \times A, \alpha \sim_G b\}.$$
The set of neighbors of left vertex $b\beta \in B^2$ is defined as
$$\Gamma(b\beta) = \{ba \in B \times A, a \sim_G \beta\}.$$

The corresponding parity-check matrices H_X and H_Z of $\mathcal{C}_X$ and $\mathcal{C}_Z$ may therefore be written in concise form as:

$$H_X = \left(\mathbb{1}_{n_A} \otimes H, H^T \otimes \mathbb{1}_{n_B}\right), \tag{1}$$
$$H_Z = \left(H \otimes \mathbb{1}_{n_A}, \mathbb{1}_{n_B} \otimes H^T\right). \tag{2}$$

It is not difficult to check [23] that the rows of H_X are orthogonal to the rows of H_Z, so that $(\mathcal{C}_X, \mathcal{C}_Z)$ makes up a valid CSS quantum code. We also see that the parity-check matrices H_X and H_Z are low density, with constant row weight $\Delta_A + \Delta_B$. It is proved furthermore in [23] that the parameters of the quantum code $\mathcal{Q}_G = (\mathcal{C}_X, \mathcal{C}_Z)$ are

$$[[n = n_A^2 + n_B^2, k \geq (n_A - n_B)^2, \min(d, d^T)]] \tag{3}$$

where d denotes the *minimum distance* of the classical code $\mathcal{C}_G$, i.e. the minimum number of columns of the $n_B \times n_A$ matrix H that sum to zero, and d^T stands for the associated *transpose minimum distance*, which is the minimum number of *rows* of H that sum to zero. This is with the convention that each of these respective minima is set to equal ∞ if there are no subsets of columns, or rows respectively, that sum to zero. The transpose minimum distance associated to a parity-check matrix of a classical linear code is not a standard parameter because it is usually ∞, but this is not necessarily the case for a matrix associated to a typical bipartite biregular graph, and it cannot be totally overlooked.

Crucially, if one fixes the degrees Δ_A, Δ_B with $\Delta_A < \Delta_B$ and one takes an infinite family of graphs $G = (A \cup B, \mathcal{E})$ with increasing number of vertices n_A, n_B, the rate of the quantum code $\mathcal{Q}_G$ is bounded from below by a non-zero constant and its typical minimum distance scales as the square-root of its length n.

We remark that the construction of [23] is somewhat more general, using two base graphs G_1 and G_2 rather than a single graph G. The above construction corresponds to the case when $G = G_1 = G_2$ and we choose to restrict ourselves to this setting for ease of description and notation.

We can now state our main result:

Theorem 2. *Let $G = (A \cup B, \mathcal{E})$ be a (Δ_A, Δ_B)-biregular $(\gamma_A, \delta_A, \gamma_B, \delta_B)$-left-right-expanding graph. Assume the conditions $\delta_A < 1/6$ and $\delta_B < 1/6$. Letting Δ_A and Δ_B be fixed and allowing n_A, n_B to grow, there exists a decoding algorithm for the associated quantum code $\mathcal{Q}_G$ that runs in time linear in the code length $n = n_A^2 + n_B^2$, and that decodes any quantum error pattern of weight less than*

$$w_0 = \frac{1}{1 + 3\Delta_B} \min\left(\gamma_A n_A, \gamma_B n_B\right). \tag{4}$$

Comments: Theorem 2 implies in particular that the quantum code $\mathcal{Q}_G$ must have a minimum distance proportional to $\min(\gamma_A n_A, \gamma_B n_B)$, and that the decoding algorithm corrects a number of adversarial errors equal to a constant fraction of the minimum distance. We stress that we need both left and right expansion from the graph G, and that for these values of δ_A and δ_B, there are no known constructions that achieve it. However, the graph G with the required expanding properties may be obtained by random choice with classical probabilistic arguments (see Chap. 8 of [20] and [5]): in particular one obtains that left and right expansion with $\delta_A < 1/6$ and $\delta_B < 1/6$ are guaranteed for degrees $\Delta_A \geq 7$ and $\Delta_B \geq 7$.

III. Description of the decoding algorithm

Informal description of the algorithm: As is standard for stabilizer codes, a quantum measurement gives the decoder the syndrome s of the error, and from then on all computations are classical. The decoder then goes over all generators, i.e. the supports of the rows of the matrices H_X and H_Z, and looks exhaustively for an error pattern e_1 inside one of these supports such that the syndrome of e_1, when summed with the original syndrome vector s, yields a vector of weight smaller than that of s. The algorithm then goes on to look for small error patterns, all contained in some generator, that keep decreasing the weight of the syndrome until eventually an error pattern is constructed whose syndrome matches the syndrome of the original error. Decoding succeeds when this pattern is equivalent to the original error.

We now describe the decoding problem precisely. An error pattern is a set of coordinates $E_X \subset \{1, \ldots, n\}$ on which there is an X Pauli error, together with a set of coordinates E_Z on which there is a Z Pauli error. There may be coordinates in $E_X \cap E_Z$ on which an X error and a Z error occur simultaneously, which is equivalent to a Y Pauli error. The error pattern can therefore be given in equivalent form as the couple of binary vectors

$e = (e_X, e_Z)$ where e_X and e_Z have supports E_X and E_Z respectively. The input to the decoder is the *syndrome* $\sigma(e)$ of the error e. It is made up of the classical syndromes of the vectors e_X and e_Z for the matrices H_X and H_Z, i.e. $\sigma(e) = (\sigma_X(e_X), \sigma_Z(e_Z))$, with $\sigma_X(e_X) = H_X e_X^T$ and $\sigma_Z(e_Z) = H_Z e_Z^T$. We will say that *the error pattern* $e = (e_X, e_Z)$ *is correctly decoded* if on input $\sigma(e)$ the decoder outputs $(e_X + f_X, e_Z + f_Z)$, with f_X and f_Z in the row space of H_Z and the row space of H_X respectively. Our purpose is to exhibit a decoder that will always correctly decode every error e of weight smaller than the quantity given in Theorem 2. The *weight* $w(e)$ of the error $e = (e_X, e_Z)$ is the number of coordinates i in which at least one of the two vectors e_X, e_Z is non-zero.

Before describing the decoder, let us fix some additional notation. The rows of the matrix H_Z are called *Z-generators* (for the purpose of this paper's internal notational coherence: note that this terminology is not standard in the stabilizer formalism). Recall that a row of H_Z is indexed by an element ba of $B \times A$. We choose to identify the corresponding generator with its support in $A^2 \cup B^2$, and denote it by g_{ba}, so that we have:

$$g_{ba} := \{\alpha a \ : \ \alpha \sim b\} \cup \{b\beta \ : \ \beta \sim a\}\,. \tag{5}$$

Similarly the X-generators are denoted by $g_{\alpha\beta}$ and we write

$$g_{\alpha\beta} := \{a\beta \ : \ a \sim \beta\} \cup \{\alpha b \ : \ b \sim \alpha\}\,. \tag{6}$$

The decoding algorithm treats X and Z errors independently. This means that the decoder applies two separate decoding procedures, one consisting of outputting $e_X + f_X$ from $\sigma_X(e_X)$, and the other outputting $e_Z + f_Z$ from $\sigma_Z(e_Z)$.

We remark that it is enough to prove Theorem 2 for error patterns of the form $(e_X, 0)$ and of the form $(0, e_Z)$. We will therefore describe the decoding algorithm for errors of type $(e_X, 0)$, the other case being symmetric. Note that the weight of the error pattern $w(e_X, 0)$ is simply the Hamming weight of the vector e_X, and we denote it by $|e_X|$.

The decoding algorithm for e_X works by going through the generators g_{ba} and for each one, by looking at whether flipping any pattern of bits strictly decreases the weight of the syndrome. More precisely: the algorithm takes as input a syndrome vector $s \in \mathbb{F}_2^{A \times B}$. It then looks for a vector $e_1 \in \mathbb{F}_2^{A^2 \cup B^2}$ such that:

- the support of e_1 is contained in some generator g_{ba},
- $s_1 = s + \sigma_X(e_1)$, with $|s_1| < |s|$,
- $\frac{|s| - |s_1|}{|e_1|}$ is maximal, subject to the above two conditions.

If the algorithm cannot find a vector e_1 satisfying the first two conditions, it outputs a decoding failure. After i decoding iterations, the algorithm is in possession of a syndrome vector $s_i \in \mathbb{F}_2^{A \times B}$, together with a vector $e_1 + e_2 + \cdots + e_i \in \mathbb{F}_2^{A^2 \cup B^2}$, such that

- the support of every e_j, $j = 1 \ldots i$, is included in some generator,
- $s_i = s + \sigma_X(e_1 + \cdots + e_i)$ and $|s_i| < |s|$.

The $(i+1)$-th decoding iteration consists in finding a vector $e_{i+1} \in \mathbb{F}_2^{A^2 \cup B^2}$ such that

- the support of e_{i+1} is included in some generator g_{ba},
- $s_{i+1} = s_i + \sigma_X(e_{i+1})$, with $|s_{i+1}| < |s_i|$,
- $\frac{|s_i| - |s_{i+1}|}{|e_{i+1}|}$ is maximal, subject to the above two conditions.

The algorithm proceeds until it reaches some iteration i after which it cannot find any generator g_{ba} which enables it to decrease the weight of s_i. If $|s_i| \neq 0$, then it outputs a decoding failure. Otherwise we have $s_i = 0$, and the algorithm outputs the vector $e_O := e_1 + e_2 + \cdots + e_i$.

We shall prove that if $s = \sigma_X(e_X)$ with $|e_X|$ sufficiently small, then the decoding algorithm never outputs a failure and its output e_O satisfies $e_O + e_X \in \mathcal{C}_Z^\perp$, equivalently $e_O + e_X$ is a sum of Z-generators, meaning the algorithm has correctly decoded the error pattern e_X.

By carefully updating the list of generators to be re-examined after every iteration, we obtain in a classical way an algorithm that runs in time linear in the number n of qubits, in the uniform cost model.

IV. Analysis of the decoding algorithm

We first recall some tools and results from [22].

Consider the graph $G = (A \cup B, \mathcal{E})$ and a subset of vertices S. Let us define the set $\Gamma_u(S)$ of *unique neighbors* of S, that is, the set of vertices v such that v has degree 1 in the graph $S \cup \Gamma(S)$ induced by S. The complement of $\Gamma_u(S)$ in $\Gamma(S)$ will be called the set of *multiple neighbors* of S and denoted $\Gamma_m(S)$. Provided that the expansion in the graph G is large enough, then the graph displays unique-neighbor expansion, in the following sense:

Lemma 3. *[22]. Let $G = (A \cup B, \mathcal{E})$ be a (γ_A, δ_A)-left-expanding graph with $\delta_A < 1/2$. Then, for any subset $S_A \subseteq A$ with $|S_A| \leq \gamma_A n_A$, we have:*

$$|\Gamma_u(S_A)| \geq (1 - 2\delta_A)\Delta_A |S_A|. \tag{7}$$

Proof: Consider the bipartite graph induced by S_A and $\Gamma(S_A)$. The number of edges incident to S_A should coincide with the number of edges incident to $\Gamma(S_A)$, that is $\Delta_A |S_A| = |\Gamma_u(S_A)| + \tilde{\Delta} |\Gamma_m(S_A)|$ where $\tilde{\Delta} \geq 2$ is the average right degree on $\Gamma_m(S_A)$. This implies that $|\Gamma_m(S_A)| \leq \frac{1}{2}(\Delta_A |S_A| - |\Gamma_u(S_A)|$. Moreover, the expansion in G requires that $|\Gamma_u(S_A)| \geq (1 - \delta_A)\Delta_A |S_A| - |\Gamma_m(S_A)|$. Combining both inequalities gives the unique-neighbor expansion. ∎

We also recall from [22]:

Proposition 4. *If G is (γ_A, δ_A)-left-expanding for $\delta_A < 1/2$, then the minimum distance d of the associated classical code $\mathcal{C}_G$ is at least $\gamma_A n_A$.*

From Proposition 4 we immediately obtain:

Corollary 5. *If the graph $G = (A \cup B, \mathcal{E})$ is $(\gamma_A, \delta_A, \gamma_B, \delta_B)$-left-right-expanding with $\delta_A, \delta_B < 1/2$, then the minimum distance of the associated quantum code $\mathcal{Q}_G$ is at least* $\min(\gamma_A n_A, \gamma_B n_B)$.

Proof: Proposition 4 implies that the minimum distance of the associated classical code is at least $\gamma_A n_A$, and, by inverting the roles of A and B, that the transpose minimum distance is at least $\gamma_B n_B$. The result follows from (3). ∎

The crucial part of the original decoding strategy of Sipser and Spielman consists in showing that, for errors of sufficiently small weight, there must exist (at least) one critical variable vertex of A in error that has many unique neighbors. Flipping the bit associated with this vertex decreases the syndrome value and this eventually leads to showing that one can always decode correctly by flipping bits that decrease the syndrome weight, provided the initial error is sufficiently small. In the present setting, we need a corresponding notion of criticality for generators. It will build upon the unique neighbor idea with a number of twists.

Definition 6. *Let $E \subset A^2 \cup B^2$ be a subset of vertices that can be thought of as an error pattern. Let us say that a generator $g_{ba} = \{\alpha a \,:\, \alpha \sim b\} \cup \{b\beta \,:\, \beta \sim a\}$ is* critical *(with respect to E) if it can be partitioned as follows:*

$$g_{ba} = x_a \cup \bar{x}_a \cup \chi_a \cup x_b \cup \bar{x}_b \cup \chi_b \tag{8}$$

where

- *$x_a \cup \bar{x}_a \cup \chi_a$ is a partition of $g_{ba} \cap A^2$ and $x_b \cup \bar{x}_b \cup \chi_b$ is a partition of $g_{ba} \cap B^2$,*
- *$x_a \cup x_b \subset E$, $\bar{x}_a \cap E = \emptyset$ and $\bar{x}_b \cap E = \emptyset$,*
- *every vertex in $\Gamma(x_a) \cap \Gamma(x_b)$ has exactly two neighbors in E,*
- *every vertex in $\Gamma(\bar{x}_a) \cap \Gamma(\bar{x}_b)$ has no neighbor in E,*
- *every vertex in $\Gamma(x_a) \cap \Gamma(\bar{x}_b)$ and every vertex in $\Gamma(\bar{x}_a) \cap \Gamma(x_b)$ has exactly one neighbor in E,*
- *$x_a \cup x_b \neq \emptyset$, $|\chi_a| \leq 2\delta_B \Delta_B, |\chi_b| \leq 2\delta_A \Delta_A$.*

Note that Definition 6 implies in particular that the syndrome vector has value 0 in the coordinates indexed by $\Gamma(x_a) \cap \Gamma(x_b)$ and $\Gamma(\bar{x}_a) \cap \Gamma(\bar{x}_b)$, and has value 1 in the coordinates indexed by $\Gamma(x_a) \cap \Gamma(\bar{x}_b)$ and in $\Gamma(\bar{x}_a) \cap \Gamma(x_b)$. See Fig. 1 for details.

Lemma 7 below asserts that a critical generator with respect to an error pattern always exists, whenever the error pattern E has sufficiently small weight (cardinality), and Lemma 8 claims that its is always possible to modify the pattern of qubits inside a critical generator in a way as to simultaneously decrease the syndrome weight and not increase the error weight. Recall that we are decoding an X-error e_X that we simply denote e from now on.

Lemma 7. *If the weight of the error e satisfies $0 < |e| \leq \min(\gamma_A n_A, \gamma_B n_B)$, then there exists a critical generator g_{ba} with respect to the support E of e.*

The proof of Lemma 7 is deferred to Section V.

If e is a vector of $\mathbb{F}_2^n$, let us call its *reduced weight*, denoted $w_R(e)$, the smallest Hamming weight of an element of the coset $e + \mathcal{C}_Z^{\perp}$, i.e. the set of vectors of the form $e + f$, where f is a sum of Z-generators.

Lemma 8. *If the reduced weight of the error e satisfies $0 < w_R(e) \leq \min(\gamma_A n_A, \gamma_B n_B)$, then there exists a critical generator together with a vector e_1 whose support is included in the generator, such that $|\sigma_X(e)| - |\sigma_X(e + e_1)| \geq |e_1|/3$. In words, flipping the k bits of the support of e_1 decreases the syndrome weight by at least $k/3$. Moreover, $w_R(e + e_1) \leq w_R(e)$.*

The proof of Lemma 8 is given in Section VI.

The decoding algorithm is analysed in two steps. First we show that it never outputs a decoding failure, i.e. always decreases the syndrome weight to zero, and secondly we prove the output error vector e_O is equivalent to the original vector e modulo the space of Z-generators.

Lemma 8 implies the *robustness* of the quantum code.

Corollary 9 (Robustness)**.** *Any error e with reduced weight $w_R(e) < \min(\gamma_A n_A, \gamma_B n_B)$ has a syndrome with weight bounded from below as $|\sigma(e)| \geq \frac{1}{3} w_R(e)$.*

Proof: Without loss of generality, suppose e is a representative of $e + \mathcal{C}_Z^{\perp}$ with minimum weight. As long as the weight of the syndrome is positive, Lemma 8 guarantees the existence of a generator and a vector e_1 whose support is included in the generator such that $|\sigma_X(e)| - |\sigma_X(e + e_1)| \geq |e_1|/3$. Moreover, since $w_R(e + e_1) \leq |e|$, we can apply Lemma 8 again to $e + e_1$ and iterate, say i times, until the syndrome weight reaches 0. After i iterations, we obtain that the syndrome of $e + e_1 + e_2 + \cdots + e_i$ is 0, hence that

$$|\sigma_X(e)| \geq \frac{1}{3}(|e_1| + |e_2| + \cdots |e_i|) \geq \frac{1}{3}|e_1 + \cdots e_i|, \tag{9}$$

and that $w_R(e + e_1 + \cdots + e_i) \leq |e| < \min(\gamma_A n_A, \gamma_B n_B)$. This last fact implies by Corollary 5 that $e + e_1 + \cdots + e_i$ is in $C_Z^{\perp}$, i.e. that e is equal to $e_1 + \cdots + e_i$ modulo a sum of Z-generators. Inequality (9) proves therefore that $|\sigma_X(e)| \geq \frac{1}{3} w_R(e)$. ∎

Unfortunately, the decoding algorithm is not guaranteed to follow the good decoding path exhibited by Lemma 8. Indeed, the decoding algorithm simply tries to optimize the weight of the syndrome, but the error weight might increase in the process. Nevertheless, we have:

Lemma 10. *If the Hamming weight of the initial error e is less than w_0, then the decoding algorithm never outputs a decoding failure and always correctly decodes e.*

Proof: The decoding algorithm chooses a sequence of vectors $e_1, e_2, \ldots$, such that

$$|\sigma_X(e)|, |\sigma_X(e + e_1)|, \ldots, |\sigma_X(e + e_1 + \cdots + e_i)|, \ldots$$

is a decreasing sequence. Set $\varepsilon_0 = e, \varepsilon_1 = e + e_1, \ldots, \varepsilon_i = \varepsilon_{i-1} + e_i$. Now whenever

$$|\varepsilon_i| < \min(\gamma_A n_A, \gamma_B n_B), \tag{10}$$

Lemma 8 ensures that

$$\frac{|\sigma_X(\varepsilon_i)| - |\sigma_X(\varepsilon_{i+1})|}{|e_{i+1}|} \geq \frac{1}{3}. \tag{11}$$

By hypothesis (10) holds for $i = 0$. Suppose it holds for $0, 1, \ldots, i$, we then have

$$\begin{aligned} |e_1| &\leq 3(|\sigma_X(\varepsilon_0)| - |\sigma_X(\varepsilon_1)|) \\ \ldots &\leq \ldots \\ |e_{i+1}| &\leq 3(|\sigma_X(\varepsilon_i)| - |\sigma_X(\varepsilon_{i+1})|) \end{aligned}$$

By summing all these inequalities we obtain $|e_1| + \cdots + |e_{i+1}| \leq 3\left(|\sigma_X(\varepsilon_0)| - |\sigma_X(\varepsilon_{i+1})|\right) \leq 3|\sigma_X(e)|$. Writing $|\varepsilon_{i+1}| \leq |e| + |e_1| + \cdots + |e_{i+1}|$ we get $|\varepsilon_{i+1}| \leq |e| + 3|\sigma_X(e)| \leq |e| + 3\Delta_B|e|$ since the syndrome $\sigma_X(e)$ has weight at most $\Delta_B|e|$. From the hypothesis on the Hamming weight of e we get that (10) holds for $i+1$, and inductively until the eventual output of the algorithm $e_O = e_1 + \cdots + e_j$, such that $\varepsilon_j = e + e_O$ has zero syndrome. Condition (10) translates to $|e + e_O| < \min(\gamma_A n_A, \gamma_B n_B)$ which ensures by Corollary 5 that $e + e_O \in \mathcal{C}_Z^\perp$, which means exactly that e_O is a valid representative of the initial error e. ■

V. Proof of Lemma 7

First, we need the following extra notation. Let S be a subset of A in the graph $G = (A \cup B, \mathcal{E})$: For a vertex $a \in S$, we denote as follows, respectively, the set of unique neighbors of a in the subgraph of G induced by $S \cup \Gamma(S)$, and the set of multiple neighbors of a in the same induced graph:

$$\Gamma_u^S(a) := \Gamma(a) \cap \Gamma_u(S), \qquad \Gamma_m^S(a) := \Gamma(a) \cap \Gamma_m(S). \tag{12}$$

Denote now by $E \neq \emptyset$ the support of the X error e to be corrected and define $E_A := E \cap A^2$ (resp. $E_B := E \cap B^2$) the set of error vertices in A^2 (resp. in B^2).

Suppose first $E_B = \emptyset$. Consider the projection E_A^2 of E_A on the second coordinate:

$$E_A^2 := \{a \in A \ : \ A \times \{a\} \cap E_A \neq \emptyset\}. \tag{13}$$

Clearly $|E_A^2| \leq |E_A| \leq w(e) \leq \gamma_A n_A$. Lemma 3 then implies that $|\Gamma_u(E_A^2)| \geq (1 - 2\delta_A)\Delta_A|E_A^2|$. Therefore, using the notation (12), there exists $a \in E_A^2$ such that $|\Gamma_u^{E_A^2}(a)| \geq (1 - 2\delta_A)\Delta_A$. Pick some $\alpha a \in E_A$ and $b \sim \alpha$. Then the generator g_{ba} can be partitioned into four components $g_{ba} = x_a \cup \bar{x}_a \cup \bar{x}_b \cup \chi_b$ which satisfy the requirements of the lemma. Specifically, we define $x_a = g_{ba} \cap E_A$, $\bar{x}_a$ as the complement of x_a in $g_{ba} \cap A^2$,

$$\chi_b = \{b\beta \in g_{ba} \cap B^2 \ : \ \beta \in \Gamma_m^{E_A^2}(a)\}$$

and $\bar{x}_b$ as the complement of χ_b in $g_{ba} \cap B^2$. In words, x_a and $\bar{x}_a$ partition the A^2 part of the generator g_{ba} into error and error-free components, and $\bar{x}_b$ and χ_b partition the B^2 part into, respectively, the set whose second coordinate is a unique neighbor and the set whose second coordinate is a multiple neighbor with respect to E_A^2.

The proof is identical with the roles of A and B interchanged if $E_A = \emptyset$.

Let us turn our attention to the remaining case where both $E_A \neq \emptyset$ and $E_B \neq \emptyset$. We define E_A^2 as above and choose a coordinate $a \in E_A^2$ such that $\left|\Gamma_u^{E_A^2}(a)\right| \geq (1 - 2\delta_A)\Delta_A$. From now on, we use the shorthand $\Gamma_u(a)$ for this set, leaving E_A^2 implicit. Next, we define:

$$E_{B,a} := \{b\beta \in E_B,\ \beta \in \Gamma_u(a)\}.$$

If $E_{B,a} = \emptyset$, we proceed as in the case where $E_B = \emptyset$. Otherwise, by assumption on the weight of the error, namely $|e| \leq \gamma_B n_B$, Lemma 3 applies again, this time to the set $E_{B,a}^1$ corresponding to the projection of $E_{B,a}$ on its first coordinate,

$$E_{B,a}^1 := \{b \ : \ \exists \beta \in \Gamma_u(a), b\beta \in E_B\}. \tag{14}$$

This implies that there exists $b \in E^1_{B,a}$ such that $\left|\Gamma_u^{E^1_{B,a}}(b)\right| \geq (1-2\delta_B)\Delta_B$. Using the shorthand $\Gamma_u(b) := \Gamma_u^{E^1_{B,a}}(b)$, we define:

$$x_a := \{\alpha a \in E_A \ : \ \alpha \in \Gamma_u(b)\} \tag{15}$$

$$\bar{x}_a := \{\alpha a \notin E_A \ : \ \alpha \in \Gamma_u(b)\} \tag{16}$$

$$x_b := \{b\beta \in E_B \ : \ \beta \in \Gamma_u(a)\} \tag{17}$$

$$\bar{x}_b := \{b\beta \notin E_B \ : \ \beta \in \Gamma_u(a)\}. \tag{18}$$

The construction ensures that every element $\alpha\beta$ that belongs both to the neighborhood of $x_a \cup \bar{x}_a$ and of $x_b \cup \bar{x}_b$ is such that α is a unique neighbor of b *and* β is simultaneously a unique neighbor of a, unique neighborhood being understood with respect to E. Moreover $x_b \neq \emptyset$ by construction of the set $E^1_{B,a} \neq \emptyset$.

VI. Proof of Lemma 8

We consider the generator g_{ba} promised by Lemma 7. Without loss of generality, we may suppose that the error vector e is in reduced form, i.e. it is the vector of lowest Hamming weight in the coset $e + \mathcal{C}_Z^\perp$. The consequence is that the number of qubit vertices in error $|E \cap g_{ba}|$ within the generator g_{ba} is not more than $(\Delta_A + \Delta_B)/2$, otherwise replacing e by $e + e'$, with e' the vector having g_{ba} for support, would yield a vector with strictly smaller Hamming weight within the coset $e + \mathcal{C}_Z^\perp$. In particular we have:

$$|x_a \cup x_b| \leq \frac{1}{2}(\Delta_A + \Delta_B). \tag{19}$$

Let us introduce the following reduced variables:

$$x := \frac{|x_a|}{\Delta_B}, \qquad z := \frac{|\chi_a|}{\Delta_B}, \qquad \bar{x} := 1 - x - z, \tag{20}$$

$$y := \frac{|x_b|}{\Delta_A}, \qquad t := \frac{|\chi_b|}{\Delta_A}, \qquad \bar{y} := 1 - y - t. \tag{21}$$

By assumption, $x + y > 0$.

The expansion condition $\delta_A, \delta_B < \frac{1}{6}$ implies that $z, t < 1/3$ by the last condition of Definition 6. However, $z\Delta_B$ and $t\Delta_A$ are integers, this implies $3z\Delta_B \leq \Delta_B - 1$ and $3t\Delta_A \leq \Delta_A - 1$, which in turn give:

$$0 \leq z \leq \frac{1}{3} - \frac{1}{3\Delta_B}, \quad 0 \leq t \leq \frac{1}{3} - \frac{1}{3\Delta_A}. \tag{22}$$

We are looking for a vector e_1, whose support is included in g_{ba}, such that

i. the syndrome of $e + e_1$ has strictly smaller weight than the syndrome of e, and the difference $|\sigma_X(e)| - |\sigma_X(e + e_1)|$ is at least $|e_1|/3$,
ii. the reduced weight $w_R(e + e_1)$ is at most equal to the Hamming weight $|e|$ of e.

We will consider four cases:

1) if $x + y \leq 2/3$, then the vector e_1 is chosen to have support $x_a \cup x_b$,
2) if $x \leq \bar{x}$ and $y \leq \bar{y}$, then the vector e_1 is chosen to have support $x_a \cup x_b$,
3) if the above two hypotheses do not hold and if $x > \bar{x}$ and $y > \bar{y}$, then the vector e_1 is chosen to have support either $\bar{x}_a \cup \bar{x}_b$, or its complement in g_{ba},
4) in the remaining cases, we show that either $x_a \cup x_b$ or $\bar{x}_a \cup \bar{x}_b$, or $x_a \cup x_b \cup \chi_a \cup \chi_b$ is an adequate choice for the support of e_1.

Let us introduce the partition of $\Gamma(g_{ba})$ induced by the partition (8):

$$\Gamma(g_{ba}) = S_a \cup S_b \cup S_{\bar{a}} \cup S_{\bar{b}} \cup S_{ab} \cup S_{a\bar{b}} \cup S_{\bar{a}b} \cup S_{\bar{a}\bar{b}} \cup \bar{S}$$

with

$$\begin{array}{llllllll} S_a & = \Gamma(x_a) \cap \Gamma(\chi_b) & S_b & = \Gamma(x_b) \cap \Gamma(\chi_a) & S_{\bar{a}} & = \Gamma(\bar{x}_a) \cap \Gamma(\chi_b) & S_{\bar{b}} & = \Gamma(\bar{x}_b) \cap \Gamma(\chi_a) \\ S_{ab} & = \Gamma(x_a) \cap \Gamma(x_b) & S_{a\bar{b}} & = \Gamma(x_a) \cap \Gamma(\bar{x}_b) & S_{\bar{a}b} & = \Gamma(\bar{x}_a) \cap \Gamma(x_b) & S_{\bar{a}\bar{b}} & = \Gamma(\bar{x}_a) \cap \Gamma(\bar{x}_b) \\ \bar{S} & = \Gamma(\chi_a) \cap \Gamma(\chi_b) & & & & & & \end{array}$$

as represented on Figure 1.

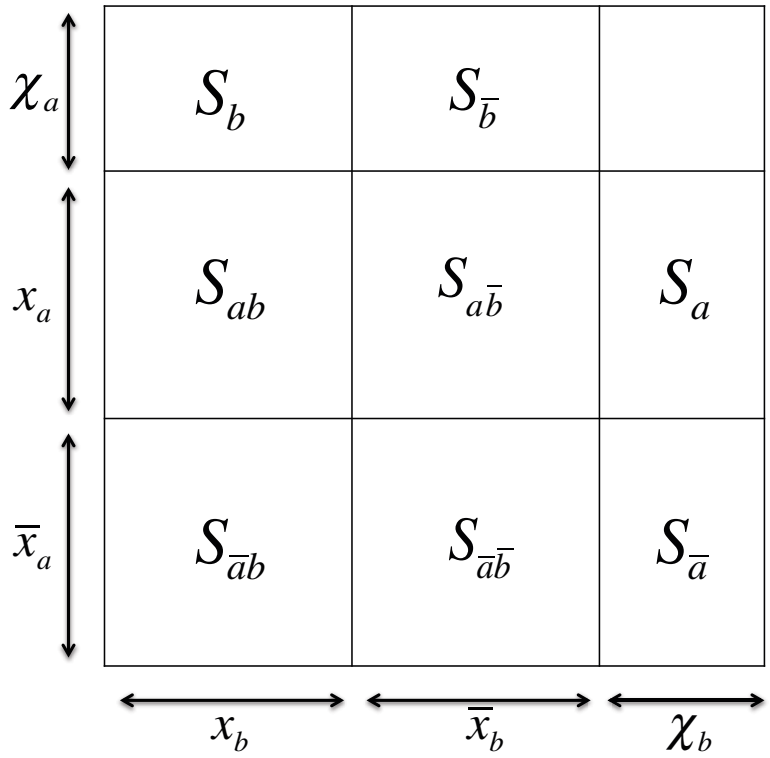

Figure 1: A critical generator g_{ba}. The generator g_{ba} is partitioned into 6 sets: x_a and x_b contain the errors we want to correct, $\bar{x}_a$ and $\bar{x}_b$ do not contain any error, and χ_a and χ_b have a small relative size. When flipping the qubits corresponding to x_a and x_b (or the qubits in $\bar{x}_a$ and $\bar{x}_b$), the syndrome bits in $S_{a\bar{b}}$ and $S_{\bar{a}b}$ are flipped from 1 to 0 while the syndrome bits in the sets S_{ab} and $S_{\bar{a}\bar{b}}$ remain unchanged; the only regions where the syndrome weight can increase are $S_a \cup S_b$ and $S_{\bar{a}} \cup S_{\bar{b}}$.

Let us denote by ∂ the decrease of the syndrome weight when we flip $x_a \cup x_b$ (i.e. choose e_1 to have support $x_a \cup x_b$). Similarly we denote by $\bar{\partial}$ the decrease of the syndrome weight when we flip either $\bar{x}_a \cup \bar{x}_b$ or $x_a \cup x_b \cup \chi_a \cup \chi_b$.

These quantities satisfy:

$$\partial/(\Delta_A \Delta_B) \geq x\bar{y} + \bar{x}y - xt - yz \tag{23}$$

$$\bar{\partial}/(\Delta_A \Delta_B) \geq x\bar{y} + \bar{x}y - \bar{x}t - \bar{y}z. \tag{24}$$

This is because when we flip the bit values on the set $x_a \cup x_b$, then the value of the syndrome is changed from 1 to 0 on the support $S_{a\bar{b}}$ and $S_{\bar{a}b}$, remains at 0 on the supports S_{ab} and $S_{\bar{a}\bar{b}}$, may possibly change from 0 to 1 on the supports $S_a \cup S_b$, and remains unchanged in the other regions. Similarly, when we flip the bit values on the set $\bar{x}_a \cup \bar{x}_b$, the syndrome is flipped from 1 to 0 on the support $S_{a\bar{b}}$ and $S_{\bar{a}b}$ and possibly from 0 to 1 on $S_{\bar{a}} \cup S_{\bar{b}}$ while the other regions remain unchanged.

We now address the four cases:

1) Suppose $x + y \leq 2/3$. Then (23) can be rewritten as:

$$\begin{aligned}\frac{\partial}{\Delta_A \Delta_B} &\geq x\bar{y} + \bar{x}y - xt - yz \\ &\geq x(1-y-t) + y(1-x-z) - xt - yz \\ &\geq x + y - 2xy - 2(tx + yz) \\ &\geq \left\{\frac{1}{3}(x+y) - 2xy\right\} + x\left(\frac{2}{3} - 2t\right) + y\left(\frac{2}{3} - 2z\right) \\ &\geq \left\{\frac{1}{3}(x+y) - 2xy\right\} + \frac{2}{3}\left\{\frac{x}{\Delta_A} + \frac{y}{\Delta_B}\right\}\end{aligned}$$

by applying Eq.(22) to z and t. The first term is nonnegative for $x + y \leq 2/3$, so that we obtain:

$$\partial \geq \frac{2}{3}|x_a \cup x_b| > \frac{1}{3}|x_a \cup x_b|.$$

Furthermore, when e_1 has support $x_a \cup x_b$, the support of e_1 is included in the support of e, hence $|e + e_1| < |e|$ and $w_R(e + e_1) < |e|$.

2) If $x \leq \bar{x}$ and $y \leq \bar{y}$, then $y = 1 - t - \bar{y} \leq \bar{y}$ gives $\bar{y} \geq \frac{1}{2} - \frac{t}{2}$, hence, applying (22),

$$\bar{y} - t \geq \frac{1}{2\Delta_A}.$$

Similarly we have $\bar{x} - z \geq \frac{1}{2\Delta_B}$, and (23) gives

$$\begin{aligned}\partial &\geq \Delta_A \Delta_B \left[x(\bar{y} - t) + y(\bar{x} - z)\right] \\ &\geq \frac{1}{2}\left[x\Delta_B + y\Delta_A\right] \\ &\geq \frac{1}{2}\left|x_a \cup x_b\right|.\end{aligned}$$

The set $x_a \cup x_b$ is again an adequate choice for the support of e_1, and $w_R(e + e_1) < |e|$ as before.

3) If $x > \bar{x}$ and $y > \bar{y}$, then by an argument symmetrical to the preceding case, exchanging x with $\bar{x}$ and y with $\bar{y}$, we get:

$$\bar{\partial} \geq \frac{1}{2}\left|\bar{x}_a \cup \bar{x}_b\right|.$$

We remark that we must have $|\bar{x}_a \cup \bar{x}_b| > 0$, otherwise $z, t \leq 1/3$ imply that $|x_a \cup x_b| > (\Delta_A + \Delta_B)/2$, which contradicts our assumption (19). Therefore, choosing e_1 to have support $\bar{x}_a \cup \bar{x}_b$ gives a strict decrease in the syndrome weight that is at least $|e_1|/3$. Now in this case, the *Hamming weight* $|e + e_1|$ is larger than $|e|$. However we need to consider its *reduced weight*. A vector equivalent modulo $C_Z^{\perp}$ to $e + e_1$ is $e + e_1'$ with the support of e_1' being $x_a \cup x_b \cup \chi_a \cup \chi_b$. The Hamming weight of $e + e_1'$ is

$$|e + e_1'| \leq |e| - (x + y) + (z + t).$$

But since $x + y > 2/3$ and $z, t, < 1/3$, we have $|e + e_1'| < |e|$, so that $w_R(e + e_1) < |e|$.

4) Finally, in the remaining case we consider $\partial + \bar{\partial}$ to show that one of the two quantities, ∂ or $\bar{\partial}$ is sufficiently large. Adding (23) and (24) yields

$$\frac{1}{\Delta_A \Delta_B}(\partial + \bar{\partial}) \geq 2x\bar{y} + 2\bar{x}y - t(1-z) - z(1-t).$$

We now observe that the negation of the condition

$$(x \leq \bar{x} \text{ and } y \leq \bar{y}) \quad \text{or} \quad (x > \bar{x} \text{ and } y > \bar{y})$$

implies $2x\bar{y} + 2\bar{x}y \geq (x+\bar{x})(y+\bar{y}) = (1-z)(1-t)$, from which we get:

$$\frac{1}{\Delta_A \Delta_B}(\partial + \bar{\partial}) \geq (1-z)(1-t) - t(1-z) - z(1-t).$$

The formal equality

$$(1-z)(1-t) - z(1-t) - t(1-z) = 3\left(\frac{2}{3} - z\right)\left(\frac{2}{3} - t\right) - \frac{1}{3}$$

yields therefore, when combined with (22),

$$\frac{1}{\Delta_A \Delta_B}(\partial + \bar{\partial}) \geq \frac{1}{3}\left(1 + \frac{1}{\Delta_A}\right)\left(1 + \frac{1}{\Delta_B}\right) - \frac{1}{3}$$

Hence, after rearranging,

$$\max\left\{\partial, \bar{\partial}\right\} \geq \frac{1}{2}(\partial + \bar{\partial}) \geq \frac{1}{3}\frac{\Delta_A + \Delta_B}{2}.$$

- if $\max\left\{\partial, \bar{\partial}\right\} = \partial$, then we choose e_1 to have support $x_a \cup x_b$, and we get $\partial \geq \frac{1}{3}|e_1|$ from condition (19). We have $w_R(e+e_1) \leq |e|$ as in cases 1. and 2.
- if $\max\left\{\partial, \bar{\partial}\right\} = \bar{\partial}$, then
  - either $|\bar{x}_a \cup \bar{x}_b| \leq \frac{\Delta_A + \Delta_B}{2}$, in which case we set e_1 to have support $\bar{x}_a \cup \bar{x}_b$ and have $\bar{\partial} \geq \frac{1}{3}|e_1|$ and $w_R(e+e_1) \leq |e|$ by the same argument as in case 3,
  - or $|\bar{x}_a \cup \bar{x}_b| > \frac{\Delta_A + \Delta_B}{2}$, in which case we set e_1 to have support $x_a \cup x_b \cup \chi_a \cup \chi_b$ so as to again have $\bar{\partial} \geq \frac{1}{3}|e_1|$. The Hamming weight $|e+e_1|$ is at most $|e| - (x+y) + (z+t)$, which is less than $|e|$ as in case 3, so that again $w_R(e+e_1) < |e|$.

VII. Concluding comments and questions

We have exhibited a linear-time decoding algorithm that corrects up to $\Omega(n^{1/2})$ *adversarial* quantum errors over n qubits. While this is the largest such asymptotic quantity to date, one would hope to break this barrier and eventually achieve correction of $\Omega(n)$ errors. If one were to do this with quantum LDPC codes, this would imply obtaining the elusive proof of existence of low-density codes with a minimum distance scaling linearly in the number of qubits.

Kovalev and Pryadko have shown [18] that the codes of [23] have the potential to correct a number of *random* depolarizing errors that scales linearly in n, with a vanishing probability of decoding error. This is without decoding complexity limitations however, and a natural question is whether the ideas of the present paper can extend to decoding $\Omega(n)$ random errors in linear or quasi-linear time.

We have worked to achieve the smallest possible value of δ_A, δ_B in Theorem 2, i.e. the smallest possible expansion coefficient for the base graph G. Can the bound $\delta_A, \delta_B < 1/6$ be decreased further ? A somewhat related question is whether the left-right expanding base graph G can be obtained constructively, rather than by random choice ?

Acknowledgment

Financial support for this research was provided by the "Investments for the future" Programme IdEx Bordeaux – CPU (ANR-10-IDEX- 03-02).

REFERENCES

[1] D. Aharonov and L. Eldar, "Commuting local Hamiltonians on expanders, locally testable quantum codes, and the qPCP conjecture," *arXiv preprint* arXiv:1301.3407 2013.

[2] D. Aharonov and L. Eldar, "Quantum locally testable codes," *arXiv preprint* arXiv:1310.5664 2013.

[3] E. Ben-Sasson, P. Harsha, and S. Raskhodnikova, "Some 3CNF properties are hard to test," *SIAM Journal on Computing*, 35(1):1–21, 2005.

[4] S. Bravyi and J. Haah, "Analytic and numerical demonstration of quantum self-correction in the 3D cubic code," *Phys. Rev. Lett.*, 111:200501, 2013.

[5] D. Burshtein and G. Miller, "Expander graph arguments for message-passing algorithms," *IEEE Transactions on Information Theory*, 47(2):782–790, 2001.

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

[7] G. Duclos-Cianci and D. Poulin, "Fast decoders for topological quantum codes," *Phys. Rev. Lett.*, 104(050504), 2010.

[8] G. Duclos-Cianci and D. Poulin, "A renormalization group decoding algorithm for topological quantum codes," In *2010 IEEE Information Theory Workshop - ITW 2010 Dublin*, pp. 1–5, 2010.

[9] Jack Edmonds, "Paths, trees, and flowers," *Canadian Journal of mathematics*, 17(3):449–467, 1965.

[10] L. Eldar, "Quantum systems with approximation-robust entanglement," *arXiv preprint* arXiv:1503.02269 2015.

[11] M. H. Freedman, D. A. Meyer, and F. Luo, "Z2-systolic freedom and quantum codes," *Mathematics of quantum computation, Chapman & Hall/CRC*, pp. 287–320, 2002.

[12] S. Gharibian, Y. Huang, Z. Landau, and S. W. Shin, "Quantum Hamiltonian complexity," *arXiv preprint* arXiv:1401.3916 2014.

[13] O. Goldreich, "Short locally testable codes and proofs: A survey in two parts," In *Property testing*, pp. 65–104. Springer, 2010.

[14] D. Gottesman, *Stabilizer codes and quantum error correction*. PhD thesis, California Institute of Technology, Pasadena, CA, 1997.

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

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

[17] M. 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.

[18] A. A. Kovalev and L. P. Pryadko, "Fault tolerance of quantum low-density parity check codes with sublinear distance scaling," *Phys. Rev. A*, 87(2):020304, 2013.

[19] D. Poulin and Y. Chung, "On the iterative decoding of sparse quantum codes," *Quantum Information and Computation*, 8:987, 2008.

[20] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.

[21] P. Sarvepalli and R. Raussendorf, "Efficient decoding of topological color codes," *Phys. Rev. A*, 85:022317, Feb 2012.

[22] M. Sipser and D. A. Spielman, "Expander Codes," *IEEE Transactions on Information Theory*, 42(6):1710–1722, 1996.

[23] J-P. Tillich and G. 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. Short version in Proceedings of the IEEE Symposium on Information Theory, ISIT 2009, Seoul, pp.799-804.