Some New Constructions and Bounds for Locally Generated Quantum Codes

Some New Constructions and Bounds for Locally Generated Quantum Codes

A dissertation presented

by

Kevin Farrell Thompson

to

The School of Engineering and Applied Sciences

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in the subject of

Applied Mathematics

Harvard University

Cambridge, Massachusetts

April 2018

Advisor

Peter W. Shor

Author

Kevin Farrell Thompson

Some New Constructions and Bounds for Locally Generated Quantum Codes

Abstract

The existence of quantum locally generated codes is a long standing open problem in quantum information theory. In this thesis, we consider a bound concerning this conjecture as well as a few constructions with codes that are 'barely non-local'. We establish a complementary result to [30], and show quantum codes which are "strongly" not embeddable into finite dimensional lattices must also have poor distance. Along the way we derive some results concerning "pseudorandom" classical codes.

Given that quantum codes seem to be difficult to construct, it seems useful to examine "bad" quantum codes for applications in information and communication. Indeed, most of the work in quantum error correction by researchers today is in this direction. We construct a "nearly local" quantum erasure code which can achieve the capacity of the quantum erasure channel. This code has very poor (adversarial) distance, but still manages to correct random erasure errors with high probability. The codes use random Erdos-Renyi graphs to construct quantum states which are nearly local, but also highly entangled across fixed cuts with high probability. We derive some new results concerning classical codes with log-sparse parity check matrices which may be of independent interest.

Inspired by this construction, we are able to construct new approximate unitary 2-designs or "scramblers". The study of scrambling is the study of the mixing properties of different distributions of random unitaries. There is an inherent duality between the study of scrambling and the study of error correction: A good quantum code will make a good scrambler and vice versa. We study the scrambling properties of our random Erdos-Renyi graph state encoding circuits. We are able to show that these circuits, when supplemented by some local "Expander Graph" quantum circuits, form approximate unitary 2-designs. This construction, strictly speaking, does not achieve more efficient parameters than existing approximate 2-designs, but might have implementation advantages over other designs and points to a conjecture which could yield approximate unitary designs with time independent qubit-to-qubit coupling. This would be an extremely interesting construction in the context of experimental randomized benchmarking.

Contents

Chapter 1

Introduction

1.1 Quantum Computers need Help

Quantum computers have been the subject of intense research for nearly thirty years now. These are devices that exploit the strange properties of quantum matter to perform computation. The first papers on quantum computation had intuitive ideas about how one might expect much more power out of a computer that can utilize quantum effects [53, 68], but no concrete benefits over classical computing for interesting problems. Factoring was the first example of a practical problem with an efficient quantum algorithm [151], but no known efficient classical algorithm. Since then, researchers have discovered applications in solving difficult Chemistry problems [11, 99], machine learning [20], and many useful schemes not directly related to computing. Quantum cryptography [16] and quantum money [162], for instance, achieve security of information transfer or of currency by exploiting particular properties of quantum systems. These proposals often come with much stronger security guarantees than their classical counterparts.

While we have known for some time that quantum computers will be capable of great things, researchers worried even in the beginning of the field about the effects of noise on quantum systems. Skeptics noted that quantum systems are naturally very sensitive to noise [127, 159], and that noisy computations may lose their quantum advantage. For these reasons, quantum fault-tolerance [56, 134, 150] and error correction [36, 149, 153] were born. These two sub-fields were founded to study ways in which a user can mitigate the effects of noise and error on a quantum system. Researchers noted that, unlike classical information, quantum information was fundamentally "un-cloneable" [163]. Hence, redundancy could not be used directly to protect the integrity of quantum states. Instead, a single copy of some

piece of information was spread over a larger quantum system in such a way that *local* disturbances would not effect the information. The information being stored in the code is "de-localized" over the quantum state.

1.2 qLDPC

After quantum error correction (QEC) was discovered, the next hallmark achievement of the sub-field was the discovery of the *stabilizer formalism* [35, 75]. This was a set of ideas that allowed one to understand these de-localized states in terms of standard sets of operators that acted on them, and provided a framework which generalized known quantum codes. In close analogy with the classical case, the stabilizer formalism can be seen as the "checks" of a (linear) quantum code. The quantum code itself can be defined from the "checks" as well as it can be defined by its generators (in analogy with vector space over finite fields). Researchers were able to reproduce many results from classical coding theory in quantum contexts [10, 103, 148]. However, constructions of locally generated codes did not easily analogize. The problem is that while classical Low-Density Parity-Check (LDPC) codes correspond to structure-less locally generated subspaces, quantum codes have additional structural requirements. A quantum stabilizer code on n qubits corresponds to an (additive) subgroup of $\mathbb{F}_4^n$ that is self-dual with respect to a special inner product [35]. Producing local quantum codes is difficult precisely because it can be hard to find codes that satisfy the "self-dual" property.

The first (asymptotically large) example of a locally generated family of quantum codes was the the Toric code [102]. This code provided a way to satisfy the "self-dual" condition

by relating it to some geometric property of the Torus. Since then, relating the stabilizer conditions to geometric properties has been a fruitful area for the production of interesting quantum codes [24,69]. Unfortunately, these codes and many others [6,28,37,70,73,102,116] were found to have distance scaling with the square root of the block length. In the QEC community this problem is colloquially referred to as the "$\sqrt{n}$ barrier", since even up to today we have no codes with distance greater than $O(\sqrt{n})$ (up to logarithmic factors). This was at least partially explained through Bravyi and Terhal's no-go result for quantum codes [30]. They discovered that quantum codes that can be embedded into a $O(1)$ dimensional lattice *must* have sublinear distance. In fact, codes which can be embedded into 2 dimensional lattices (Toric code, some Color Codes) must have distance scaling with the square root of the block length, so many of these constructions [24, 102] are "distance optimal" given their topology.

Given the apparent difficulty in constructing quantum LDPC codes, research in quantum coding theory has taken one of two directions. Either one can seek codes which are not embeddable into finite dimensional lattices (*expanding codes*), or one can ask for ways to use the geometrically local "bad" codes in quantum computing and information tasks. The latter set of ideas is heavily studied by the community, since geometrically local codes are likely to be more useful for experimentalists in the near term. One possible avenue here is the study of the effects of random errors on quantum codes. While all known local quantum codes have poor distance in the adversarial setting, it is possible (and in fact true) that local quantum codes can recover from $\Omega(n)$ many random errors [52, 109, 155]. Alternatively, there are entirely different paradigms of quantum computing with bad codes. "Magic State" quantum computing [29, 125] is a scheme for universal quantum computing using codes with

a nearly noise free implementation of simple maps (the Clifford Group), along with some consumable resource states. The two components together have the potential to realize universal fault-tolerant quantum computing.

Despite these very successful efforts, there has been some work on circumventing the no-go results by considering codes which are "unembeddable". There are several reasons for this. Linear distance locally generated codes lead to potentially very powerful fault tolerant schemes [77], as well as provide interesting physical models for "macroscopic", highly entangled physical systems constrained by local checks. Further, qLDPC is related to some important problems in quantum computer science [59], with classical analogs that have been resolved [55]. Work in this area has encompassed studies and constructions of quantum codes with "expanding topology". A randomly generated homology [28] provides such an example. This construction manages to achieve linear distance, but has stabilizer weight $\sqrt{n}$ rather than $O(1)$. Another such example is the Tillich-Zemor code [158]. They provide a special code product which allows one to construct a quantum code out of any pair of classical codes. The product of two unembeddable codes is itself unembeddable, hence the codes constructed from their scheme are likely "expanding".

1.3 Quantum Codes as Scramblers

While quantum coding theory was discovered in the context of fault tolerance, it has found other applications in describing interesting physical systems [80, 131], and in quantum pseudorandomness [32, 161], or "scrambling". Quantum pseudorandomness seeks to provide distributions which resemble Haar random unitaries in some well defined way, and it has appli-

cations for many important problems inside quantum information [87,88,104,118,156,160]. A Haar random unitary, U acting on any n qubit state has the property that most subsets of the output are maximally mixed [126], hence U forms an encoding circuit for a quantum code. Quantum encoding circuits provide examples of good scramblers, as well as point to conditions under which good scrambling might occur. The study of local scramblers is especially important because it could help to explain the seemingly non-local scrambling properties of black holes [88].

1.4 This Thesis

In this thesis we solve three problems, one in each of the subfields we have described. The first is a bound on codes *not* covered by the Bravyi-Terhal results. We formalize a natural condition corresponding to "unembeddability", and analyze codes that satisfy it. The second problem is an example of "bad codes doing good things". We provide a randomized construction which achieves the capacity of the quantum erasure channel, while at the same time having very poor distance (logarithmic with the block length). The codes we sample are 'nearly-local', since they have stabilizer weight scaling poly-logarithimically with the block length. For the third problem, we build on the second problem to construct a "nearly local" scrambler. Since good codes make good scramblers, given some code with interesting parameters it makes sense to examine its scrambling properties. We construct a 'nearly-local' scrambler which could have benefits over other known scramblers [41,47,83,121], depending on the requirements and restrictions the user has for the scrambler.

1.4.1 Problem 1

Since researchers in QEC are currently examining expanding codes to look for qLDPC codes, it makes sense to look for bounds on quantum codes which are expanding. We formulate a condition very similar to the 'Cheeger constant' for graphs and demonstrate that codes satisfying our condition have small distance. This is a very counter-intuitive observation, since classical codes that are expanding appear to be very robust [152]. Nonetheless, we are able to relate this condition to an intermediate property we refer to as *weakly binomial* [108] and we demonstrate several bounds on quantum and classical codes which satisfy this property. While this is primarily a "no-go" result, the techniques we develop could be useful for researchers studying qLDPC. They could be used to rule out candidate constructions, or studied further to find additional bounds stemming from other definitions of expanders.

1.4.2 Problem 2

There is a way (classical to quantum construction) to construct a quantum code $\mathcal{C}(G, C)$ given a classical code C and a graph G [106]. Any attempt to resolve qLDPC using this construction would be futile: the distance of $\mathcal{C}$ is upper bounded by the maximum vertex degree while the stabilizer weight is lower bounded by the maximum vertex degree. Hence, local codes derived from this construction must have poor distance or good distance codes must be non-local. However, it is possible that 'barely' non-local codes could recover from random errors with high probability: In the random error setting, any particular set of $O(1)$ qubits are disturbed with probability $\Omega(1)$, so a local code cannot correct random errors with high probability. A graph with degree $\sim \log(n)$, however, could potentially recover

from random errors since each of these sets is disturbed with at most $1/poly(n)$ probability.

The analysis is more refined than suggested above, since any direct union bound will not work. There are exponentially many subsets and each one fails with only inverse polynomial probability. For this problem, we provide a random ensemble of graphs G and a random ensemble of codes C such that graphs will have maximum vertex degree $O(\log(n))$ with high probability. These are Erdos-Renyi (ER) graphs with probability $p = \frac{w \ln(n)}{n}$ for connecting vertices. Using some asymptotics, we are able to show our ensemble corrects for the quantum erasure channel, while remaining poly-log local. Along the way we derive some interesting facts for log-sparse classical codes.

1.4.3 Problem 3

The third topic is a construction for a unitary 2-design or "scrambler" that uses many of the core ideas from problem 2. As discussed, unitary designs are naturally related to quantum codes. A good quantum coding circuit naturally makes a good design and vice versa. Hence, it makes sense to examine our Erdos-Renyi graph codes and see if they correspond to good designs. We are able to harness these encoding circuits of near-local states to obtain "near local" approximate-unitary-2-designs. Our final construction has the same or better (asymptotic) parameters than all known constructions in our regime of interest [41, 47, 121], and could have implementation advantages over these designs. While the ER circuits provide the actual scrambling, on the way to analyzing them we study another class of random quantum circuits built from expander graphs. This construction may be of independent interest, although we are not able to demonstrate scrambling with this component alone.

Chapter 2

Preliminaries

2.1 Generic Notation

In order to use notation close to common ones found in the literature, we will "overload" some notations. The meaning of a particular expression will often be context dependent. Throughout the document, all logarithms will be base 2 by default. Binomial coefficients will have their standard definition $\binom{n}{k} = n!/(k!(n-k)!)$, although in Chapter 3 we will interpret them as polynomials and in Chapter 4 we will represent them with continuous (Gamma) functions.

Suppose we randomly sample from some set of objects $\{\mathcal{O}_k\}$ and we obtain $\mathcal{O}_k$ with probability p_k. We will denote this ensemble as $\{p_k, \mathcal{O}_k\}$. Given two operators A and B on some Hilbert space we will use this notation also for the anti-commutator: $\{A, B\} = AB + BA$.

The notation $[A, B]$ will either refer to some interval in $\mathbb{R}$ or if A and B are operators it will refer to the commutator $[A, B] = AB - BA$.

The notation $\langle ... \rangle$ will have three possible interpretations. If we write $\langle S_1, ..., S_q \rangle$ for some set of $\{S_i\} \subseteq \mathcal{G}$ belonging to a group, it will denote the smallest possible subgroup of $\mathcal{G}$ containing the elements S_i. If P_a and P_b are both members of the Pauli group (defined later in this chapter), we will use $\langle ... \rangle$ to denote the relation:

$$\langle P_a, P_b \rangle = \begin{cases} 1 \text{ if } \{P_a, P_b\} = 1 \\ \\ 0 \text{ if } [P_a, P_b] = 0 \end{cases}$$

We will *never* consider a group generated by the two Pauli operators, so $\langle P_a, P_b \rangle$ should always be interpreted with the expression above. Lastly, the notation $\langle ... \rangle$ may have some

meaning according to the "bra-ket" notation, which will also be elaborated on later in the chapter.

Given some subgroup $\mathcal{S} \subseteq \mathcal{G}$, we will use $N(\mathcal{S})$ to denote the "normalizer":

$$N(\mathcal{S}) := \{\sigma \in \mathcal{G} : \forall s \in \mathcal{S} \;\; [\sigma, s] = 0\}$$

Given two distributions $\{p_i\}$ and $\{q_i\}$ on the same outcome space, we will use $\Delta(p, q)$ to denote their *Statistical Distance*: $\Delta(p, q) = \sum_i |p_i - q_i|$. Given some set of objects $E = \{\mathcal{O}_k\}$ we will use the notation $X \sim U[E]$ to denote a random variable with a uniform distribution over the set of distinct objects in E. Alternatively, if $\mathcal{E} = \{p_k, \mathcal{O}_k\}$ is some ensemble we will denote $X \sim \mathcal{E}$ as a random variable distributed according to the ensemble(i.e. $X = \mathcal{O}_k$ with probability p_k).

We will use the notation $supp(\mathcal{O})$ to denote the support of an object $\mathcal{O}$. The precise meaning will be context dependent. A binary vector has support consisting of locations on which it is 1. A Pauli operator has support on subsystems where it is not the identity.

The Kronecker delta function has it's standard interpretation. Given a set X and i, $j \in X$ we define

$$\delta_{ij} = \begin{cases} 1 \text{ if } i = j \\ \\ 0 \text{ otherwise} \end{cases}$$

We will use the logical negation symbol ($\neg$) to indicate *set complement*. Given $A \subseteq \Omega$, we denote $\Omega \setminus A$ as $\neg A$.

2.2 Quantum Mechanics

2.2.1 Qubits and Measurement

The fundamental unit of quantum information is the qubit (quantum bit). It corresponds to a classical bit in superposition. A classical bit can either be in the state 0 or the state 1. A quantum bit can be in the state $|0\rangle$ or the state $|1\rangle$ or any complex linear combination of these two states:

$$|0\rangle = \begin{bmatrix}1\\0\end{bmatrix} \qquad |1\rangle = \begin{bmatrix}0\\1\end{bmatrix} \qquad\qquad |\psi\rangle = \begin{bmatrix}\alpha\\\beta\end{bmatrix} \tag{2.1}$$

that satisfies $\alpha, \beta \in \mathbb{C}$ and $|\alpha|^2+|\beta|^2 = 1$. We will use the notation $|\psi\rangle$ to denote some generic qubit state, or the collection of many qubits where the actual state under consideration should be obvious.

While a qubit can be in any normalized superposition, that superposition is fundamentally unknowable in the sense that one can only obtain information from a qubit by measuring it. A measurement corresponds to a 'query' of the quantum system in some basis of the user's choosing. After the measurement, the state $\alpha|0\rangle + \beta|1\rangle$ collapses to either the state $|0\rangle$ or the state $|1\rangle$ with probability governed by

$$\mathbb{P}\Big[\alpha|0\rangle + \beta|1\rangle \to |0\rangle\Big] = |\alpha|^2 \qquad \mathbb{P}\Big[\alpha|0\rangle + \beta|1\rangle \to |1\rangle\Big] = |\beta|^2 \tag{2.2}$$

We could have chosen any other orthonormal basis and also executed a measurement. For instance we could have chosen the basis:

$$|+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \qquad |-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}} \tag{2.3}$$

In order to determine the measurement probabilities, we can rewrite our state:

$$\alpha|0\rangle + \beta|1\rangle = \left(\frac{\alpha+\beta}{\sqrt{2}}\right)|+\rangle + \left(\frac{\alpha-\beta}{\sqrt{2}}\right)|-\rangle \tag{2.4}$$

Hence,

$$\mathbb{P}\Big[\alpha\,|0\rangle + \beta\,|1\rangle \to |+\rangle\Big] = \left|\frac{\alpha+\beta}{\sqrt{2}}\right|^2 \qquad \mathbb{P}\Big[\alpha\,|0\rangle + \beta\,|1\rangle \to |-\rangle\Big] = \left|\frac{\alpha-\beta}{\sqrt{2}}\right|^2 \tag{2.5}$$

We are treating measurement as a "black box". It is some process which collapses the state with the probabilities described above. Indeed, describing the dynamics leading to measurement is the most important open problem in quantum foundations, normally referred to as *the measurement problem*. Note that the overall phase of a quantum state does not effect any measurement statistics. $e^{i\phi}\,|\psi\rangle$ and $|\psi\rangle$ have exactly the same measurement probabilities for *any* observable. As a result, we say that the states $e^{i\phi}\,|\psi\rangle$ and $|\psi\rangle$ are "the same". This is a general feature of quantum systems, they are only defined up to an overall phase.

What we have described here is a special case of a special case of the most general kind of measurement a quantum system can undergo. In this thesis, we will only need to generalize this to one level to a PVM or projection-valued measurement. This is a set of projectors $M = \{P_j\}$ (Hermitian matrices with eigenvalues either 0 or 1) that satisfy:

$$P_i P_j = \delta_{ij} P_i \qquad \sum_j P_j = \mathbb{I} \tag{2.6}$$

Measuring value j corresponds to projecting onto the subspace spanned by nonzero eigenvectors of P_j. We calculate the probabilities of particular measurements as follows. Let $P_j\,|\psi\rangle = \sum_q c_q\,|q\rangle$ for some orthonormal basis $\{|q\rangle\}$. Then,

$$\mathbb{P}\Big[M(|\psi\rangle) \to j\Big] = \sum_q |c_q|^2 =: \langle\psi|P_j|\psi\rangle \tag{2.7}$$

The RHS is written in the "bra-ket" notation. After the measurement, if we obtain outcome j, the new quantum state is proportional to $P_j\,|\psi\rangle$. We obtain the "actual" state by normalizing.

If we are considering a collection of n qubits, it is easy to extend these ideas. A set of

bits can be in any state $(a_1, a_2, ..., a_n)$ where each $a_i \in \mathbb{F}_2$. Quantum bits can be in any of these states $|a_1, a_2, \ldots a_n\rangle$, or any normalized superposition of these states:

$$|\psi\rangle = \sum_k \alpha_k |a_1^k, a_2^k, \ldots a_n^k\rangle \tag{2.8}$$

where $\sum_k |\alpha_k|^2 = 1$. Said another way, quantum systems compose via a tensor product. If a single qubit is a vector space $\mathbb{C}^2$ then a collection of n qubits is the tensor product of all of these vector spaces:

$$\mathbb{C}^2 \otimes \ldots \otimes \mathbb{C}^2 = [\mathbb{C}^2]^{\otimes n} \tag{2.9}$$

Measurements are the same for this case. Valid measurements correspond to sets of projectors as described in Equation (2.6), and measurement probabilities are calculated in the same way. We will use the term Hilbert space interchangeably with vector space to describe quantum states. While, for infinite dimensional systems, a Hilbert space has more structure than a vector space in every context we study here (finite dimensions) they are the same.

2.2.2 Noisy Quantum Systems

It is important to describe the "density matrix" formalism, which is a common way to denote noisy quantum states. For this, we will need to more carefully describe the "bra-ket" notation. We use "bra" notation to denote vectors in the dual space. Given some vector space, the dual is the (linear) space of functionals on that vector space. These are all linear functions which map from the vector space to the underlying field. By linearity, this space has the same dimension as the vector space on which it acts. Given some basis for V, $\{|\phi_i\rangle\}$ we can construct a basis for V^* using the functions:

$$f_{|\phi_j\rangle}(|\psi\rangle) = f_{|\phi_j\rangle}(a_1 |\phi_1\rangle + ... + a_j |\phi_j\rangle + ...) = a_j \tag{2.10}$$

where we have determined the action of $f_{|\phi_j\rangle}(|\psi\rangle)$ by writing $|\psi\rangle$ in the basis $\{|\phi_i\rangle\}$. The bra $\langle\phi_j|$ is simply a way to denote this canonical dual element corresponding to $|\phi_j\rangle$. We would evaluate it using $(\langle\phi_j|)\,|\phi_i\rangle = \langle\phi_j|\phi_i\rangle = \delta_{ij}$:

$$\langle\phi_j|\,(|\psi\rangle) = (a_1\,\langle\phi_j|\phi_1\rangle + ... + a_j\,\langle\phi_j|\phi_j\rangle + ...) = a_j \tag{2.11}$$

A linear operator on a vector space V is a linear function which acts on V to produce another vector in V. Given some orthonormal basis $\{|\phi_i\rangle\}$ there is again a canonical basis for linear operators, it corresponds to functions which map one vector in the basis to another vector in the basis and the rest of the basis vectors to zero. If, for instance f maps $|\phi_1\rangle \to |\phi_2\rangle$ we would write it in bra-ket notation as $|\phi_2\rangle\,\langle\phi_1|$. A general linear operator can be written as:

$$\sum_{ij} c_{ij}\,|\phi_i\rangle\,\langle\phi_j| \tag{2.12}$$

Quantum systems are "unknowable" in the sense that the only information that can be learned from them come from measurements as described above. The only time when writing the state of a qubit as a superposition "makes sense" is when we have designed the dynamics to exactly produce that state. A more realistic assumption is that we have some uncertainty about a quantum state. This could be because we do not have an exact knowledge of the dynamics, or because we have designed the circuit to answer some question that we do not know the answer to. We can quantify our state of knowledge of a quantum system using a *density matrix*. Suppose we have a quantum system, and we know that it is in state $|\psi_k\rangle$ with probability p_k. So, we have some ensemble of quantum states $\{p_k, |\psi_k\rangle\}$. The density matrix corresponding to this ensemble is:

$$\rho := \sum_k p_k\,|\psi_k\rangle\,\langle\psi_k| \tag{2.13}$$

Suppose we are given the density matrix from Equation (2.13) and we measure in some orthonormal basis $\{|\phi_k\rangle\}$. The probability we measure some state $|\phi\rangle \in \{|\phi_k\rangle\}$ can be determined via Bayes rule:

$$\mathbb{P}\Big[\text{Measure } |\phi\rangle\Big] = \sum_k \mathbb{P}\Big[\text{Measure } |\phi\rangle \Big| \text{state is } |\psi_k\rangle\Big]\mathbb{P}\Big[\text{state is } |\psi_k\rangle\Big] \tag{2.14}$$

$$\sum_k p_k |\langle\phi|\psi_k\rangle|^2 = \sum_k p_k \langle\phi| \Big(|\psi_k\rangle\langle\psi_k| \Big) |\phi\rangle = Tr\Big[\rho |\phi\rangle\langle\phi|\Big] \tag{2.15}$$

In general, given some measurement $M = \{P_j\}$, we can calculate the probability of measuring j for observable M in the same way:

$$\mathbb{P}\Big[M(\rho) \to j\Big] = Tr(P_j\rho) \tag{2.16}$$

and the residual density matrix after measurement is proportional to $P_j\rho P_j$.

While density matrices allow for a very convenient description of quantum systems, their real strength is in describing composite quantum systems. Suppose we have some density matrix on two quantum systems A and B. Let $\{|i_A\rangle\}$ and $\{|j_B\rangle\}$ be orthonormal bases for A and B respectively. By definition of the tensor product, any state in the composite system can be written as a linear combination of $\{|i_A\rangle \otimes |j_B\rangle\} = \{|i_A j_B\rangle\}$. Hence, we can assume the density matrix has the form:

$$\rho = \sum c_{i_Z, j_B, i'_A, j'_B} |i_A j_B\rangle\langle i'_A j'_B| \tag{2.17}$$

Suppose we are interested in determining some observable only on subsystem A. The probability of some outcome j can be written as $p_j := Tr[P_j \otimes \mathbb{I}_B\rho]$. Suppose P_j has eigenvectors $\{|\phi_k\rangle\}$. We can write:

$$p_j = Tr\Big[\Big(\sum_k |\phi_k\rangle\langle\phi_k|\Big) \otimes \Big(\sum_j |j_B\rangle\langle j_B|\Big)\rho\Big] = \sum_{j,k} \langle\phi_k| \otimes \langle j_B| \rho |\phi_k\rangle \otimes |j_B\rangle \tag{2.18}$$

Define the operator:

$$\rho_{q_B} = \sum c_{i_Z,q_B,i'_A,q_B} \left|i_A q_B\right\rangle \left\langle i'_A q_B\right| \tag{2.19}$$

This is the (un-normalized) density matrix we would have obtained if subsystem B had been measured in the basis $\{|j_B\rangle\}$ with outcome q_B. We can write:

$$p_j = \sum_k \langle\phi_k| \Big(\sum_{q_B} \rho_q \Big) |\phi_k\rangle = Tr[P_j \rho_A] \tag{2.20}$$

where $\rho_A = \sum_q \rho_q$. ρ_A is the *reduced density matrix* for subsystem A. As the above discussion shows, the probability of outcomes for any measurement can be predicted from ρ_A alone. Note further that we could have obtained the same ρ_A by assuming that the subsystem B was measured in the computational basis, or any basis for that matter. This observation is known as the *principle of implicit measurement*. If we have a composite density matrix on two subsystems A and B, and we lose access to subsystem B, we can assume that that subsystem has been measured in some basis of our choosing. Since the dynamics of B do not effect our reduced density matrix for A, they do not actually effect any predictions we would make on subsystem A, and hence the dynamics of B does not *really* effect any quantum information present in A. We say that the mixed state ρ_A is "the same" for all possible evolutions of B. This notion will be particularly important for Chapter 4.

Fact 2.1 (Principle of Implicit Measurement)**.** *Let ρ be some density matrix supported on two subsystems A and B. If subsystem B is lost then we can assume it has been measured in some basis of our choosing.*

2.2.3 Dynamics for Pure States

The allowed set of quantum dynamics which map pure states to pure states includes measurement, as described as well as any other linear map which preserves normalization:

Definition 2.2 (Unitary matrices)**.** *Unitary matrices are the set of matrices* $\{U\}$ *which satisfy:*

$$U^\dagger U = \mathbb{I}$$

where $U^\dagger$ *is the matrix satisfying* $U^\dagger_{ij} = U^*_{ji}$*.*

It is not hard to see that unitaries preserve normalization since $||U|\psi\rangle||^2 = \langle\psi|U^\dagger U|\psi\rangle = \langle\psi|\psi\rangle = ||\,|\psi\rangle\,||^2$. We will be interested in some specific unitaries, the Pauli group. First we define the Pauli matrices, which are single qubit unitaries that are also Hermitian:

$$X = \begin{matrix} \\ |0\rangle \\ |1\rangle \end{matrix}\!\!\begin{matrix} \begin{matrix} |0\rangle & |1\rangle \end{matrix} \\ \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{matrix} \quad Y = \begin{matrix} \\ |0\rangle \\ |1\rangle \end{matrix}\!\!\begin{matrix} \begin{matrix} |0\rangle & |1\rangle \end{matrix} \\ \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \end{matrix} \quad Z = \begin{matrix} \\ |0\rangle \\ |1\rangle \end{matrix}\!\!\begin{matrix} \begin{matrix} |0\rangle & |1\rangle \end{matrix} \\ \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{matrix} \tag{2.21}$$

We define $\mathcal{P}_1 = \{\mathbb{I}, X, Y, Z\}$.

It is easy to see that the Pauli matrices satisfy a simple anti-commutation relation:

$$\forall A, B \in \{X, Y, Z\} \text{ such that } A \neq B \text{ it holds that } AB + BA = 0 \tag{2.22}$$

This relation shows that given some initial state $|\psi\rangle$ we can only visit three other states using the Pauli matrices. Since overall phase does not change the state, the only achievable states are $|\psi\rangle$, $X|\psi\rangle$, $Y|\psi\rangle$ and $Z|\psi\rangle$. Any other dynamics can be brought to this form using the above relation and the fact that $A^2 = \mathbb{I}$ if $A \in \mathcal{P}_1$. These operators have a simple generalization to multiple qubits:

Definition 2.3. *We define $\mathcal{P}_n$ as the set of operators on $[\mathbb{C}^2]^{\otimes n}$ with the form:*

$$\phi A_1 \otimes A_2 \ldots \otimes A_n$$

where each $A_i \in \mathcal{P}_1$ and $\phi \in \{1, -1, i, -i\}$.

In many places, we will index the Pauli operators with a subscript. We will denote $P_0 = \mathbb{I}$ and some other arbitrary indexing for the rest. Naturally, $x \neq x' \Rightarrow P_x \neq P_{x'}$. The particular indexing for nonzero P_x will not matter for us, it is a technical tool to keep track of Pauli matrices in sums.

In analogy to the Hamming weight of classical coding theory, we can define the Pauli weight of a given operator as the number of non-identity terms:

Definition 2.4 (Pauli Weight)**.** *If $\sigma \in \mathcal{P}_n$ then let $\sigma = \phi A_1 \otimes .... \otimes A_n$. We define $|\sigma| = |\{A_i : A_i \neq \mathbb{I}\}|$.*

One of the important facts about Pauli operators is that they form a complete basis for the space of operators on $[\mathbb{C}^2]^{\otimes n}$. This is easy to see since they consist of a sufficiently large set whose elements are orthogonal under the Hilbert-Schmidt product $\langle A, B\rangle_{HS} := Tr(A^\dagger B)$.

Fact 2.5 (Pauli Completeness)**.** *Let $B : [\mathbb{C}^2]^{\otimes n} \to [\mathbb{C}^2]^{\otimes n}$, then B can be written as $B = \sum_x c_x P_x$ where $c_x \in \mathbb{C}$ and $P_x \in \mathcal{P}_n$.*

Two Pauli operators commute or anti-commute depending on the relation Equation (2.22), evaluated at each term in the tensor product separately. To reiterate, given two Pauli operators $P_i, P_j \in \mathcal{P}_n$, we will use the notation:

$$\langle P_i, P_j \rangle = \begin{cases} 1 \text{ if } \{P_i, P_j\} = 1 \\ 0 \text{ if } [P_i, P_j] = 0 \end{cases} \tag{2.23}$$

Any non-identity Pauli Pauli operator commutes with exactly half of the other Pauli operators. This is easy to see with statistical arguments:

Proposition 2.6. *Let* $\sigma = A_1 \otimes ... \otimes A_n \in \mathcal{P}_n, \neq \mathbb{I}$ *and* $\gamma \in B_1 \otimes ... \otimes B_n \sim U[\mathcal{P}_n]$. *Define a random variable* X*:*

$$X = \begin{cases} 1 \text{ if } \{\sigma, \gamma\} = 0 \\ 0 \text{ if } [\sigma, \gamma] = 0 \end{cases} \tag{2.24}$$

Then $\mathbb{E}[X] = 1/2$.

Proof. Define

$$X_i = \begin{cases} 1 \text{ if } \{A_i, B_i\} = 0 \\ 0 \text{ if } [A_i, B_i] = 0 \end{cases} \tag{2.25}$$

and let $supp(\sigma)$ be the subsystems on which σ is not the identity. Since $\mathbb{I}$ commutes with all the other Pauli matrices,

$$X = \sum_{i \in supp(\sigma)} X_i \mod 2 \tag{2.26}$$

It is clear that $\mathbb{P}[X_i = 1] = 1/2$ if $i \in \text{supp}(\sigma)$ since the matrix B_i has probability $1/2$ of commuting with some (non-identity) A_i. It holds by induction that:

$$\mathbb{P}[X = 1] = \frac{1 - (1 - 2\,(1/2))^n}{2} \quad \text{and} \quad \mathbb{P}[X = 0] \frac{1 - (1 + 2\,(1/2))^n}{2} \tag{2.27}$$

The proposition follows. □

We will need another special set of unitaries known as the *Clifford Group*. These are the unitaries which conjugate Pauli matrices to other Pauli matrices:

Definition 2.7 (Clifford Group)**.** $\mathcal{C}\ell_n$ *is the set of unitaries* $\{U\}$ *on* n *qubits which satisfy:*

$$\forall \sigma \in \mathcal{P}_n \qquad U \sigma U^\dagger \in \mathcal{P}_n$$

Clifford dynamics are also easy to describe, in fact one can classically simulate [76] the dynamics of simple states under the full set of Clifford operations. We will need some specific Clifford unitaries at different points in the thesis:

$$H := \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} \qquad CP := \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & -1\end{bmatrix} \qquad S := \begin{bmatrix}1 & 0\\ 0 & i\end{bmatrix}$$

It will be important to have conjugation relations handy for these specific Clifford operators. The effect of $UPU^\dagger$ for any of the above and any Pauli operator P can be inferred from the following relations:

$$\begin{gathered} HXH^\dagger = Z \qquad HZH^\dagger = X \\ CP(X\otimes\mathbb{I})CP^\dagger = X\otimes Z \qquad CP(Z\otimes\mathbb{I})CP^\dagger = Z\otimes\mathbb{I} \\ CP(\mathbb{I}\otimes X)CP^\dagger = Z\otimes X \qquad CP(\mathbb{I}\otimes Z)CP^\dagger = \mathbb{I}\otimes Z \\ SXS^\dagger = Y \qquad SZS^\dagger = Z \end{gathered} \tag{2.28}$$

We will specify a notation which is relied on heavily on in Chapter 4 and Chapter 5. Given some vector $\mathbf{e} \in \mathbb{F}_2^n$ we will use $\mathcal{O}_\mathbf{e}$ to indicate the tensor product of copies of $\mathcal{O}$ supported at sites dictated by the support of $\mathbf{e}$.

$$\mathcal{O}_\mathbf{e} := \prod_{j\in supp(\mathbf{e})} \mathbb{I}^{\otimes(j-1)} \otimes \mathcal{O} \otimes \mathbb{I}^{\otimes(n-j)} \tag{2.29}$$

As an example,

$$X_{(1,1,0)} = X\otimes X\otimes\mathbb{I} \tag{2.30}$$

We will also use CP_{ij}. This denotes a controlled phase between qubits i and j.

Dynamics of Noisy Quantum Systems

In a closed quantum system, the only allowed dynamics are exactly as we have described. We are allowed to use unitaries and measurements. However, there are many cases in which we would like to describe a quantum system with uncertainty (density matrix) or we wish to describe a small piece of an overall quantum system. For this we will need a more general kind of quantum map. The most general kind of quantum map is a completely-positive trace-preserving map (CPTP). The set of CPTP maps is the largest set of maps consistent with our probabilistic interpretation of the density matrix.

Suppose we have some map $\mathcal{E}$ on some system A and we append some other subsystem B. Let ρ be any positive operator on the combined Hilbert space of A and B. Complete positivity means that $\mathbb{I}_B \otimes \mathcal{E}(\rho)$ is a positive operator for any ρ (possibly supported on A and B). The spectrum of a density matrix corresponds to a distribution over the eigenvalues. Complete-positivity means that we cannot generate a distribution with any negative probabilities. Similarly, trace preserving implies the normalization of the density matrix after the map. The most general such maps can be written in the following way

Definition 2.8. *We say* $\mathcal{E}(\rho) = \sum_j A_j \rho A_j^\dagger$ *is CPTP if* $\sum_j A_j^\dagger A_j = \mathbb{I}$.

It will be important that maps of the above form are physical, not that all physical maps have the above form, so we only demonstrate the "one-way" implication. It is easy to see the map is trace preserving:

$$Tr\Big[\mathcal{E}(\rho)\Big] = Tr\Big[\sum_j A_j \rho A_j^\dagger\Big] = Tr\Big[\sum_j A_j^\dagger A_j \rho\Big] = Tr(\rho)$$

To see that $\mathcal{E}(\rho)$ is positive it is enough to show that $\langle\psi|\mathcal{E}(\rho)|\psi\rangle > 0$ for all $|\psi\rangle$. Define

the un-normalized state $|\psi_j\rangle = A_j^\dagger |\psi\rangle$, then $\langle\psi|\mathcal{E}(\rho)|\psi\rangle = \sum_j \langle\psi_j|\rho|\psi_j\rangle$. Each term in the sum is positive, so the sum itself must be positive.

2.2.4 Distance Measures in Quantum Information

For density matrices, the most commonly used distance metric is the *Trace distance*. Given two density matrices ρ and σ the trace distance is proportional to the sum of the absolute eigenvalues of the matrix $\rho - \sigma$.

Definition 2.9. *Given two density matrices ρ and σ we define the Trace distance between them as:*

$$T(\rho, \sigma) = ||\rho - \sigma||_{Tr} = \frac{1}{2} Tr\left[\sqrt{(\rho - \sigma)^2}\right] = \frac{1}{2}\sum_i |\lambda_i| \tag{2.31}$$

where $\{\lambda_i\}$ is the set of eigenvalues of $\rho - \sigma$.

Note that this is also referred to as the Schatten 1-Norm. As a norm, it satisfies the triangle inequality[1]:

$$||A + B||_{Tr} \leq ||A||_{Tr} + ||B||_{Tr} \tag{2.32}$$

The Trace Distance is a measure of the "distinguishability" of the ensembles described by ρ and σ. If the trace distance is 0, then the two ensembles are identical (and hence indistinguishable) while if it is 1 (it's maximum value) then ρ and σ can be perfectly distinguished with measurement. In the latter case the eigenvectors of ρ and σ are orthogonal.

We will also need a notion of distance for quantum channel (CPTP) maps. Seemlingly, the most natural definition is to ask for the "worst-case" Trace distance for the outputs of the channels given the same input density matrix. The problem with this definition is that if we

[1] This is easy to verify with the Rayleigh quotient

append another subsystem, we can drastically alter the distance. $||\mathbb{I} \otimes \Phi_1(\rho) - \mathbb{I} \otimes \Phi_2(\rho)||_{Tr}$ may be very different from $||\Phi_1(\rho) - \Phi_2(\rho)||_{Tr}$ [101]. We can resolve this problem with the *Diamond Norm*:

Definition 2.10. *Let Φ_1 and Φ_2 be two CPTP maps with the same finite dimensional domain/range. Suppose they both act on some Hilbert space $\mathcal{H}$.*

$$||\Phi_1 - \Phi_2||_\diamond := \max_\rho ||\mathbb{I} \otimes \Phi_1(\rho) - \mathbb{I} \otimes \Phi_1(\rho)||_{Tr} \tag{2.33}$$

where maximization is taken over density matrices ρ on two copies of $\mathcal{H}$

If we maximize the Trace distance using two copies of a quantum system, it is enough to guarantee that appending more subsystems will not increase the distance [101]. In the last chapter, we will be interested in computing the diamond norm between two Pauli channels. Let $\Phi_1(\rho) = \sum_i p_i P_i \rho P_i^\dagger$ and $\Phi_2(\rho) = \sum_i q_i P_i \rho P_i^\dagger$ where each P_i is some Pauli operator. From the triangle inequality for the Trace norm, it is easily seen that the diamond distance between these two channels is upper bounded by the statistical distance (ℓ_1) between the distributions $\{p_i\}$ and $\{q_i\}$.

Lemma 2.11 ([139]). *Let $\Phi_1(\rho) = \sum_i p_i P_i \rho P_i^\dagger$ and $\Phi_2(\rho) = \sum_i q_i P_i \rho P_i^\dagger$ where each $P_i \in \mathcal{P}_n$. Then,*

$$||\Phi_1 - \Phi_2||_\diamond \leq \Delta(p, q) = \sum_i |p_i - q_i| \tag{2.34}$$

Proof.

$$||(\mathbb{I}_\mathcal{H} \otimes \Phi_1 - \mathbb{I}_\mathcal{H} \otimes \Phi_2)\rho||_{Tr} = \left|\left|\sum_i (p_i - q_i) \mathbb{I}_\mathcal{H} \otimes P_i \rho \mathbb{I}_\mathcal{H} \otimes P_i^\dagger\right|\right|_{Tr} \leq \tag{2.35}$$
$$\sum_i |p_i - q_i| \left|\left|\mathbb{I}_\mathcal{H} \otimes P_i \rho \mathbb{I}_\mathcal{H} \otimes P_i^\dagger\right|\right|_{Tr} = \sum_i |p_i - q_i|$$

where the inequality follows from Equation (2.32). □

2.3 Codes

The majority of this thesis is concerned with the study of locally generated quantum codes. We will restrict our attention to binary linear codes, and their quantum analogs, qubit stabilizer codes. The definitions and many of the ideas presented in this thesis are easily extended to the case where codes are defined over higher dimensional subspaces [92], for concreteness we will focus on the binary case. First, we define Hamming distance between two words in a binary vector space:

Definition 2.12. *Given* $\mathbf{x} = (x_1, ..., x_n) \in \mathbb{F}_2^n$, *we define* $|\mathbf{x}| = |\{x_i : x_i = 1\}$.

A code C is simply a subspace of the vector space $\mathbb{F}_2^n$. Naturally it has some minimal basis $\{\mathbf{c}_1, ..., \mathbf{c}_k\}$. We can represent this using the generator matrix G, which is a $\mathbb{F}_2^{k \times n}$ matrix with row i equal to $\mathbf{c}_i$. We can think about C as a way to encode k bits into n bits with $k \leq n$ using the mapping:

$$(a_1, ..., a_k)G = \sum_{i=1}^{k} a_i \mathbf{c}_i \tag{2.36}$$

where $(a_1, ..., a_k)$ is the string to be encoded. Let us define $d = \min_{\mathbf{c} \in C} |\mathbf{c}|$. It is easy to see that the bits $(a_1, ..., a_k)$ can be recovered from any set of the bits of size at least $n - (d - 1)$. If we write down the generator matrix and "zero-out" the columns corresponding to the lost bits, the generator matrix must still have full rank, otherwise we can find a word in the code of weight smaller than the distance. It follows that we can invert what is left of the generator matrix to obtain the bits $(a_1, ..., a_k)$. Another interpretation of the distance is the smallest size set with which we can change one valid codeword to another. If we have access to a certain set of bits of size d, say the support of the smallest weight word $\mathbf{c}$, then we can

exchange $\mathbf{0} \leftrightarrow \mathbf{c}$. If this was possible with any smaller set of bits, then we would be able to find a word smaller than the distance of the code (just add the two valid words together).

We can associate the code to a dual space $C^\perp$ where $C^\perp$ consists only of vectors orthogonal to all words in C:

Definition 2.13. *Given some subspace $C \subseteq \mathbb{F}_2^n$, we define:*

$$C^\perp = \{\mathbf{c}^\perp \in \mathbb{F}_2^n : \mathbf{c}^\perp \cdot \mathbf{c} = \sum_i c_i^\perp c_i = 0 \ \forall \ \mathbf{c} \in C\}$$

By standard linear algebra facts, codes uniquely correspond to dual spaces and vice versa. It is common to specify a code by it's dual space and not by the generators themselves. Each dual word corresponds to a set of bits which must sum to 0 (over $\mathbb{F}_2$) for any codeword, hence one can think about these as "checking" whether a word is in the code by summing those entries. For this reason, dual elements are often referred to as the "checks" of a code. If a particular binary vector satisfies all checks, then it must be a codeword. It a codeword is not in the code, then it must violate at least one of the checks. In the interest of decoding, we will often want codes which have very small checks, preferably constant size as $n \to \infty$. Hence, we will include a parameter w corresponding to the maximum Hamming weight of the smallest basis for $C^\perp$. Let $\{\mathbf{c}_1^\perp, \mathbf{c}_2^\perp, ..., \mathbf{c}_{n-k}^\perp\}$ be a basis for the code $C^\perp$ of minimal Hamming weight. Let $w = \max_i |\mathbf{c}_i^\perp|$ be the greatest Hamming weight in the set. We seek the choice that minimizes w. Now we are in a position to formally define the code C:

Definition 2.14. *We say C is a $[n, k, d, w]$ code if*

1. *C is a subspace of $\mathbb{F}_2^n$ of dimension k*

2. $\min_{\mathbf{c} \in C} |\mathbf{c}| = d$

3. *Let* $A = \{\mathbf{c}_1^\perp, \mathbf{c}_2^\perp, ..., \mathbf{c}_{n-k}^\perp\}$ *be the minimal size basis minimizing* $\max_{\mathbf{c}_i^\perp} |\mathbf{c}_i^\perp|$. *Then,* $w \geq \max_{\mathbf{c}_i^\perp} |\mathbf{c}_i^\perp|$.

2.3.1 Quantum Codes

A quantum code can be defined in much the same way as a classical code. It is a subspace of the (Hilbert) space with some well defined notion of redundancy. We will formally define it using the "checks" rather than the codewords themselves. Quantum (linear) codes come with Pauli measurements that allow us to check if a particular state is in the code. These measurements have the effect of "stabilizing" words in the code, since words (quantum states) in the code should be unaffected by the measurements. The projectors for the stabilizing measurements are of the form

$$M_j = \left\{ \frac{\mathbb{I} + \sigma_j}{2}, \frac{\mathbb{I} - \sigma_j}{2} \right\} \text{ with } \sigma_j \in \mathcal{P}_n$$

Requiring "stabilization" enforces $\forall |\psi\rangle \in \mathcal{C},\ \ P_j |\psi\rangle = |\psi\rangle \Rightarrow \sigma_j |\psi\rangle = |\psi\rangle$. Hence, the Pauli operator σ_j is referred to as a *stabilizer* of the code $\mathcal{C}$. It is easy to see the set of stabilizers is closed under (matrix) multiplication and hence forms a group. From this observation it becomes clear that different stabilizers must commute, otherwise $|\psi\rangle = \sigma_i\sigma_j |\psi\rangle = -\sigma_j\sigma_i |\psi\rangle = -|\psi\rangle$. We are now in a position to define a stabilizer quantum code. Note that in many places we will leave off the parameter w if it is not relevant.

Definition 2.15. *We define a* $[[n, k, d, w]]$ *quantum code* $\mathcal{C}$ *as subspace of* $(\mathbb{C}^2)^{\otimes n}$ *of dimension* 2^k *such that there is a subgroup* $\mathcal{S} \subseteq \mathcal{P}_n$ *that satisfies:*

- $\forall s_1, s_2 \in \mathcal{S},\ [s_1, s_2] = 0$

- $\forall s \in \mathcal{S}, |\psi\rangle \in \mathcal{C}: \quad s|\psi\rangle = |\psi\rangle$
- *There is a generating set of* $\mathcal{S}$ *with maximum Pauli weight* w
- d *is the Pauli weight of the smallest element of* $N(\mathcal{S}) \setminus \mathcal{S}$.

If we have some quantum state $|\psi\rangle \in [[n, 0, \sim]]$ it must be the *only* state in the code. It has some set of n commuting stabilizers which define it. We will refer to such a code or state as a *stabilizer state*.

The set $N(\mathcal{S}) \setminus S$ is the set of *logical* operators. They correspond to Pauli operators that can change one state in the code $|\psi\rangle \in \mathcal{C}$ to some other valid state in the code $|\phi\rangle \in \mathcal{C}$. If $\sigma \in N(\mathcal{S}) \setminus \mathcal{S}$ had no effect on any word in $|\psi\rangle \in \mathcal{C}$, then it would be in the stabilizer, since it commutes with $\mathcal{S}$ by assumption. Accordingly, distance has the same interpretation for both classical and quantum codes. It is the smallest set of bits/qubits such that we can change one valid codeword to another by acting only on that set.

There is another parallel with classical coding theory: if an adversary alters some known set of bits Q with $|Q| < d$ the code is correctable. More generally, we can correct any known set of disturbed qubits Q if there are no logical operators supported on Q alone:

Lemma 2.16. *Let* $\mathcal{C} \in [[n, k, d]]$ *with stabilizer* $\mathcal{S}$*, and let* Q *be some set* $Q \subset [n]$*. Define:*

$$N(\mathcal{S})_Q := \left\{\sigma \in N(\mathcal{S}) : supp(\sigma) \subseteq Q\right\} \text{ and } \mathcal{S}_Q := \left\{\sigma \in \mathcal{S} : supp(\sigma) \subseteq Q\right\}$$

Then,

$$\begin{array}{r} \exists CPTP \text{ map } \mathcal{R} \text{ such that} \\ \text{for all } |\psi\rangle \in \mathcal{C}, \text{ and for all } E \in \mathcal{P}_n \text{ satisfying } supp(E) \subseteq Q \\ \text{it holds that } \mathcal{R}\Big(E\,|\psi\rangle\,\langle\psi|\,E^\dagger\Big) = |\psi\rangle\,\langle\psi| \end{array} \Leftrightarrow N(\mathcal{S})_Q = \mathcal{S}_Q$$

While the above may seem hard to parse, it is saying that there is a recovery operation $\mathcal{R}$ that recovers from any possible error supported on Q if and only if there are no logical operators supported on Q. The proof is straightforward. We can break up $\mathcal{P}_{|Q|}$ into cosets of $N(\mathcal{S})_Q$. Measurements of the stabilizers correspond to syndrome values. Each coset corresponds to exactly one set of syndrome values. So, we can simply choose any other operator in the same coset of E (say E') to get $E'E\left|\psi\right\rangle$. EE' must be in the normalizer and hence the stabilizer by hypothesis. More general noise operations can be dealt with using the Pauli basis completeness [123].

What we have defined above is the most general kind of (linear) quantum code. CSS codes [36,153] are a particular kind of stabilizer code with some additional structure in their stabilizer group which allows one to think of them as pairs of classical codes.

Definition 2.17. *$\mathcal{C}$ is a CSS code if it is a stabilizer code with stabilizer $\mathcal{S}$ where $\mathcal{S}$ "splits up" into two sets $\mathcal{S}_X$ and $\mathcal{S}_Z$ satisfying:*

- $\mathcal{S}_X,\ \mathcal{S}_Z \subseteq \mathcal{P}_n$ *and* $\langle \mathcal{S}_X, \mathcal{S}_Z \rangle = \mathcal{S}$[2]
- *If* $s_x = A_1 \otimes ... \otimes A_n \in \mathcal{S}_X$ *then each* $A_i \in \{\mathbb{I}, X\}$
- *If* $s_z = A_1 \otimes ... \otimes A_n \in \mathcal{S}_Z$ *then each* $A_i \in \{\mathbb{I}, Z\}$

A CSS code has a stabilizer that breaks up further into two groups, a group of X operators and a group of Z operators. Let us consider one of the subgroups $\mathcal{S}_X$. We will associate this group to a subgroup (code) of $\mathbb{F}_2^n$ we will denote C_X. Construct the following bijection:

$$s_x = \phi A_1 \otimes ... \otimes A_n \in \mathcal{S}_X \leftrightarrow \mathbf{c}_x = (c_1, c_2, ..., c_n) \in C_X$$

[2]In this context we mean $\langle \mathcal{S}_X, \mathcal{S}_Z \rangle$ as the minimal group generated by $\mathcal{S}_X$ and $\mathcal{S}_Z$.

where $c_i = 1$ iff $A_i = X$. It is clear that these operators multiply in the same way that the binary vectors add, so the space C_X is a subspace. We can go through the same procedure with $\mathcal{S}_Z$ to obtain a different subspace C_Z.

Since $s_x \in \mathcal{S}_X$ and $s_z \in \mathcal{S}_Z$ commute, it is easy to see that the corresponding $\mathbf{c}_x$ and $\mathbf{c}_z$ must be orthogonal $\mathbf{c}_x \cdot \mathbf{c}_z = 0$. This forces our linear codes to satisfy $C_Z \subseteq C_X^\perp$ and $C_X \subseteq C_Z^\perp$. The normalizer is the set of Z operators corresponding to $C_X^\perp$, along with the set of X operators corresponding to $C_Z^\perp$. The quantum distance is therefore:

$$d = \min_{\mathbf{x} \in C_X^\perp \setminus C_Z, C_Z^\perp \setminus C_X} |\mathbf{x}| \tag{2.37}$$

In the reverse direction, in order to find a CSS code, it is enough to find a pair of classical codes that are perpendicular to each other. The quantum distance is then related to the codes as described above.

CSS codes have the specific property that bit errors (X errors) and phase errors (Z errors) can be corrected separately. If we have some erred codeword $X_\mathbf{a} Z_\mathbf{b}$, measurements from $\mathcal{S}_X$ will give us information about $Z_\mathbf{b}$ and measurements from $\mathcal{S}_Z$ will give us information about $X_\mathbf{a}$. Assuming the error is correctable, we can determine $Z_\mathbf{b}$ using only $\mathcal{S}_X$ amd hence can decode the CSS code "one half at a time".

2.3.2 Entanglement of Quantum States

Entanglement is a feature of quantum states that quantifies how "separable" a quantum system is. It is an important resource for quantum computation, and seems to behind many of the strange powers afforded to quantum computers. We will describe entanglement only of pure states, since this thesis only requires this level of understanding. Given some

quantum state $|\psi\rangle$ on two subsystems A and B, we say that $|\psi\rangle$ has zero entanglement if it is completely separable, i.e. it can be written as a tensor product of two states $|\phi_1\rangle_A \otimes |\phi_2\rangle_B$. The simple way to understand the degree to which a particular state is entangled is via the Schmidt decomposition. Every pure state on two quantum systems can be written as:

$$|\psi\rangle = \sum_i \sqrt{p_i} \, |\phi_i\rangle_A \otimes |\eta_i\rangle_B$$

where $\{|\phi_i\rangle\}$ and $\{|\eta_i\rangle\}$ are orthonormal sets of vectors. Clearly, $\{p_i\}$ must be some distribution.

If there is only one term in this sum, then the state is seperable and hence not entangled. If there are many terms in the sum, then we say that the state is "very far" from being entangled. The degree to which something is entangled depends both on the schmidt number (the number of terms) and the distribution $\{p_i\}$. We could imagine if some p_0 is very close to 1 and the rest are very small, we would not say such a state is very entangled since most of the terms are barely represented. The conventional way to formalize these notions is to associate the entanglement with the entropy of the distribution $\{p_i\}$. If this distribution has large entropy, then the quantum state would have large entanglement and vice versa. This has a convenient notation using the reduced density matrices of the two subsystems:

$$\rho_A = \sum_i p_i \, |\phi_i\rangle \langle\phi_i|_A \qquad \rho_B = \sum_i p_i \, |\eta_i\rangle \langle\eta_i|_B$$

We can evaluate the entropy by first constructing the matrix $\log(\rho_A)$, which is $\sum_i \log(p_i) \, |\phi_i\rangle \langle\phi_i|_A$, and then multiplying by the original matrix and taking the trace:

Definition 2.18 (Entanglement Entropy)**.** *Let $|\psi\rangle$ be a pure quantum state on two subsystems A and B with reduced density matrices ρ_A and ρ_B. We define the entanglement entropy*

as:

$$E^{(A,B)}(|\psi\rangle) = Tr[\rho_A \log(\rho_A)] \tag{2.38}$$

2.4 More Notation and Statistical Facts

The majority of this thesis consists of probabilistic constructions that achieve novel parameters for different quantum processing tasks. As such, probably the most important theorem for us is the *Chernoff-Hoeffding* bound:

Theorem 2.19 ([38]). *Let $P = \sum_{i=1}^{n} P_i$ be a random variable where each $\mathbb{E}[P_i] = p_i$ and let $p = (\sum_i p_i)/n$. Then it holds that:*

$$\mathbb{P}\left(\frac{1}{n}\sum_i P_i \geq p + \varepsilon\right) \leq 2^{-D(p+\varepsilon||p)n} \qquad \mathbb{P}\left(\frac{1}{n}\sum_i P_i \leq p - \varepsilon\right) \leq 2^{-D(p-\varepsilon||p)n}$$

where

$$D(p||q) = p\log\left(\frac{p}{q}\right) + (1-p)\log\left(\frac{1-p}{1-q}\right) \tag{2.39}$$

This bound is very important in many technical disciplines. In information theory, Chernoff-like behavior is referred to as 'typical', invoking Chernoff in this context is invoking 'typicality'. We will refer to the above in this way, or simply as the Chernoff bound at many points in the body of the thesis. Another pillar of our analysis is the *union bound*:

Fact 2.20. *Let $\{A_i\}$ be a set of events in some outcome space. Then,*

$$\mathbb{P}\left[\bigcup_i A_i\right] \leq \sum_i \mathbb{P}[A_i] \tag{2.40}$$

While the following bounds bounds do not qualify as statistical facts, we will make heavy use of them in statistical arguments. First, define the binary entropy function:

$$h(p) := p \log\left(\frac{1}{p}\right) + (1-p)\log\left(\frac{1}{1-p}\right) \tag{2.41}$$

We will need the following standard bounds involving the entropy function:

Proposition 2.21. *Let $1 \leq k \leq n$, $\varepsilon = k/n$ and $x \in (0, 1/2)$.*

- $\binom{n}{k} \leq 2^{h(\varepsilon)n}$
- $\binom{n}{k} \geq \frac{1}{k\sqrt{8n\varepsilon(1-\varepsilon)}} 2^{nh(\varepsilon)}$
- $x < h(x) < 2x\log(1/x)$

We will further need a handful of specific distributions in the document, we specify these now to avoid cluttering up the thesis:

Definition 2.22.

1. *A random variable X is distributed according to $Bern(p)$ ($X \sim Bern(p)$) if it has two outcomes $\{0, 1\}$ and $\mathbb{P}(X = 1) = p$.*
2. *$X \sim Bin(n, p)$ if $X \in \{0, 1, ..., n\}$ and $\mathbb{P}\Big[X = k\Big] = \binom{n}{k} p^k (1-p)^{n-k}$*

2.4.1 Graph Theoretic Notations

A graph is a set of vertices V and a set of edges E where the elements of E are unordered pairs of elements from V. We can denote the connectivity structure with a graph with an adjacency matrix A. Given some vertex set V of size n and some edges set E, we construct

a binary $n \times n$ matrix by associating the rows and columns of the matrix with elements of V. We then place a 1 in the ith row and jth column if vertices i and j are connected.

Given some partition of the vertices (A, B), we can consider this as a "cut" of the graph into two pieces. The edges where one end is in A and the other end is in B form the cut edges. We define the cut matrix as the subset of the adjacency matrix corresponding to a cut of the vertices:

Definition 2.23. *Given a cut (A, B) construct a binary $|A| \times |B|$ matrix A_{cut}. Associate the rows to the vertices of A and the columns to the vertices in B. Set $A_{cut,ij} = 1$ if element $i \in A$ is connected to element $j \in B$.*

There is a similar notion of a "neighbor set" in a graph. Given a set of vertices A, the neighbor set is the set of vertices outside of A that are connected to at least one vertex in A. The formal definition is:

Definition 2.24. *Given a graph $G = (V, E)$ and a subset $S \subseteq V$, denote:*

$$N_G(S) = \{v \in V \setminus S : \exists s \in S \text{ with } (s, v) \in E\}$$

Much of this work is based off of random constructions from randomly chosen graphs. For this we define Erdos Renyi graphs (ER graphs) [64]. Given some number of vertices n and some probability p, we can construct $ER(n, p)$ by beginning with the empty graph, and adding each possible pair (i, j) with probability p independent of the other pairs:

Definition 2.25. *Let $ER(n, p)$ be the graph corresponding to a random $n \times n$ binary adja-*

cency matrix A, where

$$A_{ij} = \begin{cases} Bern(p) \text{ if } i \neq j \\ \\ 0 \text{ if } i = j \end{cases}$$

with each entry independent of the others.

Chapter 3

Topologies of Potential qLDPC Codes

3.1 Introduction

3.1.1 The need for qLDPC

Recall that qLDPC codes with linear distance are not known to exist. We formally state the conjecture that they do exist as follows:

Conjecture 3.1. *(qLDPC) There exists a family of quantum codes* $\{\mathcal{C}_n\}_{n=1}^{\infty}$ *where each* $\mathcal{C}_n \in [[n, k, d_{min}, w]]$ *with* $d_{min} = \Omega(n)$ *and* $w = O(1)$.

Note that in the above we do not have any requirements for the rate k/n of the quantum code. It is unknown whether there exist families of this type for any k, even $k = O(1)$. Classical LDPC codes, however, are well known and have been for some time [72]. Classical LDPC codes are known to have very low complexity decoders [152] for communication over various channels. It is known that these properties are generic [152], or that most randomly sampled LDPC codes will have efficient encoding/decoding schemes given that they are sampled from appropriate distributions. In practice randomly sampled codes can be used to communicate at any (sufficiently large) blocklength. Some locally generated quantum codes are known to have efficient encoding/decoding for average errors [52, 79], but these codes all have sublinear distance (worst-case errors). It is possible that local codes with linear distance will have much stronger decoding properties.

As an additional motivation for this work, it is known that quantum codes with linear distance could be very powerful tools for Fault-Tolerant quantum computing [77, 150, 154]. This is an area of quantum information which seeks to provide resilient computing in the presence of errors in all aspects of the computation. This model assumes that all components

(gates, measurements, wires) are subject to some constant error rate. In particular, qLDPC codes would imply very attractive logical error rates for Gottesman's construction [77]. This rate is exponentially small for local codes with linear distance, while only superpolynomially small for codes with distance $\sqrt{n}$.

Aside from these considerations, I think the problem is interesting in it's own right, from a physical perspective. The Toric code [102] was the first instance of quantum information stored in non-local degrees of freedom. A qLDPC code with linear distance would be an instance of a much stronger notion of this property. Such a quantum state would store information that is robust to all weak forms of noise, including adversarial noise. The Toric code is immune to sufficiently "uncorrelated" weak noise [52], but can be disturbed adversarially with very small operators (it has distance $\sqrt{n}$).

3.1.2 The Difficulty in Studying Quantum LDPC

Since researchers have been working on this problem for some time, it is natural to try and quantify why the problem is so difficult. Many of the early results for classical LDPC codes were found by sampling the local codes randomly, and examining the *expected* properties under certain random ensembles [72]. For quantum codes, it is not clear *how* to sample random locally generated codes, and naive methods quickly become intractable. Suppose we are interested in sampling uniformly a classical code $[n, k, \sim, w]$ with $w = 4$. We can simply sample the first generator of the dual (take a random word of weight 4), then sample the second generator by sampling a random word of weight 4 not contained in the subspace of the previous generators, and so on. In each step, we have to find a small weight word not in

the span of the previous ones. While this sounds like a (NP) hard problem [19], if we choose a word randomly and check if it is in the span of the previous ones, we are very likely to succeed, or we can repeat until we do succeed. Alternatively, we could have simply sampled a number of words of weight 4 independently, and condition on finding a full rank set. Both of these methods produce uniform distributions over the space of locally generated codes. The only structural requirement here is that the new word not be in the span of the previous ones. We require no relations amongst pairs of words except that they are not equal.

The analogous quantum problem is very different because we have additional structural requirements for pairs of words that we sample (recall quantum codes must be self-dual under some specific inner product). We can sample the first stabilizer just by choosing a random Pauli operator of weight 4 (choose the positions randomly and then choose from the set of Pauli matrices uniformly for each position). For the second generator, we can determine the space of Pauli operators that commute with the first and sample from that space, however, we also need to meet locality. In each step after the first, we need to solve a problem much like finding a small weight word in a code, which is known to be intractable [19, 94]. If we attempt to sample local stabilizers and check as before, we are unlikely to find one which commutes with all the previous stabilizers. Similarly, if we sample a number of Pauli operators of weight 4, we are very unlikely to find a commuting set. From here, some of the difficulty in studying quantum LDPC should be obvious. A theorist does not have access to uniform random instances of local quantum codes. [1]

[1] We could use random classical codes in classical to quantum constructions [44, 158], but this would not produce a *uniform* distribution over local codes, and in many cases will necessarily have distance $O(\sqrt{n})$

3.1.3 Surface Codes

There are many constructions of quantum LDPC codes derived from tilings of surfaces [24, 49, 102]. In these works one starts with a surface or manifold and partitions it into faces edges and points according to some particular lattice. The Toric code, for instance, is derived from a "square" tiling of the Torus:

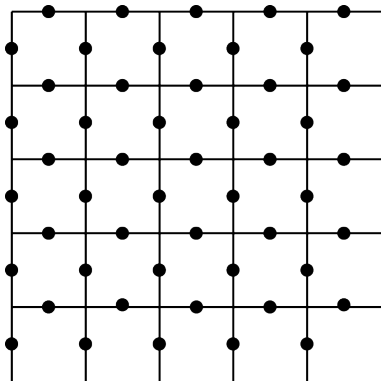


Figure 3.1: Square Tiling of the Torus

The qubits can be thought of as points on the edges. The generators for the stabilizer code are generated from characteristic vector of plaquettes and line intersections:

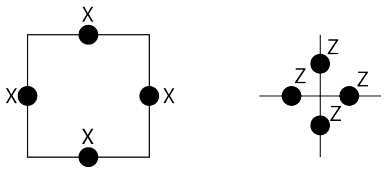


Figure 3.2: Generators of Toric Code

It is a CSS code where each of C_X, C_Z corresponds to the set of closed "topologically trivial" boundary regions. Any X stabilizer supported on the lattice is some region which can be "deformed" to the identity via the stabilizers. The smallest logical operator of the stabilizer code corresponds to the smallest topologically nontrivial vector:

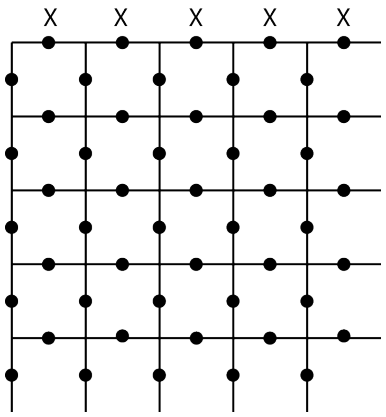


Figure 3.3: Smallest Logical operator of Toric Code

There are also different ways of Tiling that yield different codes. Color codes [24], for instance can be constructed by tiling the Torus with a 3-valent, 3 colorable lattice:

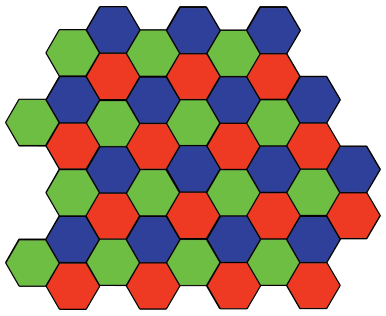


Figure 3.4: Lattice for Color Code

This is a lattice with faces that can each be assigned a color in such a way that no two faces of the same color are adjacent, and that three edges meet any vertex. The hexagonal lattice above is the typical example of a lattice which satisfies these properties. For this code, the qubits live on the vertices of the lattice and not on the edges. It is again a CSS code, but the X and Z generators are the same. Given any face of the lattice, we have a Z or an X stabilizer associated with the vertices on that face:

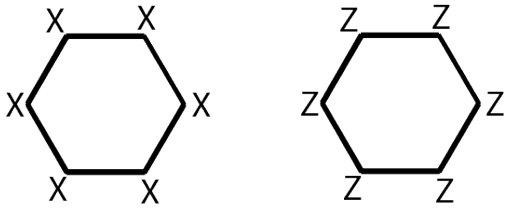


Figure 3.5: Generators of Color Code

Just as the Toric code, this code also has distance scaling with the square root of the number of qubits: The smallest logical operator is again the smallest topologically nontrivial operator which wraps around the Torus. Note that both of these constructions also generalize to different boundary conditions [23], but that these alterations still give distance an most $O(\sqrt{n})$.

3.1.4 What "kinds" of LDPC codes with distance $\Omega(n)$ could exist?

The codes we have described are regular tilings of some finite dimensional surface that unfortunately only achieve distance scaling with the square root of the block length. It turns out that these codes must have been "bad" because they are derived from $O(1)$ dimensional surfaces. A seminal paper in this respect by Bravyi and Terhal [30] characterizes the set of all low-dimensional grids as one ensemble where one cannot hope to find good quantum LDPC codes: quantitatively, a D-dimensional grid of n qubits in which the local checks of the code are spatially-local according to this grid have a minimal distance scaling as $O(n^{1-1/D})$. This result implies that for regular grids locality and minimal distance are adversely coupled - increasing the distance requires increasing the locality - so one cannot hope to achieve both linear distance and $O_n(1)$ locality simultaneously. We will describe this paper in detail here, since it is important for understanding our motivations.

Lemma 3.2 ([30] "Cleaning Lemma"). *Let $\mathcal{S} = \langle S_1, S_2, ..., S_m \rangle$ be the stabilizer for some $[[n, k, d, w]]$ quantum code with set of qubits V, and fix some subset of the qubits M. Then, exactly one of the following holds:*

1. *There is a non trivial logical operator supported only on M.*
2. *For any logical operator P, one can choose a stabilizer $S \in \mathcal{S}$ such that PS is supported only on $V \setminus M$ and S consists only of stabilizer generators that overlap with M.*

The result is intuitive, if we are in case 1, then M must correspond to a correctable erasure. If a logical operator cannot be localized to $V \setminus M$, then it cannot be measured and the erasure must not have been correctable. The lemma is used to clean out portions of the Lattice and provide an upper bound on the minimum distance of the "lattice code". Let us say a code $\mathcal{C}$ is embeddable into a D dimensional lattice if we can associate the qubits to vertices in the lattice in such a way that the stabilizer generators $\mathcal{S} = \langle S_1, ..., S_m \rangle$ are supported only on one hypercube of the lattice.

Theorem 3.3 ([30]). *Let $\{\mathcal{C}_n\}$ be an infinite family of quantum stabilizer codes with increasing block length such that each code $\mathcal{C}_n \in [[n, k, d_{min}, w]]$ is embeddable into a D dimensional lattice. Then,*

$$d_{min} = O(n^{1-1/D}) \tag{3.1}$$

Proof. (sketch) We will prove the result for 2 dimensional lattices, the higher dimensional case will be obvious from $D = 2$. Suppose the lattice has side length L, so there are $L^2 = n$ qubits, and suppose that the distance of the quantum code is at least $c\sqrt{n}$ for some $c >> 1$. Let us break up the lattice into an even number of vertical strips such that the width of each

strip is at most $c/2$, but still much larger than 1 (the size of a hypercube). This should be possible for sufficiently large L. Further, let us fix some nontrivial logical operator on the lattice.

Each vertical strip is correctable, hence we can "clean it out" with Lemma 3.2 using only stabilizers which overlap with that strip:

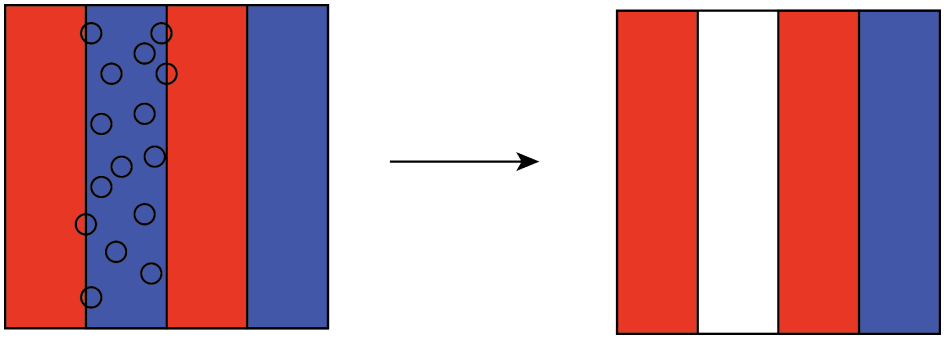


Figure 3.6: First step of Proof

Observe that the stabilizers we use to clean out any particular strip are located on that strip, and potentially adjacent strips, but not on any other strips. It follows that we can repeatedly apply the cleaning lemma to every other strip to obtain a logical operator that is supported only on odd strips:

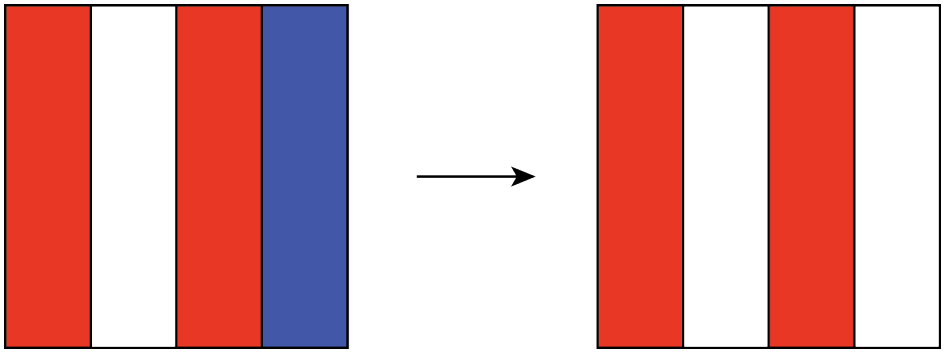


Figure 3.7: Second Step of Proof

Now observe that there must be at least one colored strip that corresponds to a nontrivial

logical operator. Since the stabilizers that overlap with one strip are disjoint from the stabilizers that overlap with another strip, it must be that each strip commutes with all the stabilizers. At least one of these strips must be a logical operator, otherwise the composite operator is a stabilizer. We obtain a logical operator of size less than $(c/2)\sqrt{n}$, and reach a contradiction. d_{min} must be upper bounded by $c\sqrt{n}$ for some sufficiently large constant c.

□

While the above proof was given for $(D = 2)$-lattices, it is easy to see how to generalize it. In higher dimensions we can break the lattice up into "slices" where one dimension of the slice is a constant. Every other slice can then be cleaned out in the same way to provide an upper bound on the distance.

One should think about the above results as demonstrating that certain *topologies* cannot support quantum LDPC codes. In particular, codes which can be embedded into lattices cannot possibly have linear distance. It makes sense, then, to examine the "other end" of topologies, or codes which are fundamentally unembeddable into finite dimensional lattices.

Expander graphs [114,137], and hypergraphs [113,129], are a well studied class of regular graphs that satisfy this criteria. There are many equivalent [21,96] ways to define expansion, for this chapter we will focus on (hyper) graphs which satisfy an "expander mixing lemma" (EML) [5][2]. For a graph $G = (V, E)$ and subsets $S, T \subseteq V$ define:

$$E(S,T) = |\{(i,j) \in E : i \in S \text{ and } j \in T\}| \tag{3.2}$$

as the number of edges between the subsets S and T. Define:

[2]The Expander Mixing Lemma implies that graphs with good spectral expansion satisfy this definition. We are using the outcome of the lemma as a definition

Definition 3.4 (ε-EML)**.** *Let $\{G_n\}_{n=1}^{\infty}$ be some infinite family of d regular graphs where each G_n is on n vertices. The family is an ε-EML family if*

$$\forall S, T \quad \left| \frac{E(S,T)}{|E|} - \frac{2|S||T|}{n^2} \right| < \varepsilon$$

for sufficiently large n.

The simple way to think about the above definition is that "random edge sampling from the graph very nearly matches random graph sampling from the complete graph". Suppose we fix and subsets S, T and sample a random edge from the complete graph and a random edge from E. The probability that the edge overlaps S and T are nearly the same for both, up to statistical distance ε.

To see how the above implies the graph cannot be embedded into a lattice imagine the opposite, namely that a graph satisfying the above condition can be embedded into a lattice. Choose S to be a block of some linear size, T to be its complement and sample an edge randomly. The probability that edge lies between S and T is proportional to the surface area of the box S, not it's volume as the expander mixing lemma implies. Note that if the dimension is D, then the surface area is $O(n^{D-1})$ while the volume is $\Omega(n^D)$ (Figure 3.8).

3.1.5 Quantum LDPC Codes Corresponding to Expanding (hyper) Graphs

We are interested in studying the parameters of locally generated quantum codes given that they correspond to some expanding classical object. To fix some structure, we will only be studying CSS codes where one of the classical codes, say C_X, corresponds to an expanding

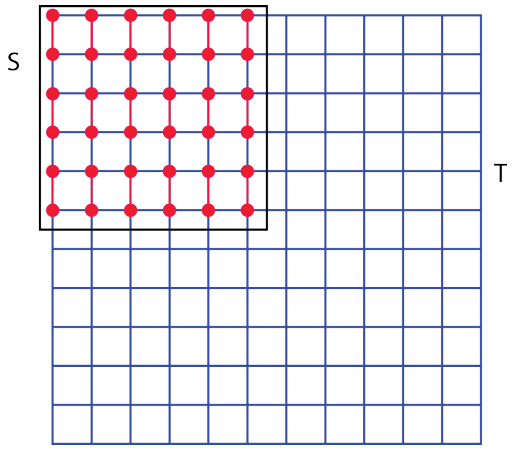


Figure 3.8: EML codes cannot be embedded into Lattices

hypergraph under some appropriate definition of expansion. We will assume that the codes C_X and C_Z are K-regular d-uniform. This means that each one is generated by words of weight d such that each bit is incident with K words of weight d. We will make the additional assumption that C_X has a larger rate than C_Z.

One natural direction is to assume that the code C_X satisfies Definition 3.4. The problem with this is that if $d = 2$ we already know the code will be locally entangled [2] and a code satisfying EML is necessarily over specified and must have small dual:

Lemma 3.5. *Let C be generated be generated by words of weight 2 such that these generators correspond to an ε-EML graph. Then, if $|C^\perp| \geq 4$ then*

$$\min_{\mathbf{c}^\perp \in C^\perp} |\mathbf{c}^\perp| \leq 2\varepsilon n \tag{3.3}$$

Proof. Let S be some set of bits corresponding to a word $\mathbf{w}$ in $C^\perp$. It must be, then, that $E(S, V \setminus S) = 0$ so

$$\left| \frac{2|\mathbf{w}|(n - |\mathbf{w}|)}{n^2} \right| < \varepsilon \tag{3.4}$$

This condition implies that $|\mathbf{w}| < \varepsilon n$ or $|\mathbf{w}| \geq (1 - \varepsilon)n$. If the code $C^\perp$ contains at least 4

words then at least one of them has weight less than $2\varepsilon n$.

□

A CSS code of this form would automatically have either small rate or small distance. If $C_X = C$, then C_Z either contains all of the low weight words of $C_X^{\perp}$ (small quantum rate) or it does not (small quantum distance). This definition is unsuitable because it implies the dual of the code is also "strongly" unembedable into a lattice. Since we are interested in studying stabilizer groups that cannot be embedded, it is natural to alter the definition to constrain only the *span* of the code C. Further, we need to consider the case of hypergraphs with $d \geq 4$ [2]. We obtain our working definition by considering EML with these two alterations. First, given some subset $S \subseteq V$ and some set of hyperedges E where each edge contains d vertices, define:

$$A(S, j) = \left| \{ e \in E : \ |e \cap S| = j \} \right|$$

Definition 3.6 (Type-1-ε-pseudorandom hypergraph)**.** *Let $d, K = O(1)$ be integers and $\mathcal{G} = \{G_n\}_n$, $G_n = (V_n, E_n)$ be a family of d-uniform hypergraphs where each vertex is incident on K hyperedges. Let* $\mathrm{span}(E_n)$ *denote the linear subspace of $\mathbb{F}_2^n$ spanned by the check terms E_n as vectors of $\mathbb{F}_2^n$. G_n is said to be Type-1-ε-pseudorandom if:*

$$\forall j \in [0, ..., d], \mathbf{w} \in \mathrm{span}(E_n) \quad \left| \frac{A(supp(\mathbf{w}), j)}{|E_n|} - \binom{d}{j} \left(\frac{i}{n}\right)^j \left(1 - \frac{i}{n}\right)^{d-j} \right| \leq \varepsilon \quad (3.5)$$

and family $\mathcal{G}$ is said to be Type-1-ε-pseudorandom if the above holds for all sufficiently large n.

We will also consider a slightly weaker definition:

Definition 3.7 (Type-2-ε-pseudorandom hypergraph)**.** *Let $d, K = O(1)$ be integers and $\mathcal{G} = \{G_n\}_n$, $G_n = (V_n, E_n)$ be a family of d-uniform hypergraphs where each vertex is incident on K hyperedges. Let* $\mathrm{span}(E_n)$ *denote the linear subspace of $\mathbb{F}_2^n$ spanned by the check terms E_n as vectors of $\mathbb{F}_2^n$. G_n is said to be Type-2-ε-pseudorandom if:*

$$\forall j \in [0, ..., d], i \in [0, ..., n] \quad \left| \mathbb{E}_{\substack{\mathbf{w} \in span(E_n) \\ |w| = i}} \left[\frac{A(supp(\mathbf{w}), j)}{|E_n|} \right] - \binom{d}{j} \left(\frac{i}{n}\right)^j \left(1 - \frac{i}{n}\right)^{d-j} \right| \leq \varepsilon \tag{3.6}$$

and family $\mathcal{G}$ is said to be Type-2-ε-pseudorandom if the above holds for all sufficiently large n.

It is clear that Type-1 pseudorandomness matches up with our informal notion of "unembeddability". We can use the same kind of informal argument we used for the EML definition. Fix K to be odd. Construct a word by choosing a point, adding all edges adjacent to it, then choosing another point on the exterior and repeating, etc. At some point the word will have at least $\Omega(n)$ many points and correspond to some body in the embedded space. A random check will overlap with probability proportional to the surface area, and not the volume. Assuming that the smallest simplex is only a constant factor smaller than the largest simplex (i.e. it can be embedded with low distortion) the resulting shape must have surface area $O(n^{D-1})$ and volume $\Omega(n^D)$. Our pseudorandom condition implies a "volume law" for random overlap with a closed region rather than an "area law" that we would get from an embeddable code. It is easy to see that regular tilings (as in the Toric or color codes) will not satisfy our definition by this argument (Figure 3.9).

The second definition does not have as clear a relation to expansion, since we cannot

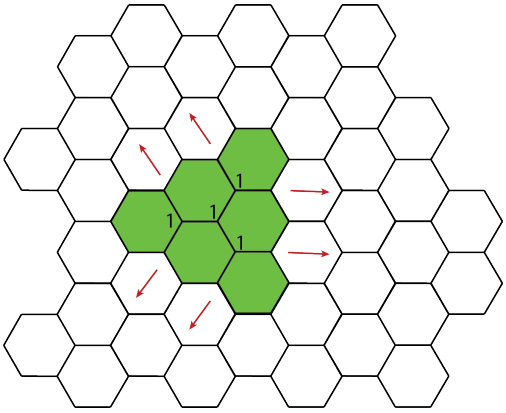


Figure 3.9: Type-1-pseudorandom hypergraphs cannot be embedded into regular lattices, since we can choose a point and move outward to find a large connected region. Random check sampling will satisfy an area law, rather than a volume law for this shape.

apply the definition to a region of our choice. Instead we must ask about the overlap of a generator with a randomly sampled word. However, it does seem to be a very practical definition appropriate to some notion of "structurelessness". Also, it should be easier to check that a given construction satisfies Type-2 rather than Type-1, since it requires only checking expansion "on average" rather than finding the worst-case set. Indeed, determining the Cheeger constant of a regular graph is NP-hard [97]. All the results we present hold equally well for either definition, the only change needed is Proposition 3.32 which holds in both cases. Since they both work, we will refer to a code which is Type-1 or Type-2 simply as as ε-Pseudorandom or ε-PR code.

3.1.6 Interesting Value of ε

There is a limit on the expansion of asymptotically large graphs. The Alon-Boppana theorem [124] combined with the Expander mixing lemma converse [21] assert that when $d = 2$ the best possible ε is $\Omega(1/\sqrt{K})$. Assuming generalizations of these theorems to the hypergraph

case, it is natural to expect a similar bound here[3]. It is not hard to see that for Type-1 pseudorandomness we have a lower bound $\varepsilon = \Omega(1/K)$, but this bound stems from technical reasons and not fundamental ones. A d-uniform K-regular code can be seen as K separate partitions of the bits into sets of size d. Suppose we choose our word $\mathbf{w}$ to be the sum of half of the sets from one partition, and we sample a random check $\mathbf{e} \in E$. If $\mathbf{e}$ comes from the same partition that $\mathbf{w}$ is from, we obtain a very 'atypical' distribution of overlap. Either the check is fully overlapping with the word $\mathbf{w}$ or it is disjoint. We can condition on our check coming from that partition. Since the check comes from each partition with equal probability, we obtain the 'atypical' distribution with probability $\approx 1/K$. The 'resolution' of our Type-1 condition is about $1/K$.

Note that Type-2 pseudorandomness has no clear lower bound as a function of d and K. It is also not reasonable to conjecture anything here since the corresponding statement for graphs seems to be poorly studied [4]. Any upper bound on Type-1 implies the same upper bound on Type-2, but it is conceivable that a hypergraph which is ε-Type-2 and ε'-Type-1 has $\varepsilon << \varepsilon'$.

3.1.7 Statement of Results

We obtain several bounds on the parameters of classical and quantum codes satisfying our pseudorandom conditions. First, we obtain a bound on the rate of regular classical codes that are ε-pseudorandom[5].

[3] There are known generalizations of these theorems [129], but the definition in these works differs from ours.

[4] I could not find *any* papers which discuss the expansion of randomly chosen subsets of expander graphs.

[5] While the machinery behind the proof is from our own techniques, the actual proof was suggested by an anonymous referee

Theorem 3.8. *Let $\{C_n\}$ be some family of classical codes that are d-sparse and ε-PR. Define the functions $\eta(\varepsilon) := h(3^{1/d}\varepsilon^{1/(2d)})$ and $\zeta(\varepsilon) := 3\varepsilon^{1/2}$. The rate m/n satisfies:*

$$1 - (d-1)\,\eta(\varepsilon) - \zeta(\varepsilon) - o(1) \leq \frac{m}{n} \tag{3.7}$$

Clearly this theorem applies to quantum codes when interpreted correctly. We are able to derive some additional bounds specifically for quantum CSS codes that are "one-sided" pseudorandom:

Theorem 3.9 (Main Theorem)**.** *Let $d, K = O(1)$ be integers, and let $\theta = 0.04$, $\varepsilon_0 = \frac{\theta^{2d}}{9}$ and $0 < \varepsilon < \varepsilon_0$ and $d \geq 4$. Let $\left\{C_X^{(n)}, C_Z^{(n)}\right\}_{n\in\mathbb{N}}$ be a family of quantum CSS codes with a d-sparse generating set, and in which each bit is incident on K generators of $C_Z^{(n)}$ and $C_X^{(n)}$. Let m_X be the rate of C_X seen as a linear code, let G_n be the hypergraph corresponding to the minimal weight words of $C_X^{(n)}$, and suppose $\mathcal{G} = \{G_n\}_n$ is ε-pseudorandom. Define the functions $\eta(\varepsilon) := h(3^{1/d}\varepsilon^{1/(2d)})$ and $\zeta(\varepsilon) := 3\varepsilon^{1/2}$. Then*

$$1 - (d-1)\eta(\varepsilon) - \zeta(\varepsilon) - o(1) \leq \frac{m_X}{n} \leq \eta(\varepsilon) + h(\eta(\varepsilon))$$

$$d_{\min} \leq 2h(h(\eta(\varepsilon)))n \leq \frac{2\cdot 3^{1/d}\varepsilon^{1/(2d)}}{d^2} \log\left(\frac{1}{\varepsilon}\right)^2 n.$$

The proof of the above crucially uses the "quantumness" of the corresponding classical codes. More specifically, the result applies to linear classical codes whose dual contains a large rate code (as in the CSS construction). Indeed one of the important contributions of this work is to develop techniques for bounding parameters of quantum codes using some standard classical coding tricks [6].

[6] Of course, classical techniques are frequently applied to quantum codes [10]. To my knowledge, this

3.1.8 Prior work

Recalling the result of Bravyi and Terhal [30] that places a sublinear limit on the distance of quantum codes whose underlying topology is a regular grid, our result can be viewed as the "other end" of this limit. We note, however, that our bounds can only show a small constant linear distance upper bound, and not a sublinear distance bound.

Interestingly, Hastings [86] recently conjectured the existence of a quantum CSS code with distance $n^{1-\varepsilon}$, for arbitrarily small $\varepsilon > 0$, whose parity check matrices have sparsity $\log(n)$. These codes are constructed from a cellulation of a family of random lattices, called LDA. Given the inherent embedding in a lattice, we conjecture that the 2-complex of his code would be somewhat far from pseudorandom. This suggests that perhaps the "right" way to resolve the qLDPC conjecture is to look for high-dimensional manifolds that avoid pseudorandomness.

Perhaps more fundamentally, our work suggests that quantum multi-particle entanglement may be inherently limited on random-looking topologies, contrasting the intuition stemming from the classical theory of computer science that random-looking topologies are more "robust". The notion that highly pseudorandom topologies are not compatible with large-scale quantum entanglement resonates with a sequence of results of a somewhat different context: In [25] the authors show that the ground states of 2-local Hamiltonians whose graph is *expanding* can be approximated by tensor-product states. A similar result in [3] shows this for k-local commuting Hamiltonians with a bipartite form of expansion, using a more stringent criterion called *local expansion*. These results impose a restriction

is the first work to use classical coding theory on locally generated CSS codes to obtain bounds on their parameters.

on the structure on the quantum system (namely 2-locality, or local expansion) and derive not only a minimal distance upper bound for a corresponding quantum code, but in fact an upper bound on the locality of entanglement. Our definition is more general, but only places a cap on the formal quantum minimal error-correcting distance without ruling out global entanglement altogether.

Our result is hence also relevant in the context of high-dimensional expanders: Evra and Kaufman recently showed [66] that the $d-1$ skeletons of Ramanujan d-complexes are so-called "co-systole" expanders. Since there exists a natural map from $\mathbb{F}_2$-complexes to CSS codes our main theorem can be used to place a lower bound on the pseudo-randomness of such skeletons. Hence, it is an example where the quantum perspective may help shed light on questions in other fields.

3.1.9 Overview of the proof

Placing bounds on the minimal quantum error-correcting distance of LDPC codes is usually a difficult task [15, 34, 71], and these techniques are generally not useful in the quantum regime. Even in the context of CSS codes, which are arguably, the most "classical" quantum codes we know of, arguing about the minimal quantum error-correcting distance turns out to be a challenging task since it is not merely a property of the dual subspace of an $\mathbb{F}_2$ vector space as in classical error correction, but rather a property of the quotient of subspaces which is a more complex object to analyze.

The strategy we use in this chapter, that will be shortly described in more detail, is to in fact find a way to employ classical techniques to study CSS codes - namely Kravchuk

polynomials, and the MacWilliams identities. [7]

We begin by defining a linear space $C \subseteq \mathbb{F}_2^n$ to be *weakly-binomial* if its weight enumerator $(B_0, \ldots, B_n)$ - i.e. the vector of $n+1$ bins, specifying the number of words of C in any given weight - is upper bounded by the binomial distribution up to a mild exponential factor:

Definition 3.10 (Weakly-binomial subspace)**.** *A subspace $C \subseteq \mathbb{F}_2^n$ on n bits is* (ζ, η)-weakly-binomial *if for some constants $\zeta > 0$ and $\eta > 0$ we have:*

$$\forall k \in \{0, \ldots, n\}, \quad B_k \leq \frac{2^{\zeta n}\binom{n}{k}}{|C^\perp|} + 2^{\eta n}, \tag{3.8}$$

where $\{B_k\}$ is the weight enumerator of C, $C^\perp$ is the dual space of C.

Our proof consists of two main steps:

1. We show that the subspace $C \subseteq \mathbb{F}_2^n$ spanned by the generators (hyperedges) of an ε-pseudorandom hypergraph is weakly-binomial.

2. We show that any weakly-binomial subspace C for which, in addition, the dual code $C^\perp$ also *contains* a LDPC code, must have a relatively small dimension.

Hence for a CSS code $\mathcal{C} = (C_X, C_Z)$ with C_X corresponding to an ε-PR hypergraph, the relative dimension of C_X must be small, which implies the quantum code dimension $k = n - \dim(C_X) - \dim(C_Z)$ must be large. By standard distance-rate trade-offs in quantum error-correction this then implies an upper bound on the minimal quantum error-correcting distance of $\mathcal{C}$.

[7] There are quantum analogs of the MacWilliams identities [148] [135] but these are morally different than their classical counterparts, and do not crucially exploit the structure of the CSS code as a pair of weakly self-dual *classical* codes. Ashikhmin and Litsyn [10] study the parameters of quantum codes using the MacWilliams transform over $\mathbb{F}_4$, our techniques involve invoking MacWilliams on the binary code of one side of the CSS code (say the X checks).

An ε-PR hypergraph spans a weakly-binomial subspace

In the first step, we would like to approximate the weight enumerator of a space C_X spanned by a set of generators that satisfy ε-PR. The weight enumerator is approximated by considering a random walk $\mathcal{M}^1$ on the Cayley graph of the space C_X using its set of LDPC-sparse, and ε-pseudorandom set of generators. The stationary distribution of $\mathcal{M}^1$, when summed-up over separate shells of $\mathbb{F}_2^n$ of fixed-weight then provide exactly the weight enumerator.

We would hence like to "project"-down $\mathcal{M}^1$ to a random walk $\mathcal{M}^2$ defined on $n+1$ nodes corresponding to "shells" of fixed weight in $\mathbb{F}_2^n$. Hence, we would like to define transition probabilities between "weight-bins" that are independent of which word we choose in a fixed-weight bin. To do this, we define a coarse-graining of the chain over some fixed partition of the outcome space. We choose the shells of fixed weight as the sets in our partition.

So now, consider the line walk $\mathcal{M}^2$: it is comprised of $n+1$ bins, with nonzero transition probabilities between nodes of distance at most q - the locality of the generators. We consider a bin B_k - i.e. the set of words in C_X of weight k, and ask: suppose that we sample a uniformly random generator $\mathbf{g}$ from C_X and add it to a random $\mathbf{w} \in B_k$: what is the distribution of $|\mathbf{w}+\mathbf{g}|$?

Generically - this might be a hard problem to solve. However, using the ε-PR condition it becomes simpler: this condition, when interpreted in the appropriate way, tells us that the probability that $|\mathbf{w}+\mathbf{g}| = k+j$, where $j \leq q$, and q is the locality of each generator - behaves approximately like sampling a word of weight q *uniformly at random* and adding it to $\mathbf{w}$. As a result $\mathcal{M}^2$ assumes the form of a Markov chain whose transition probabilities are governed by the binomial distribution, i.e. adding uniformly random words of weight q,

up to an *additive* error at most ε. We then analyze this chain, and show it implies that the stationary distribution of this perturbed chain deviates from the pure unperturbed chain by a modest multiplicative exponential error, so long as the bins we consider are "close" to the center $n/2$. Far from the center bin $B_{n/2}$ we have no control - but this is translated to a small exponential additive error - which together implies that C_X is a weak binomial space.

Weakly-binomial subspaces with LDPC duals have large dimension

In this part of the proof we are given a subspace C_X, such that $C_X^\perp$ contains an LDPC code, namely C_Z, and C_X is weakly binomial. We think of C_X as being the span of parity checks C_X of a CSS code, and hence $C_X^\perp$ *contains* also the space spanned by a set of LDPC parity checks C_Z. By the assumption of the theorem the hypergraph of C_Z is d-uniform and the degree of each vertex is $K = O(1)$.

To place an upper bound on the dimension of C_X we invoke the Sloane MacWilliams transform [117] which translates any weight enumerator on a code C_X, to the weight enumerator on the dual code $C_X^\perp$. The *crux* of the argument is essentially a weak converse to previous results Litsyn et al. [108] which use the transform in the context of classical codes.

Consider a classical code of large distance. It's weight enumerator $(B_0, \ldots, B_n)$ is by definition such that $B_0 = 1$ (the zero word) and $B_i = 0$ for all $0 < i \leq \delta_{\min}$. The result by Litsyn shows that if this is the case, then the weight enumerator of the dual code $(B_0^\perp, \ldots, B_n^\perp)$ has an upper bound that is very close to being binomial: if one considers an interval of linear size around $n/2$, say $[n/3, \ldots, n/2, \ldots, 2n/3]$ then it is the case that each B_k is at most $\binom{n}{k}/|C_X|$ up to a multiplicative polynomial factor.

In our case, we consider a *quantum* code, namely a CSS code, and argue the opposite

way: we show that if the weight enumerator (of one of the corresponding classical codes) $(B_0, \ldots, B_n)$ is *weakly* binomial in the sense defined above, then the weight enumerator of the dual $B^\perp := (B_0^\perp, \ldots, B_n^\perp)$ does not have a precise prefix of 0 bins, as in the classical case of a large-distance code, but the first αn bins are still very small, for some constant $\alpha > 0$.

On the other hand, and this happens only for weakly self-dual codes, as in the case of quantum CSS codes: we know that the dual code $C_X^\perp$ contains an LDPC code. Any LDPC code - whether it is ε-PR or not, has the property that in the appropriate scale, the number of words in bin B_k grows exponentially fast with k, at least for sufficiently small $k = \beta n$.

So together, we collide on the bins of the dual code the opposing forces of the upper bound implied by the weak binomial distribution, which implies that the lower-prefix of $B^\perp$ is very small, with the fact that this lower-prefix blows-up exponentially fast because it contains an LDPC code. This implies a stringent limit on the dimension of the parity check spaces - and hence a lower bound on the rate of the code, which in turn implies a stringent upper bound on the minimal distance of its corresponding quantum code.

3.2 Preliminaries

3.2.1 Notation

We adopt the following conventions and notation throughout the chapter. Even though some of the machinery we use (e.g. MacWilliams identity) hold in a more general setting, we restrict our attention to binary linear codes in the classical case and binary (qubit) codes based on the CSS construction in the quantum case. Consequently, all linear operators in

this chapter are over $\mathbb{F}_2$. We will write $\rho := k/n$ for the *rate* of a code, classical or quantum, and $\delta_{\min} := d_{\min}/n$ for the *relative distance* of a code.

We will use d to denote the largest size of a hyperedge in a complex (not to be confused with the distance $d_{\min}$). The letter K will be used to denote the number of hyperedges incident on a given vertex.

Definition 3.11. *Let $g(n)$ and $h(n)$ be two functions of n. We write $g(n) \geq_p h(n)$ if, for all $n \geq 1$,*

$$g(n) \geq h(n)n^z \tag{3.9}$$

for some constant z. Similarly, we write $g(n) \leq_p h(n)$ if, for all $n \geq 1$,

$$g(n) \leq h(n)n^z. \tag{3.10}$$

We will reserve ε for the pseudorandom parameter defined in the introduction, we reserve θ for the constant 0.04, and we reserve $\varepsilon_0 = \frac{\theta^{2d}}{9}$ where d is the regularity of the hypergraph. **We will assume throughout that $0 < \varepsilon < \varepsilon_0$ defined in the main theorem, and $d \geq 4$.**[8]

3.2.2 Technical Facts Needed for the Results

We will need many technical lemmas to establish the results in this chapter. While the expression below looks strange, it will be important in our analysis.

Fact 3.12. *Let*

$$f_{d,\beta}(\gamma) := 2\gamma + (1-2\gamma)h\left(\frac{\beta-\gamma}{1-2\gamma}\right) + \frac{1}{d}h\left(2\gamma\right) - h(\beta)$$

[8] We restrict our attention to $d \geq 4$ since, otherwise, the code is locally entangled [2].

and suppose that $\beta \leq 2^{-2}$. *Then*

$$f_{d,\beta}(\beta^d) \geq \beta^d.$$

Proof. We obtain, based on lemma 3.13, :

$$h\left(\frac{\beta - \beta^d}{1 - 2\beta^d}\right) \geq h(\beta - \beta^d) \geq h(\beta) + \beta^d \cdot \frac{5}{4}\log(\beta).$$

Clearly it also holds that:

$$h\left(\frac{\beta - \beta^d}{1 - 2\beta^d}\right) \leq 1 \tag{3.11}$$

So we obtain the lower bound:

$$(1 - 2\beta^d)h\left(\frac{\beta - \beta^d}{1 - 2\beta^d}\right) \geq h(\beta) + \beta^d\frac{5}{4}\log(\beta) - 2\beta^d \tag{3.12}$$

Applying this expression to $f_{d,\beta}(\beta^d)$, we obtain:

$$f_{d,\beta}(\beta^d) \geq \frac{5\beta^d}{4}\log(\beta) + \frac{1}{d}h(2\beta^d) \tag{3.13}$$

By our upper bound on β, it holds that $2\beta^d < 1/2$, so we can apply Proposition 2.21:

$$\begin{aligned} f_{d,\beta}(\beta^d) \geq \frac{5\beta^d}{4}\log(\beta) + \frac{2\beta^d}{d}\log\left(\frac{1}{2\beta^d}\right) &= -\frac{5\beta^d}{4}\log\left(\frac{1}{\beta}\right) + 2\beta^d\log\left(\frac{1}{\beta}\right) - \frac{2\beta^d}{d} \\ &= \frac{3\beta^d}{4}\log\left(\frac{1}{\beta}\right) - \frac{2\beta^d}{d} \end{aligned} \tag{3.14}$$

Now we apply our assumed bounds on β and d:

$$f_{d,\beta}(\beta^d) \geq \beta^d\left(\frac{3\log(2^2)}{4} - \frac{1}{2}\right) = \beta^d \tag{3.15}$$

□

We will refer to the following lemma at many points in the chapter. It provides some simple estimates of functions of interest, using some straight-forward calculus.

Lemma 3.13. *1. Let $\varepsilon < \varepsilon_0$ defined in the chapter. It holds that*

$$\frac{1+\varepsilon^{1/2}}{1-\varepsilon^{1/2}} \leq 1 + 3\varepsilon^{1/2} \tag{3.16}$$

2. For $\beta \leq \frac{1}{4}$, it holds that

$$h(\beta - \beta^d) \geq h(\beta) + \frac{5}{4}\beta^d \log(\beta) \tag{3.17}$$

3. Let $\varepsilon < \varepsilon_0$ defined in the chapter. It holds that

$$\log(1 + 3\varepsilon^{1/2}) \leq 5\varepsilon^{1/2} \tag{3.18}$$

4. For all $\varepsilon < \varepsilon_0$ it holds that:

$$2 \cdot 3^{1/d} \varepsilon^{1/(2d)} \log \frac{1}{\varepsilon^{1/(2d)}} \leq \frac{1}{2} \tag{3.19}$$

Proof. **Proof of Item 1** Taylor expand:

$$\frac{1}{1-\varepsilon^{1/2}} = \sum_{j=0}^{\infty} \varepsilon^{j/2} = 1 + \varepsilon^{1/2} + \varepsilon \left(\frac{1}{1-\varepsilon^{1/2}} \right) \leq 1 + \varepsilon^{1/2} + \varepsilon \left(\frac{1}{1-\varepsilon_0^{1/2}} \right) \tag{3.20}$$

We obtain:

$$\frac{1+\varepsilon^{1/2}}{1-\varepsilon^{1/2}} \leq 1 + 2\varepsilon^{1/2} + \left(\frac{\varepsilon}{1-\varepsilon_0^{1/2}} + \varepsilon + \frac{\varepsilon^{3/2}}{1-\varepsilon_0^{1/2}} \right) \tag{3.21}$$

To complete the proof observe that the third term can be upper bounded as:

$$\varepsilon^{1/2} \left(\frac{\varepsilon^{1/2}}{1-\varepsilon_0^{1/2}} + \varepsilon^{1/2} + \frac{\varepsilon}{1-\varepsilon_0^{1/2}} \right) \leq \varepsilon^{1/2} \left(\frac{\varepsilon_0^{1/2}}{1-\varepsilon_0^{1/2}} + \varepsilon_0^{1/2} + \frac{\varepsilon_0}{1-\varepsilon_0^{1/2}} \right) \tag{3.22}$$

The absolute upper bound we have on ε_0 is $\frac{\theta^{2d}}{9} \leq \frac{\theta^8}{9}$ (recall we are assuming $d \geq 4$). It is easy to check for this specific value of ε_0 that the term in parenthesis is smaller than 1.

Proof of Item 2 Apply the mean value theorem. Define the function:

$$f(x) := h(\beta - x) \tag{3.23}$$

and take the derivative to find:

$$\frac{df}{dx} = \log(\beta - x) - \log(1 - \beta + x) \tag{3.24}$$

By the mean value theorem, there exists a point $\zeta \in (0, \beta^d)$ such that:

$$\frac{df}{dx}(\zeta) = \frac{f(\beta^d) - h(\beta)}{\beta^d} \tag{3.25}$$

Rewriting and plugging in our expression for the derivative:

$$h(\beta - \beta^d) = \beta^d \left(\log(\beta - \zeta) - \log(1 - \beta + \zeta)\right) + h(\beta) \tag{3.26}$$

Elementary lower bounds imply:

$$h(\beta - \beta^d) \geq \beta^d \log(\beta - \beta^d) + h(\beta) \tag{3.27}$$

Now to finish the proof we need to show that:

$$\log(\beta - \beta^d) \geq \frac{5}{4} \log(\beta) \tag{3.28}$$

Taking powers of 2, this is equivalent to:

$$1 \geq \beta^{1/4} + \beta^{d-1} \tag{3.29}$$

Using our lower bound on d, this condition is implied by:

$$1 \geq \beta^{1/4} + \beta^3 \tag{3.30}$$

Once again it is easy to check that this holds for $1/4$, and we can observe that for decreasing β the RHS is decreasing. Hence it holds for all $0 < \beta < 1/4$.

Proof of Item 3 Taylor expand $f(x) = \log(1+x)$ around $x = 0$. The series is:

$$\sum_{k=1}^{\infty}(-1)^{k+1}\frac{x^k}{k\log_e(2)} \tag{3.31}$$

Observe that this is an alternating convergent Taylor series that is term by term decreasing in magnitude if x is smaller than 1. It follows that the function can be upper bounded by the first element of the Taylor series:

$$\log(1+3\varepsilon^{1/2}) \leq \frac{3\varepsilon^{1/2}}{\log_e(2)} \leq 5\varepsilon^{1/2} \tag{3.32}$$

Proof of Item 4 We use the same type of analysis. Define $f(x) := x\log(1/x)$. It is simple to show that $\frac{df}{dx} > 0$ for all $x \in (0,1)$. It follows that:

$$\begin{aligned} 2\cdot 3^{1/d}\varepsilon^{1/(2d)}\log\left(\frac{1}{3^{1/d}\varepsilon^{1/(2d)}}\right) &\leq 2\cdot 3^{1/d}\varepsilon^{1/(2d)}\log\left(\frac{1}{\varepsilon^{1/(2d)}}\right) \leq 2\cdot 3^{1/d}\varepsilon_0^{1/(2d)}\log\left(\frac{1}{\varepsilon_0^{1/(2d)}}\right) \\ &= 2\cdot 3^{1/d}\frac{\theta}{9^{1/(2d)}}\log\left(\frac{9^{1/(2d)}}{\theta}\right) \leq 2\cdot\theta\log\left(\frac{9^{1/(8)}}{\theta}\right) \leq \frac{1}{2} \end{aligned} \tag{3.33}$$

□

Regularity is useful in our bounds exactly because of the following proposition. Codes which are sparsely generated and regular must have small weight codewords. In particular, we must be able to tile the bits with generators from the code. At this level of "coarse-graining" we can lower bound the weight enumerator by taking generators that add up to the correct weight:

Proposition 3.14. *Let C be an m-dimensional linear code over $\mathbb{F}_2$ on n bits. Suppose each generator of the code has weight d, and that the degree of every vertex is $K = O(1)$. For k divisible by d and $k \leq n/d$ we have:*

$$B_k \geq \binom{\frac{n}{d}}{\frac{k}{d}} \tag{3.34}$$

where $\{B_k\}$ *is the weight enumerator of* C*.*

Proof. Any (d, K)-regular bi-partite graph can be written as a family of K partitions of $[n]$ into n/d parts. From any such partition the number of words of weight k is

$$\binom{n/d}{k/d}$$

which is a lower bound on B_k.

□

The following proposition will be useful for converting the bounds we derive in the quantum code rate to bounds on the quantum distance. Unlike the Hamming bond, which only holds for certain quantum codes, this bound applies to all $[[n, k, d_{min}]]$ codes. It's derivation falls from some standard entropy relations (see Gottesman's lecture notes from CO639 at the Perimeter institute).

Proposition 3.15 ([103] Quantum Singleton Bound)**.** *A quantum code with parameters* $[[n, k, d_{\min}]]$ *satisfies:*

$$\frac{k}{n} \leq 1 - 2\delta_{min} + o(1) \tag{3.35}$$

3.2.3 Some Classical Coding Theory

Kravchuk polynomials are a special set of orthogonal polynomials with many applications in error correction [117, p. 130]. They have a simple interpretation which makes their definition and many of their properties intuitive.

We fix n to be some positive integer throughout. Let $m \in \{0, \ldots, n\}$ and denote by

$$S_m := \{\mathbf{w} \in \mathbb{F}_2^n : |\mathbf{w}| = m\} \tag{3.36}$$

the set of all length-n strings of Hamming weight m. Let $\chi_{\mathbf{u}} : \mathbb{F}_2^n \to \{-1, +1\}$ be a character of $\mathbb{F}_2^n$ for some $\mathbf{u} \in \mathbb{F}_2^n$, i.e. a function of the form $\chi_{\mathbf{u}}(\mathbf{v}) := (-1)^{\mathbf{u}\cdot\mathbf{v}}$ where $\mathbf{u} \cdot \mathbf{v} := \sum_{i=1}^n u_i v_i$ denotes the inner product modulo 2. The m-th Kravchuk polynomial evaluated at $x \in \{0, \ldots, n\}$ is then defined as

$$P_m(x) := \sum_{\mathbf{w} \in S_m} \chi_{\mathbf{u}}(\mathbf{w}) = \sum_{\mathbf{w} \in S_m} (-1)^{\mathbf{u}\cdot\mathbf{w}}, \tag{3.37}$$

where $\mathbf{u} \in \mathbb{F}_2^n$ is any vector of Hamming weight $|\mathbf{u}| = x$. Note by symmetry that $P_m(x)$ does not depend on the word $\mathbf{u}$ chosen as long as $|\mathbf{u}| = x$. Also note that $P_m(x)$ implicitly depends also on the dimension n of the underlying space $\mathbb{F}_2^n$, which should be clear from the context.

For any integer $l \geq 0$ and formal variable x, we define the *binomial coefficient* as the following degree-l polynomial in x:

$$\binom{x}{l} := \frac{x(x-1)...(x-l+1)}{l!}. \tag{3.38}$$

For integers $l < 0$ the binomial coefficient is taken to be zero. Using this, Kravchuk polynomials can be written explicitly as follows:

Definition 3.16. *The m-th* Kravchuk polynomial, *for $m \in \{0, \ldots, n\}$, is a degree-m polynomial in $x \in \mathbb{R}$ given by*

$$P_m(x) := \sum_{l=0}^{m} (-1)^l \binom{x}{l} \binom{n-x}{m-l}. \tag{3.39}$$

It is not hard to see that equations 3.37,3.39 agree for integer values of x.

One of the most important properties of Kravchuk polynomials is that they are orthogonal under a particular inner product. This fact that can be easily verified using the above

interpretation:

Lemma 3.17 (Kravchuk Orthogonality [117])**.** *For $i, j \in \{0, \ldots, n\}$, the Kravchuk polynomials $P_i(k)$ and $P_j(k)$ satisfy*

$$\sum_{k=0}^{n} \binom{n}{k} P_i(k) P_j(k) = \delta_{ij} 2^n \binom{n}{i} \tag{3.40}$$

where δ_{ij} is the Kronecker delta.

One important application of this orthogonality relation is that any polynomial $g(x)$ with $\deg(g) \leq n$ has a unique Kravchuk decomposition. A simple way of determining such a decomposition is as follows:

Fact 3.18. *If $g(x)$ is a polynomial of degree at most n, its Kravchuk decomposition is*

$$g(x) = \sum_{j=0}^{n} g_j P_j(x) \tag{3.41}$$

where

$$g_j := \frac{1}{2^n \binom{n}{j}} \sum_{k=0}^{n} \binom{n}{k} P_j(k) g(k). \tag{3.42}$$

Following the line of argument in [108], we will make use of a particular decomposition for the polynomial $P_m(x)^2$:

Lemma 3.19. *[108] For any $m \in \{0, \ldots, \lfloor n/2 \rfloor\}$,*

$$(P_m(x))^2 = \sum_{i=0}^{m} \binom{2i}{i} \binom{n-2i}{m-i} P_{2i}(x). \tag{3.43}$$

Proof. According to equation 3.37,

$$(P_m(x))^2 = \sum_{\mathbf{w}, \mathbf{w}' \in S_m} (-1)^{\mathbf{u} \cdot (\mathbf{w} + \mathbf{w}')} \tag{3.44}$$

for any $\mathbf{u} \in \mathbb{F}_2^n$ such that $|\mathbf{u}| = x$. Note that $|\mathbf{w} + \mathbf{w}'| = 2i$ for some $i \in \{0, \ldots, m\}$, so we can rewrite the right-hand side of equation 3.44 as

$$\sum_{i=0}^{m} \sum_{\mathbf{v} \in S_{2i}} c_{\mathbf{v}} (-1)^{\mathbf{u} \cdot \mathbf{v}}, \tag{3.45}$$

where the integer $c_{\mathbf{v}}$ accounts for the number of ways two n-bit strings $\mathbf{w}$ and $\mathbf{w}'$ (each of Hamming weight m) can overlap to produce a given string $\mathbf{v} = \mathbf{w} + \mathbf{w}'$ of weight $|\mathbf{v}| = 2i$. It is not hard to see that $c_{\mathbf{v}}$ depends only on the Hamming weight of $\mathbf{v}$ and is given by

$$c_{\mathbf{v}} = \binom{2i}{i} \binom{n-2i}{m-i}. \tag{3.46}$$

Indeed, we simply need to account for all ways of splitting the $2i$ ones of $\mathbf{v}$ into two groups of size i each (one of the groups is contributed by $\mathbf{w}$ while the other by $\mathbf{w}'$) as well as picking $m - i$ out of the remaining $n - 2i$ locations where the remaining $m - i$ ones of $\mathbf{w}$ and $\mathbf{w}'$ would cancel out. □

We will also need the following simple upper bound on Kravchuk polynomials:

Lemma 3.20. *[117] For any $k \in \{0, \ldots, n\}$,*

$$P_m(k) \leq \binom{n}{m}. \tag{3.47}$$

Proof. This follows easily from equation 3.37. Let $\mathbf{u}$ be any binary vector with $|\mathbf{u}| = k$. Then

$$P_m(k) = \sum_{\mathbf{w} \in S_m} (-1)^{\mathbf{u} \cdot \mathbf{w}} \leq \sum_{\mathbf{w} \in S_m} (1)^{\mathbf{u} \cdot \mathbf{w}} = \binom{n}{m} \tag{3.48}$$

as claimed. □

We will use an important relation known as MacWilliams identity [117]. Suppose we have some linear code C in $\mathbb{F}_2^n$. We define the weight enumerator of the code C as a set of coefficients $\{B_k\}$ where each B_k denotes the number of words of weight k in the code. Of

course the code $C^{\perp}$ has its own weight enumerator $\{B_k^{\perp}\}$. We state these notions formally in the following definitions:

Definition 3.21. *Given a code C, and for all $k \in \{0, \ldots, n\}$, we define the weight enumerator B_k as:*

$$B_k = |\{\mathbf{x} \in C : |\mathbf{x}| = k\}| \tag{3.49}$$

Naturally the dual code has an analogously defined weight enumerator:

Definition 3.22. *For a code C and $C^{\perp}$, we define:*

$$B_k^{\perp} = |\{\mathbf{x} \in C^{\perp} : |\mathbf{x}| = k\}| \tag{3.50}$$

The MacWilliams identity provides a way to write the weight enumerator of the dual code in terms of the weight enumerator of C, and the Kravchuk polynomials.

Theorem 3.23 ([117]). *Let C be a linear code over $\mathbb{F}_2^n$ with weight enumerator $\{B_k\}$. Denote the dual code $C^{\perp}$ and its weight enumerator $\{B_k^{\perp}\}$. Then, it holds that:*

$$B_k^{\perp} = \frac{1}{|C|} \sum_{j=0}^{n} P_k(j) B_j \tag{3.51}$$

where $|C|$ denotes the number of codewords in C.

Lemma 3.24 ([108]). *Let $\{B_j\}$ be the weight enumerator of a code C on n bits, and $\{B_j^{\perp}\}$ be the weight enumerator of its dual. If $\alpha(x) := \sum_{j=0}^{n} \alpha_j P_j(x)$ for some coefficients α_j, then*

$$|C| \sum_{j=0}^{n} \alpha_j B_j^{\perp} = \sum_{j=0}^{n} \alpha(j) B_j. \tag{3.52}$$

Proof. By the MacWilliams identity,

$$\sum_{j=0}^{n} \alpha_j B_j^{\perp} = \sum_{j=0}^{n} \alpha_j \left[\frac{1}{|C|} \sum_{k=0}^{n} B_k P_j(k) \right] = \tag{3.53}$$
$$\frac{1}{|C|} \sum_{k=0}^{n} B_k \sum_{j=0}^{n} \alpha_j P_j(k) = \frac{1}{|C|} \sum_{j=0}^{n} B_j \alpha(j).$$

□

For obvious reasons, this identity has many applications in error correction. In particular it is a useful tool for establishing many interesting bounds on quantum codes [10].

3.2.4 Markov Chains

We will use coarse-grained Markov chains in our analysis. We wish to partition the discrete state space of a Markov chain and analyze the coarse-grained dynamics. Given a Markov chain $\mathcal{M}$ with state space Ω, and some subset $A \subseteq \Omega$, for any probability distribution π over Ω we will denote:

$$\pi_A := \sum_{i \in A} \pi_i \tag{3.54}$$

We then define a coarse-graining of a Markov chain:

Definition 3.25 (Coarse-grained Markov chain)**.** *Let $\mathcal{M}$ be an irreducible Markov chain with state space Ω. Denote the probability of transitioning from i to j as $\mathcal{M}_{i,j}$. Suppose we have a partition $\{S_k\}$ of Ω (i.e. $\cup_k S_k = \Omega$ and $S_k \cap S_j = \emptyset$ for $k \neq j$). Denote the Markov chain's stationary distribution by $\{\pi_j\}$. We denote the coarse-grained Markov chain with respect to $\{S_k\}$ and π by $\mathcal{M}'$. It has exactly one state for each set in $\{S_i\}$. If A and B are two sets in $\{S_i\}$ we define:*

$$\mathcal{M}'_{A,B} = \sum_{i \in A} \sum_{j \in B} \frac{\pi_i}{\pi_A} \mathcal{M}_{i,j}. \tag{3.55}$$

Lemma 3.26. *Let $\mathcal{M}$ be an irreducible Markov chain with stationary distribution $\{\pi_j\}$, and suppose we have some partition of the state space $\{S_i\}$. Suppose we construct the coarse-grained Markov chain $\mathcal{M}'$ with respect to the partition $\{S_i\}$. Denote the stationary distribution of $\mathcal{M}'$ as $\{\pi'_{S_i}\}$. The stationary distribution of $\mathcal{M}'$ satisfies:*

$$\forall S_i \in \{S_i\} \quad \pi'_{S_i} = \pi_{S_i} = \sum_{j \in S_i} \pi_j \tag{3.56}$$

Proof. Fix some $B \in \{S_i\}$, we can evaluate:

$$\sum_{S_i} \pi_{S_i} \mathcal{M}'_{S_i,B} = \sum_{S_i} \pi_{S_i} \sum_{j \in S_i} \sum_{k \in B} \frac{\pi_j}{\pi_{S_i}} \mathcal{M}_{j,k} \tag{3.57}$$

$$= \sum_{S_i} \sum_{j \in S_i} \sum_{k \in B} \pi_j \mathcal{M}_{j,k} = \sum_{j \in \Omega} \sum_{k \in B} \pi_j \mathcal{M}_{j,k} = \sum_{k \in B} \sum_{j \in \Omega} \pi_j \mathcal{M}_{j,k} \tag{3.58}$$

Since $\{\pi_j\}$ is stationary for the original chain, then

$$= \sum_{k \in B} \pi_k = \pi_B \tag{3.59}$$

So, the distribution $\{\pi_{S_i}\}$ is stationary for the coarse-grained chain $\mathcal{M}'$. □

Definition 3.27 (Reversible Markov chains)**.** *Let $\mathcal{M}$ be a Markov chain on space Ω with stationary distribution π. $\mathcal{M}$ is said to be reversible if*

$$\forall i, j \in \Omega, \quad \pi_i \mathcal{M}_{i,j} = \pi_j \mathcal{M}_{j,i}$$

The following fact is standard: it says that the random walk on the Cayley graph of a

finite group is reversible. This follows almost immediately from the fact that this walk is invariant under left multiplication by an element of the group:

Fact 3.28. *Let G be a group, and $\mathcal{G}(G, s)$ be the Cayley graph of G w.r.t. some generating set $s \subseteq G$. Let $\mathcal{M}$ denote the random walk on $\mathcal{G}$. Then $\mathcal{M}$ is reversible.*

Next, we show that under our natural definition of coarse-graining, reversible chains remain reversible:

Fact 3.29. *Let $\mathcal{M}$ be a reversible Markov chain on space Ω, with a stationary distribution π. Let $\{S_k\}$ be some partition of Ω. Then the coarse-grained chain $\mathcal{M}'(\{S_k\}, \pi)$ is reversible.*

Proof. Let $\mathcal{M}$ be a reversible Markov chain. By definition, then,

$$\pi_i \mathcal{M}_{i,j} = \pi_j \mathcal{M}_{j,i} \tag{3.60}$$

We can write:

$$\begin{aligned} \pi_{S_i} \mathcal{M}'_{S_i,S_j} &= \pi_{S_i} \sum_{k_1 \in S_i} \sum_{k_2 \in S_j} \frac{\pi_{k_1}}{\pi_{S_i}} \mathcal{M}_{k_1,k_2} \\ &= \sum_{k_1 \in S_i} \sum_{k_2 \in S_j} \pi_{k_1} \mathcal{M}_{k_1,k_2} \end{aligned} \tag{3.61}$$

by reversibility,

$$= \sum_{k_1 \in S_i} \sum_{k_2 \in S_j} \pi_{k_2} \mathcal{M}_{k_2,k_1} = \pi_{S_j} \mathcal{M}'_{S_j,S_i} \tag{3.62}$$

□

3.3 Proofs

As mentioned in the introduction, our work roughly consists of two implications. We first show that ε-pseudorandomness implies weak-binomiality, and then that weakly-binomial implies several bounds on the classical/quantum codes.

3.3.1 ε-Pseudorandom Implies Weakly Binomial Weight Enumerator

Lemma 3.30 (ε-PR hyperedges span weakly-binomial space)**.** *Let $G = (V, E)$ be a d-uniform hypergraph, i.e. $E \subseteq \binom{n}{d}$ - subsets of vertices of V of size exactly d, where $\deg(v) = K$ for each $v \in V$. If G is ε-pseudorandom (ε-PR), then the span of E as words over $\mathbb{F}_2^n$ is (ζ, η) weakly binomial with constants $\zeta = 3\varepsilon^{1/2}$ and $\eta = h(3^{1/d}\varepsilon^{1/(2d)})$.*

Markov Chains of Interest

We analyze the weight enumerator of ε-PR hypergraphs by associating them to natural Markov chains, and then analyzing the stationary distributions of these Markov chains.

Definition 3.31. *Let $G = (V, E)$ be a hypergraph $|V| = n$, and let $C \subseteq \mathbb{F}_2^n$ be the space spanned by the hyperedges of G as vectors over $\mathbb{F}_2^n$. We associate a pair of Markov chains to C: $\mathcal{M}^1, \mathcal{M}^2$ as follows:*

- *$\mathcal{M}^1$ is the Markov chain defined by a random walk on to the Cayley graph of C using the set of generators E.*
- *Let S_i be the set of words in C of weight i, formally defined as $S_i = \{\mathbf{x} \in C \mid |\mathbf{x}| = i\}$. $\mathcal{M}^2$ is defined as the coarse-grained Markov chain with respect to the partition $\{S_i\}$, and some stationary distribution of $\mathcal{M}^1$.*

Let us denote the Markov chains corresponding to the complete hypergraph as $\mathcal{M}^1$ and $\mathcal{M}^2$ (in this case E contains all possible words of weight d). Denote the corresponding stationary distributions as π^1 and π^2. Similarly, let us denote the Markov chains corresponding

to our ε-pseudorandom hypergraph as $\mathcal{M}^{1,\varepsilon}$ and $\mathcal{M}^{2,\varepsilon}$ and the corresponding stationary distributions as $\pi^{1,\varepsilon}$ and $\pi^{2,\varepsilon}$. The main observation is that to understand the weight enumerator of $C = \mathrm{span}(E)$ for our ε-pseudorandom hypergraph, one can instead look at the 1-dimensional stationary distributions $\pi^{2,\varepsilon}$ of $\mathcal{M}^{2,\varepsilon}$. Define $m = \dim(\mathrm{span}(E))$ as the dimension of the code C. It is easy to check that

$$\forall c \in C, \quad \pi_c^{1,\varepsilon} = \frac{1}{|C|} \tag{3.63}$$

and

$$\forall k \in [n], \quad \pi_k^{2,\varepsilon} = \frac{B_k}{2^m}. \tag{3.64}$$

where B_k is the number of words in the code of weight k (weight enumerator).

So now, if a d-uniform hypergraph G is ε-pseudorandom we would like to characterize its corresponding $\mathcal{M}^{1,\varepsilon}, \mathcal{M}^{2,\varepsilon}$ Markov chains as an approximation of the Markov chains $\mathcal{M}^1$ and $\mathcal{M}^2$.

Proposition 3.32. *Let $G = (V, E)$, $|V| = n$, be a d-uniform hypergraph which is ε-PR. Then we have in $\mathcal{M}^{2,\varepsilon}$ the following transition probabilities for all $i \in [n], j \in [d]$:*

$$\left| \mathcal{M}^{2,\varepsilon}_{i,i+d-2j} - \binom{d}{j} (i/n)^j (1 - i/n)^{d-j} \right| \leq \varepsilon. \tag{3.65}$$

Proof. This is the only result in the chapter which crucially uses either Type-1 or Type-2 pseudorandomness. It holds as stated either way. Let us define:

$$S_i = \{\mathbf{z} \in \mathrm{span}(E) : |\mathbf{z}| = i\} \tag{3.66}$$

It holds by definition that:

$$\mathcal{M}^{2,\varepsilon}_{i,i+d-2j} = \sum_{\substack{\mathbf{w} \in \mathrm{span}(E) \\ |\mathbf{w}| = i}} \frac{\pi^{1,\varepsilon}_{\mathbf{w}}}{\pi^{1,\varepsilon}_{S_i}} \frac{A(\mathrm{supp}(w), j)}{|E|} \tag{3.67}$$

If we are in the Type-1 case, observe that this is a convex combination of terms of the form $\frac{A(\text{supp}(\mathbf{w}),j)}{|E|}$. Applying pseudorandomness as well as the triangle inequality completes the proof. In the Type-2 case, observe that the RHS is exactly equal to

$$\mathbb{E}_{\substack{\mathbf{w}\in\text{span}(E)\\|\mathbf{w}|=i}}\frac{A(supp(\mathbf{w}),j)}{|E|}$$

and apply the pseudorandomness definition directly. □

There is an important technical point to make here regarding the walks $\mathcal{M}^1, \mathcal{M}^2$. If d is even, then starting at the all zeros word and adding words of even weight, we can only obtain even weight words in $\mathbb{F}_2^n$. Hence, the walk has nonzero probabilities of landing on even weight words, but zero probability of ever having odd weight. If d is odd then we can obtain any word of any weight. In order for our analysis to make sense in both the d is even and d is odd case, we will need to define the following relation:

$$\gamma_i^{(d)} = \begin{cases} 1 \text{ if } d \text{ is odd} \\ 2 \text{ if } d \text{ is even and } i \text{ is even} \\ 0 \text{ otherwise} \end{cases} \tag{3.68}$$

We note that when $\mathcal{M}^1$ is the Markov chain corresponding to the span of *all* words in $\mathbb{F}_2^n$ of weight d we have that the stationary distribution of its 1-dimensional corresponding chain $\mathcal{M}^2$, π^2 satisfies

$$\forall k \in [n], \quad \pi_k^2 = 2^{-n}\binom{n}{k}\gamma_k^{(d)}. \tag{3.69}$$

Stationary distributions

Having established that the transition probabilities of $\mathcal{M}^{2,\varepsilon}$ are very close to the transition probabilities of $\mathcal{M}^2$, we prove that the stationary distribution π^2 is an upper bound on $\pi^{2,\varepsilon}$,

up to a modest exponential factor. In the following the sets $\mathcal{I}$ and $\mathcal{S}$ may seem cumbersome. The reason we are proving the proposition in this way is that it is possible that the walks $\mathcal{M}^2$ or $\mathcal{M}^{2,\varepsilon}$ never reach a certain weight i. Say in the even case above $\mathcal{M}^1$ never touches an odd weight word so $\mathcal{M}^2$ is never an odd number. In order for the proof to make sense, we need to argue only on the bins with a nonzero stationary probability. This added indexing is used to account for this issue.

Proposition 3.33. *Let $\mathcal{M}^1$ be the Markov chain corresponding to the random walk on $\mathbb{F}_2^n$ using all words of weight d. Let $\mathcal{M}^{1,\varepsilon}$ denote the Markov chain corresponding to the random walk on the Cayley graph $(E, \mathrm{span}(E))$. Let $S = \{i_1, \ldots, i_s\}, |S| = s$ denote the set of bins for which $\pi_i^{\varepsilon} \neq 0$, where $i_j < i_{j+1}$ for all $j < s$. Let $\mathcal{I}$ denote the interval $[n\varepsilon^{1/(2d)}, n(1-\varepsilon^{1/(2d)})]$. Let σ denote the lexicographical ordering of the set $\mathcal{I} \cap S$, and denote $k = |\mathcal{I} \cap S|$.*

Let π^1 and $\pi^{1,\varepsilon}$ denote the stationary distributions of $\mathcal{M}^1$ and $\mathcal{M}^{1,\varepsilon}$, respectively. Let $\mathcal{M}^2, \mathcal{M}^{2,\varepsilon}$ denote the coarse-graining of $\mathcal{M}^1, \mathcal{M}^{1,\varepsilon}$, to the $n+1$ shells $\{S_k\}_{k=0}^n$ using the stationary distributions $\pi^1, \pi^{1,\varepsilon}$ respectively. Let $\pi^2, \pi^{2,\varepsilon}$ denote their corresponding stationary distributions. Then

$$\forall i \in [k] \quad \pi^{2,\varepsilon}_{\sigma(i)} \leq_p \pi^2_{\sigma(i)} \cdot 2^{5\varepsilon^{1/2}(n/2-\sigma(i))}.$$

Proof. Since both $\mathcal{M}^1, \mathcal{M}^{1,\varepsilon}$ are random walks on finite Cayley graphs then by Fact 3.28 they are reversible. Hence by Fact 3.29 the coarse-grained Markov chains $\mathcal{M}^2(\{S_k\}, \pi^2), \mathcal{M}_1^{2,\varepsilon}(\{S_k\}, \pi^{2,\varepsilon})$ are also reversible. This implies

$$\forall i \in [k], \ \mathcal{M}^2_{\sigma(i),\sigma(i+1)}\pi^2_{\sigma(i)} = \mathcal{M}^2_{\sigma(i+1),\sigma(i)}\pi^2_{\sigma(i+1)}. \tag{3.70}$$

$$\forall i \in [k], \ \mathcal{M}^{2,\varepsilon}_{\sigma(i),\sigma(i+1)}\pi^{2,\varepsilon}_{\sigma(i)} = \mathcal{M}^{2,\varepsilon}_{\sigma(i+1),\sigma(i)}\pi^{2,\varepsilon}_{\sigma(i+1)}. \tag{3.71}$$

By definition of the interval $\mathcal{I}$, any transition probability is lower bounded by $(\varepsilon^{1/(2d)})^d = \varepsilon^{1/2}$ over $|j| \leq d$. Hence:

$$\forall i, j |\sigma(i) - \sigma(i+j)| \leq d, \;\; i, j \in [k] \;\; \mathcal{M}^2_{\sigma(i),\sigma(i+j)} \geq \varepsilon^{1/2}. \tag{3.72}$$

Therefore by definition 3.6

$$\begin{aligned} \forall j \in [k] \; \mathcal{M}^{2,\varepsilon}_{\sigma(j),\sigma(j-1)} &\leq \mathcal{M}^2_{\sigma(j),\sigma(j-1)} \cdot (1 + \varepsilon^{1/2}) \\ \mathcal{M}^{2,\varepsilon}_{\sigma(j-1),\sigma(j)} &\geq \mathcal{M}^2_{\sigma(j-1),\sigma(j)} \cdot (1 - \varepsilon^{1/2}) \end{aligned} \tag{3.73}$$

We can write:

$$\forall i \in [k], \quad \frac{\pi^{2,\varepsilon}_{\sigma(i)}}{\pi^{2,\varepsilon}_{\sigma(i+1)}} = \frac{\mathcal{M}^{2,\varepsilon}_{\sigma(i+1),\sigma(i)}}{\mathcal{M}^{2,\varepsilon}_{\sigma(i),\sigma(i+1)}} \leq \frac{\mathcal{M}^2_{\sigma(i+1),\sigma(i)}}{\mathcal{M}^2_{\sigma(i),\sigma(i+1)}} \frac{1+\varepsilon^{1/2}}{1-\varepsilon^{1/2}} = \frac{\pi^2_{\sigma(i)}}{\pi^2_{\sigma(i+1)}} \frac{1+\varepsilon^{1/2}}{1-\varepsilon^{1/2}} \tag{3.74}$$

The first equality follows by reversibility, the first inequality follows by equation 3.73, and the second equality follows from reversibility. Using lemma 3.13, we can write:

$$\forall i \in [k], \quad \frac{\pi^{2,\varepsilon}_{\sigma(i)}}{\pi^{2,\varepsilon}_{\sigma(i+1)}} \leq \frac{\pi^2_{\sigma(i)}}{\pi^2_{\sigma(i+1)}} \left(1 + 3\varepsilon^{1/2}\right) \tag{3.75}$$

Let n_0 be the closest index in $\mathcal{I} \cap S$ to $n/2$ and let $z \in [n]$ be such that $\sigma(z) = n_0$. We have:

$$\frac{\pi^{2,\varepsilon}_{\sigma(i)}}{\pi^{2,\varepsilon}_{\sigma(i+1)}} \cdot \quad \dots \quad \cdot \frac{\pi^{2,\varepsilon}_{\sigma(z-1)}}{\pi^{2,\varepsilon}_{\sigma(z)}} \leq \frac{\pi^2_{\sigma(i)}}{\pi^2_{\sigma(i+1)}} \cdot \dots \cdot \frac{\pi^2_{\sigma(z-1)}}{\pi^2_{\sigma(z)}} \cdot (1 + 3\varepsilon^{1/2})^{(n/2-\sigma(i))}. \tag{3.76}$$

or

$$\pi^{2,\varepsilon}_{\sigma(i)} \leq \pi^2_{\sigma(i)} \left(\frac{\pi^{2,\varepsilon}_{\sigma(z)}}{\pi^2_{\sigma(z)}}\right) \left(1 + 3\varepsilon^{1/2}\right)^{n/2-\sigma(i)} \tag{3.77}$$

Now we apply lemma 3.13 once more:

$$\begin{aligned} \pi^{2,\varepsilon}_{\sigma(i)} \leq \pi^2_{\sigma(i)} \left(\frac{\pi^{2,\varepsilon}_{\sigma(z)}}{\pi^2_{\sigma(z)}}\right) 2^{\wedge} \left\{(n/2 - \sigma(i)) \log(1 + 3\varepsilon^{1/2})\right\} \leq \\ \pi^2_{\sigma(i)} \left(\frac{\pi^{2,\varepsilon}_{\sigma(z)}}{\pi^2_{\sigma(z)}}\right) 2^{5\varepsilon^{1/2}(n/2-\sigma(i))} \leq \left(\frac{\pi^2_{\sigma(i)}}{\pi^2_{\sigma(z)}}\right) \cdot 2^{5\varepsilon^{1/2}(n/2-\sigma(i))} \end{aligned} \tag{3.78}$$

where in the last step we used $\pi^{2,\varepsilon}_{\sigma(z)} \leq 1$.

Since $|\sigma(z) - n/2| = O(1)$, it must be that $\pi^2_{\sigma(z)} \geq n^{-k}$ for some constant $k > 0$ because:

$$\pi^2_{\sigma(z)} = \frac{\binom{n}{n/2-O(1)}}{2^n}\gamma^{(d)}_{\sigma(z)} = \frac{1}{n^{O(1)}} \tag{3.79}$$

Hence

$$\forall i \in [k], \quad \pi^{2,\varepsilon}_{\sigma(i)} \leq_p 2^{5\varepsilon^{1/2}(n/2-\sigma(i))}\pi^2_{\sigma(i)}. \tag{3.80}$$

□

Proof of Lemma 3.30

Let $\mathcal{M}^1$ be the random walk using all words of weight d, and let $\mathcal{M}^{1,\varepsilon}$ be the random walk using generators of the code. Denote the stationary distributions similarly, as is described in the statement of proposition 3.33. By equations 3.64 and 3.69, the stationary distributions have the form:

$$\pi^{2,\varepsilon}_j = \frac{B_j}{2^m} \qquad \pi^2_j = \frac{\binom{n}{j}}{2^n}\gamma^{(d)}_j \tag{3.81}$$

Proposition 3.33 implies that, for $j \in [n\varepsilon^{1/(2d)}, n(1-\varepsilon^{1/(2d)})]$:

$$\frac{B_j}{2^m} \leq \frac{2^{5/2\varepsilon^{1/2}n}\binom{n}{j}}{2^n}\gamma^{(d)}_j \leq \frac{2^{3\varepsilon^{1/2}n}\binom{n}{j}}{2^n}\gamma^{(d)}_j \tag{3.82}$$

or

$$B_j \leq \frac{2^{3\varepsilon^{1/2}n}\binom{n}{j}}{|C^\perp|}\gamma^{(d)}_j \tag{3.83}$$

For j outside the interval, we know nothing. We can then write the general upper bound for

all j by including an additive "error floor term". This error floor term has magnitude:

$$\binom{n}{n\varepsilon^{1/(2d)}} \leq 2^{nh(\varepsilon^{1/(2d)})} \leq 2^{nh(3^{1/d}\varepsilon^{1/(2d)})} \tag{3.84}$$

So we can write the general upper bound as:

$$\forall j \in [0, \dots n], \quad B_j \leq \frac{2^{3\varepsilon^{(1/2)}n}\binom{n}{j}}{|C^{\perp}|}\gamma_j^{(d)} + 2^{nh(3^{1/d}\varepsilon^{1/(2d)})}. \tag{3.85}$$

Hence C is weakly-binomial with parameters

$$\zeta = 3\varepsilon^{1/2}, \quad \eta = h(3^{1/d}\varepsilon^{1/(2d)}).$$

3.3.2 Weakly Binomial Weight Enumerator Implies Upper bound on m_X, Quantum Distance

In this section we use the definition of weakly-binomial spaces to argue an upper bound on the minimal distance of quantum codes.

Lemma 3.34. *Let $\mathcal{C} = (C_X, C_Z)_n$ be a family of quantum $[[n, k, d_{\min}]]$ CSS codes as described where C_X is ε-PR, and $\delta_{\min} = d_{\min}/n > 0$ is a constant independent of n. Let $m_X = \dim(C_X)$ and $m_Z = \dim(C_Z)$, and suppose $m_X \geq m_Z$. Suppose that C_X, C_Z are spanned by generators of weight d, such that in the hypergraph of each of C_X, C_Z the degree is some constant $K = O(1)$. If C_X is (ζ, η) weakly-binomial for $\zeta < \eta^d$ and $\eta \leq 1/4$ then $\delta_{\min} \leq 2h(\eta)$ and $\frac{m_X}{n} \leq h(\eta) + \eta$.*

Proof. For integer t consider the t-th Kravchuk polynomial $P_t^2(x)$. Using equation 3.43 express $P_t^2(x)$ in the Kravchuk basis as:

$$P_t^2(k) = \sum_{i=0}^{t} \binom{2i}{i} \binom{n-2i}{t-i} P_{2i}(k), \tag{3.86}$$

and denote

$$\alpha_{2i} := \binom{2i}{i} \binom{n-2i}{t-i}. \tag{3.87}$$

Let $\{B_k\}$ be the weight enumerator of C_X defined in definition 3.21 and $\{B_k^\perp\}$ be the weight enumerator of $C_X^\perp$ defined in definition 3.22. By MacWilliams theorem 3.24 and equation 3.86:

$$|C_X| \sum_{i=0}^{t} \alpha_{2i} B_{2i}^\perp = \sum_{k=0}^{n} B_k P_t^2(k) \tag{3.88}$$

Since the RHS is a linear combination of B_k's with positive coefficients, we can apply the upper bound we have on B_k assuming weak-binomiality:

$$\begin{aligned} |C_X| \sum_{i=0}^{t} \alpha_{2i} B_{2i}^\perp &\leq \sum_{k=0}^{n} \left[\frac{2^{\zeta n} \binom{n}{k}}{|C_X^\perp|} + 2^{\eta n} \right] P_t^2(k) \\ &= \left[\sum_{k=0}^{n} \frac{2^{\zeta n} \binom{n}{k} |C_X|}{2^n} P_t^2(k) \right] + \left[\sum_{k=0}^{n} P_t^2(k) 2^{\eta n} \right] \end{aligned} \tag{3.89}$$

Now apply equation 3.86 to the first term:

$$= \left[\sum_{k=0}^{n} \frac{2^{\zeta n} \binom{n}{k} |C_X|}{2^n} \sum_{i=0}^{t} \alpha_{2i} P_{2i}(k) \right] + \left[\sum_{k=0}^{n} P_t^2(k) 2^{\eta n} \right]$$

Reversing the order of summation yields:

$$= \sum_{i=0}^{t} \frac{2^{\zeta n} |C_X| \alpha_{2i}}{2^n} \sum_{k=0}^{n} \binom{n}{k} P_{2i}(k) + \sum_{k=0}^{n} P_t^2(k) 2^{\eta n}$$

Now observe that we can interpret the inner sum in the left summand as an inner product between P_{2i} and P_0 - i.e. the constant function. By Lemma 3.17 it must be zero unless $i = 0$. Thus, we have:

$$= \frac{2^{\zeta n}|C_X|\alpha_0}{2^n}2^n + \sum_{k=0}^{n} P_t^2(k)2^{\eta n} \tag{3.90}$$

Now we can apply lemma 3.20

$$\leq 2^{\zeta n}|C_X|\alpha_0 + n\binom{n}{t}^2 2^{\eta n} \tag{3.91}$$

and so by Equation 3.87 we derive the inequality:

$$|C_X|\sum_{i=0}^{t}\alpha_{2i}B_{2i}^{\perp} \leq 2^{\zeta n}|C_X|\binom{n}{t} + n\binom{n}{t}^2 2^{\eta n} \tag{3.92}$$

Dividing by $|C_X|$,

$$\sum_{i=0}^{t}\alpha_{2i}B_{2i}^{\perp} \leq 2^{\zeta n}\binom{n}{t} + n\binom{n}{t}^2\frac{2^{\eta n}}{|C_X|} \leq_p 2^{\zeta n}\binom{n}{t} + \binom{n}{t}^2\frac{2^{\eta n}}{|C_X|}$$

Since each of the α_i and $B_i^{\perp}$ are positive, we have that for all $i \leq t$:

$$\alpha_{2i}B_{2i}^{\perp} \leq_p 2^{\zeta n}\binom{n}{t} + \binom{n}{t}^2\frac{2^{\eta n}}{|C_X|} \tag{3.93}$$

Let $\gamma = \eta^d$, $t = \eta n, j = \gamma n$. Since $t = \eta n$ then for $j = \gamma n = \eta^d n \leq t$ we have:

$$\alpha_{2j}B_{2j}^{\perp} \leq_p 2^{\zeta n}\binom{n}{t} + \binom{n}{t}^2\frac{2^{\eta n}}{|C_X|} \tag{3.94}$$

Applying equation 3.87:

$$\binom{2j}{j}\binom{n-2j}{t-j}B_{2j}^{\perp} \leq_p 2^{\zeta n}\binom{n}{t} + \binom{n}{t}^2\frac{2^{\eta n}}{|C_X|} \tag{3.95}$$

By the definition of CSS codes, $C_Z \subseteq C_X^{\perp}$ so we can lower bound the weight enumerator of $C_X^{\perp}$ with the weight enumerator of C_Z. Since by hypothesis C_Z is also generated by a d-uniform hypergraph of some fixed degree $K = O(1)$, we can apply proposition 3.14:

$$\binom{2j}{j}\binom{n-2j}{t-j}\binom{\frac{n}{d}}{\frac{2j}{d}} \leq_p 2^{\zeta n}\binom{n}{t} + \binom{n}{t}^2 \frac{2^{\eta n}}{|C_X|}$$

Next, by the binomial coefficient approximations in Proposition 2.21:

$$2^{2j+(n-2j)h\left(\frac{t-j}{n-2j}\right)+\frac{n}{d}h\left(2\frac{j}{n}\right)} \leq_p 2^{\zeta n+nh\left(\frac{t}{n}\right)} + 2^{n(2h(t/n)+\eta-m_X/n)} \tag{3.96}$$

Now rewrite equation 3.96 in terms of γ and η:

$$2^{n\left(2\gamma+(1-2\gamma)h\left(\frac{\eta-\gamma}{1-2\gamma}\right)+\frac{1}{d}h(2\gamma)\right)} \leq_p 2^{n(\zeta+h(\eta))} + 2^{n(2h(\eta)+\eta-m_X/n)} \tag{3.97}$$

Now observe by that by our choice of parameters the first summand of RHS is negligible compared to LHS as follows: Let

$$f_{d,\eta}(\gamma) = 2\gamma + (1-2\gamma)h\left(\frac{\eta-\gamma}{1-2\gamma}\right) + \frac{1}{d}h\left(2\gamma\right) - h(\eta)$$

We invoke fact 3.12 by which:

$$f_{d,\eta}(\eta^d) \geq \eta^d. \tag{3.98}$$

By assumption, C_X is (ζ,η)-weakly binomial, for $\zeta < \eta^d$, and so

$$2\gamma + (1-2\gamma)h\left(\frac{\eta-\gamma}{1-2\gamma}\right) + \frac{1}{d}h\left(2\gamma\right) \geq \eta^d + h(\eta) > \zeta + h(\eta).$$

Therefore, together with Equation 3.97 this implies the following upper bound:

$$\frac{1}{d}h(2\gamma) + 2\gamma + (1-2\gamma)h\left(\frac{\eta-\gamma}{1-2\gamma}\right) \leq 2h(\eta) + \eta - m_X/n + o(1)$$

or

$$f_{d,\eta}(\eta^d) - h(\eta) \leq \eta - m_X/n + o(1) \tag{3.99}$$

or

$$o(1) + h(\eta) - f_{d,\eta}(\eta^d) \geq m_X/n - \eta \tag{3.100}$$

Since $f_{d,\eta}(\gamma) > \zeta > 0$,

$$o(1) + h(\eta) + \eta \geq m_X/n \tag{3.101}$$

Since $m_X \geq m_Z$,

$$1 - \rho = \frac{m_X}{n} + \frac{m_Z}{n} \leq 2\frac{m_X}{n} \tag{3.102}$$

where ρ is the quantum code rate. We derive:

$$\frac{1-\rho}{2} \leq \frac{m_X}{n} \tag{3.103}$$

Hence, using equation 3.101

$$\frac{1-\rho}{2} \leq h(\eta) + \eta + o(1) \tag{3.104}$$

Applying the quantum singleton bound Proposition 3.15:

$$\delta_{\min} \leq h(\eta) + \eta + o(1) \tag{3.105}$$

By Proposition 2.21, for $\eta < 1/2$

$$\delta_{\min} \leq 2h(\eta) \tag{3.106}$$

□

3.3.3 Lower Bound on m_X

We can also argue a direct lower bound on the rate from Weak Binomiality. We are indebted to the anonymous referee who suggested this proof.

Theorem 3.35. *Let $\{C_n\}$ be some family of classical codes that are d-sparse and ε-PR. Define the functions $\eta(\varepsilon) := h(3^{1/d}\varepsilon^{1/(2d)})$ and $\zeta(\varepsilon) := 3\varepsilon^{1/2}$. The rate m/n satisfies:*

$$1 - (d-1)\,\eta(\varepsilon) - \zeta(\varepsilon) - o(1) \leq \frac{m}{n} \tag{3.107}$$

Proof. We apply Lemma 3.30 and Proposition 3.14 to obtain:

$$\forall k \leq n/d: \qquad \binom{n/d}{k/d} \leq \frac{\binom{n}{k}2^{\zeta(\varepsilon)n}}{2^{n-m_X}} + 2^{\eta(\varepsilon)n} \tag{3.108}$$

Let us choose c to be some constant such that the L.H.S is much larger than $2^{\eta(\varepsilon)n}$. The required condition is:

$$\binom{n/d}{cn/d} >> 2^{\eta(\varepsilon)n} \tag{3.109}$$

or using Proposition 2.21

$$h(c)\frac{1}{d} > \eta(\varepsilon) \tag{3.110}$$

Let us choose c so that $h(c)\frac{1}{d}$ is *slightly* larger than $\eta(\varepsilon)$. We choose c so that

$$h(c) = d\eta(\varepsilon) + \nu \tag{3.111}$$

We obtain the bound:

$$\binom{n/d}{cn/d} \leq \frac{\binom{n}{cn}2^{\zeta(\varepsilon)n}}{2^{n-m}} \tag{3.112}$$

Using Proposition 2.21,

$$h(c)\frac{n}{d} \leq h(c)n + \zeta(\varepsilon)n - n + m \tag{3.113}$$

or

$$1 - \left(1 - \frac{1}{d}\right) h(c) - \zeta(\varepsilon) \leq \frac{m_X}{n} \tag{3.114}$$

Applying Equation (3.111),

$$1 - \left(1 - \frac{1}{d}\right) (d\eta(\varepsilon) + \nu) - \zeta(\varepsilon) \leq \frac{m_X}{n} \tag{3.115}$$

Asymptotically, we can make ν arbitrarily small with respect to the other parameters. The

lemma follows. □

3.3.4 Proof of Main Theorem

Theorem 3.36. *Let $d, K = O(1)$ be integers, and let $\theta = 0.04$, $\varepsilon_0 = \frac{\theta^{2d}}{9}$ and $0 < \varepsilon < \varepsilon_0$ and $d \geq 4$. Let $\left\{C_X^{(n)}, C_Z^{(n)}\right\}_{n\in\mathbb{N}}$ be a family of quantum CSS codes with a d-sparse generating set, and in which each bit is incident on K generators of $C_Z^{(n)}$ and $C_X^{(n)}$. Let m_X be the rate of C_X seen as a linear code, let G_n be the hypergraph corresponding to the minimal weight words of $C_X^{(n)}$, and suppose $\mathcal{G} = \{G_n\}_n$ is ε-pseudorandom. Define the functions $\eta(\varepsilon) := h(3^{1/d}\varepsilon^{1/(2d)})$ and $\zeta(\varepsilon) := 3\varepsilon^{1/2}$. Then*

$$1 - (d-1)\eta(\varepsilon) - \zeta(\varepsilon) - o(1) \leq \frac{m_X}{n} \leq \eta(\varepsilon) + h(\eta(\varepsilon))$$

$$d_{\min} \leq 2h(h(\eta(\varepsilon)))n \leq \frac{2 \cdot 3^{1/d}\varepsilon^{1/(2d)}}{d^2} \log\left(\frac{1}{\varepsilon}\right)^2 n.$$

Proof. Let (C_X, C_Z) be a CSS code satisfying the assumptions of theorem 3.9 for some $\varepsilon, d, K = O(1)$. Invoking lemma 3.30 we have that C_X is weakly-binomial with parameters $\zeta = 3\varepsilon^{1/2}, \eta = h(3^{1/d}\varepsilon^{1/(2d)})$. We can apply Theorem 3.8 to obtain the lower bound on m_X/n. Now for the upper bound on m_X/n and the distance bound. Since by assumption both codes are K-regular d-uniform we can invoke Lemma 3.34. The lemma states that if $\zeta < \eta^d$ and $\eta \leq 1/4$ then $\delta_{\min} \leq 2h(\eta)$. Indeed, by Proposition 2.21 we can write:

$$\eta^d = \left[h(3^{1/d}\varepsilon^{1/(2d)})\right]^d > 3\varepsilon^{(1/(2d))d} = 3\varepsilon^{1/2} = \zeta$$

The first inequality follows because $\varepsilon^{1/(2d)} < 1/2 \Leftrightarrow \varepsilon < 2^{-2d}$ and $\varepsilon \leq \frac{\theta^{2d}}{9} < 2^{-2d}$ holds by assumption. Observe also that:

$$\eta = h\left(3^{1/d}\varepsilon^{1/(2d)}\right) \leq h\left(3^{1/d}\varepsilon_0^{1/(2d)}\right) = h\left(3^{1/d}\frac{\theta}{3^{1/d}}\right) = h(\theta) < 1/4 \tag{3.116}$$

By applying Lemma 3.34 we upper-bound the distance as follows:

$$\delta_{\min} \leq 2h(h(3^{1/d}\varepsilon^{1/(2d)})) \tag{3.117}$$

For $3^{1/d}\varepsilon^{1/(2d)} \leq 1/2$, applying Proposition 2.21

$$\delta_{\min} \leq 2h(2 \cdot 3^{1/d}\varepsilon^{1/(2d)} \log(1/(3^{1/d}\varepsilon^{1/(2d)}))) \tag{3.118}$$

We again leverage lemma 3.13 which implies for $\varepsilon < \varepsilon_0$

$$2 \cdot 3^{1/d}\varepsilon^{1/(2d)} \log(1/(3^{1/d}\varepsilon^{1/(2d)})) \leq \frac{1}{2} \tag{3.119}$$

Therefore, the term inside the entropy function in equation 3.118 is less than 1/2. It follows that we can apply Proposition 2.21 once more to obtain an upper bound on $\delta_{\min}$.

$$\delta_{\min} \leq 2 \cdot 2 \cdot 3^{1/d}\left[2\varepsilon^{1/(2d)} \log\left(\frac{1}{3^{1/d}\varepsilon^{1/(2d)}}\right)\right] \log\left[\frac{1}{2 \cdot 3^{1/d}\varepsilon^{1/(2d)} \log\left(\frac{1}{3^{1/d}\varepsilon^{1/(2d)}}\right)}\right] \tag{3.120}$$

It holds that

$$\log\left[\frac{1}{2 \cdot 3^{1/d}\varepsilon^{1/(2d)} \log\left(\frac{1}{3^{1/d}\varepsilon^{1/(2d)}}\right)}\right] \leq \log\left(\frac{1}{\varepsilon^{1/(2d)}}\right) + \log\left(\frac{1}{\log\left(\frac{1}{3^{1/d}\varepsilon^{1/(2d)}}\right)}\right) \leq \frac{1}{2d}\log\left(\frac{1}{\varepsilon}\right) \tag{3.121}$$

This implies that:

$$\delta_{\min} \leq \frac{2 \cdot 3^{1/d}\varepsilon^{1/(2d)}}{d^2} \log\left(\frac{1}{\varepsilon}\right)^2 \tag{3.122}$$

□

3.4 Interpretation of Results and Future Directions

The obvious interpretation of our results is that quantum codes which meet our conditions are constrained by our bounds. Hence, the best qualitative statement of results is "Quantum and Classical codes which look too random must be trivial". Classically a locally generated code with $\varepsilon << 1$ must contain almost all the words of $\mathbb{F}_2^n$. The same observation constrains quantum codes, since then one half of the CSS code must be too large to have a reasonable dual. Our hope is that the additional bounds we derive might be useful for researchers studying qLDPC. If, for instance, potential construction is shown to satisfy some "volume law" rather than some "area law", our results indicate that the resulting distance is bounded.

Another interpretation of our classical result is a "weak" Expander Mixing Lemma Converse theorem for our notions of expansion. As we stated in Section 3.1.6, ε in the Type-2 definition has no *a prori* lower bound in terms of d and K, our results provide such a bound under some assumptions:

Theorem 3.37. *Let $\{G_n\}$ be a family of d-uniform K-regular hypergraphs that is ε-Type-2-PR, where the code generated by the edges of the hypergraph has rate m/n. Then, ε is lower bounded by some function of d and m/n: $\varepsilon \geq \varepsilon_0(d, m/n)$.*

There are many open directions from here. We have given a natural correspondence between the Markov walk on codewords and the weight enumerator. It is possible that this connection can be used with other natural pseudorandom definitions to derive bounds. Additionally, I believe the intermediate property may be worth some study on it's own without appealing to pseudorandomness. It is known that codes with large dual distance are weakly-binomial [108], so this property forms some kind of 'anti-qLDPC' property. A code which is weakly binomial does not have much room in the dual for another code, and

vice versa. High dimensional expanders do not have as closely connected notions of expansion [96, 157]. Indeed for graphs it is known that spectral and combinatorial expansion are equivalent while for complexes it is unclear about what the "right" definition of combinatorial expansion is [66, 130]. Hence, while our results rule out one "kind" of expansion there are many other important kinds of expansion. In particular, it would be interesting to rule our co-systolic expanders as good quantum codes [66]. This notion of expansion has a very precise correspondence with "unembeddability" [57], known as the topological overlap property.

Chapter 4

New poly-log LDPC codes for the Quantum Erasure Channel from Erdos-Renyi Graph States

4.1 Introduction

We have seen that lattice codes cannot provide good quantum distance, and our research provides some evidence that strongly "non-lattice" codes cannot provide good distance properties. It is natural to examine the fault tolerant properties of "bad" codes. It is known that many constructions achieving sublinear distance still can correct errors "on average" [52, 79, 85, 107]. For surface codes, the common technique is to prove that 'typical' errors are likely to form small independent clusters and that these clusters are correctable. Here the code needs to have the property that distinct local errors (modulo the stabilizer group) have distinct syndrome patterns. In the case of the Toric code, these clusters take the form of small connected "lines" on the Torus (see Figure 4.1). Since it is a CSS code, we can correct the X and Z errors separately. In this context we are interested in correcting the X error given the Z stabilizer measurements (syndrome). The errors are detected by the Z stabilizers corresponding to the intersections where each of the lines end (see Figure 4.1). The decoder works by "matching" the observed defects at the marked points and acting X operators along paths between marked nodes. If the decoder is successful, the ends of these open lines will be joined by a path of X operators turning the X error into a stabilizer and restoring the code.

By connecting measured "defects" at the marked points above we are turning the error into a topologically trivial operator, and hence a stabilizer of the code. Even if we connect the "incorrect" nodes are connected we will still correct for the error as long as we do not induce a logical error by wrapping all the way around one side of the Torus. It can be proven that correction will succeed with high probability as long as the probability of X and Z error

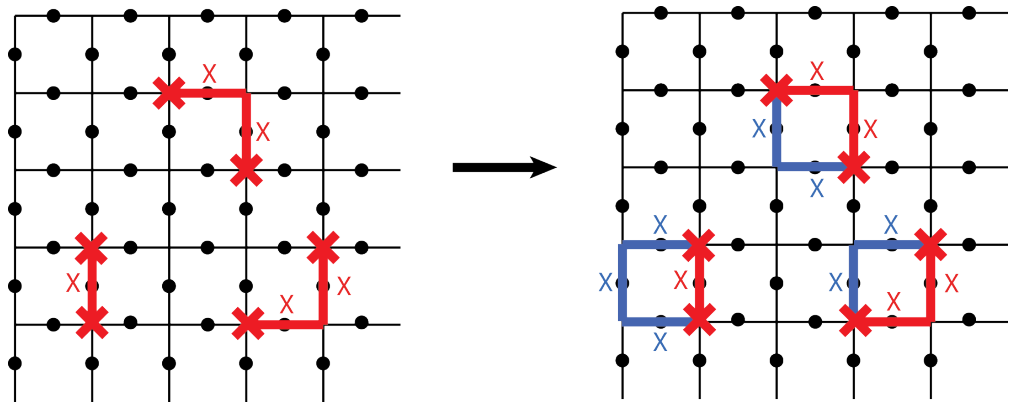


Figure 4.1: The Toric code experiences some X error (red Xs on the left). The error is detected by the Z stabilizers at the marked nodes (intersections of lines) in the Torus. The decoder matches up nearby marked intersections and acts an X "string" between matched nodes (blue Xs). If the decoder is successful (right) the correct matching nodes will be connected.

is less than ~ 0.01, and simulations suggest we can go up to ~ 0.11 [52].

The quantum erasure channel is simpler because we know exactly where X errors might have occurred. Imagine we have some erasures on the Torus (Figure 4.2). As long as the erasures fall into small clusters (the erasure probability is smaller than the percolation threshold) the erasure is correctable. We can simply measure the stabilizers and match up defects which fall in the same cluster. The percolation threshold for the square lattice is 1/2 [100], so the Toric code corrects against erasure errors of any probability smaller than 1/2 as long as the the errors are random are uncorrelated.

We will provide an example which resonates with the Toric code example above. While the actual distance of the Toric code is only $O(\sqrt{n})$, it can correct a constant fraction of errors $\Omega(n)$ in the random setting. Our codes will correct against $\Omega(n)$ many random erasures, but has distance $O(\log(n))$. Our scheme will also code at a very high rate, unlike most surface codes.

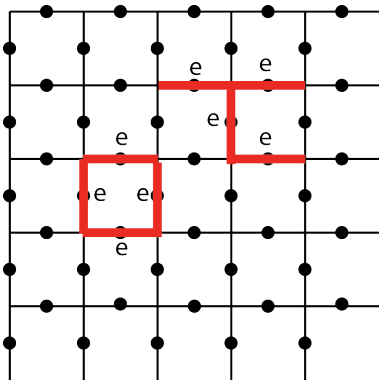


Figure 4.2: A 'typical' erasure error on the Torus.

4.1.1 Previous Work

There are many codes which correct for a linear number of erasure errors [51, 52, 85], but none achieve the capacity (highest possible rate) of the channel while remaining "local". It is known that TZ codes [158] can correct for a constant fraction of erasures [107], even efficiently [79] but it is not clear that these ideas can be used near the capacity. Additionally, there are codes constructed from tilings of hyperbolic surfaces [85] with interesting empirical logical error rates. However, these families are "stiff", since they are based on particular tilings. It is unclear how to vary the rate within the family and hence not clear how to get close to the capacity.

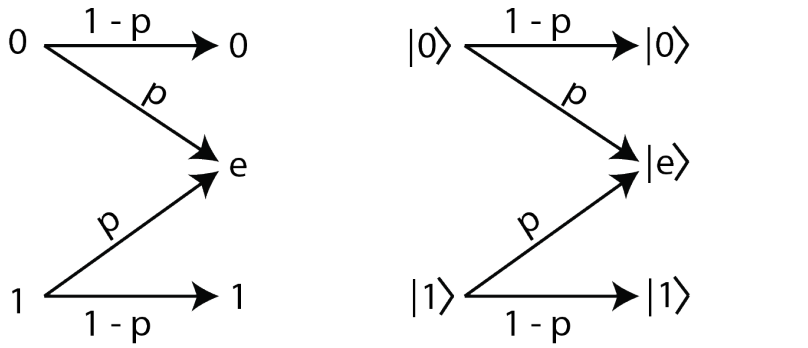
4.1.2 Erasure Channel

The erasure channel [61] can easily be understood with Figure 4.3. In the quantum and classical case, the erasure channel passes it's input through undisturbed with probability $1-p$ and sends it to some "error state" with probability p. The distinguishing feature in both cases is that errors are "detectable". In the classical case we can simply observe the output to determine exactly where transmission errors occurred. In the quantum case we

have the guarantee that $\langle e|0\rangle = 0 = \langle e|1\rangle$, hence we can determine exactly which systems were disturbed during transmission[1]. We can simply measure the observable:

$$P_0 = \begin{array}{c} \\ |0\rangle \\ |1\rangle \\ |e\rangle \end{array} \begin{array}{c} \begin{array}{ccc} |0\rangle & |1\rangle & |e\rangle \end{array} \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{array} \qquad P_1 = \begin{array}{c} \\ |0\rangle \\ |1\rangle \\ |e\rangle \end{array} \begin{array}{c} \begin{array}{ccc} |0\rangle & |1\rangle & |e\rangle \end{array} \\ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{array} \tag{4.1}$$

on each subsystem to determine the erasures.



$$\mathcal{E}(\rho) = (1-p)\rho + p|e\rangle\langle e|$$

Figure 4.3: Erasure Channel

The classical erasure channel [61] was originally formulated as a simpler alternative to the binary symmetric channel. Coding results that were difficult to prove with the Binary symmetric channel (BSC) were often very easy to prove with the erasure channel. It made sense in, many cases to try and build intuition on a difficult problem over the binary symmetric channel by first attempting to solve the analogous erasure channel problem [61, 132, 133, 143]. Capacity achieving and efficient coding schemes for the erasure channel are well studied [12, 132, 133, 141, 147], while there are only a handful of results for the BSC [9]. In the case of the erasure channel, we even have a good understanding of the types of local codes which lead to capacity achieving ensembles [132, 141, 142]. Apart from these considerations, erasure channel coding has found practical applications in many important

[1]Note that the channel is a CPTP map from a 2-dimensional space to a 3-dimensional space

problems [115, 128], not the least of which is coding for "packet loss" over the internet.

As usual, there are many parallels with quantum coding theory. The quantum erasure channel is easier to analyze than the quantum de-polarizing channel (quantum analog of the binary symmetric channel) because it is 'degradable' [54]. This property allows a simple calculation of the capacity of the quantum erasure channel [17], while the capacity of the de-polarizing channel is a log-standing open question in quantum information theory [111]. Efficient (polynomial time) decoding of the erasure channel is simple if one can find an operator in the same coset as the error (see Section 2.3.1), which is simple with some linear algebra over $\mathbb{F}_4$. Decoding the de-polarizing channel, however, is a very non-trivial problem [79], and requires the use of code specific techniques. Additionally, finding good bounds on LDPC codes over the erasure channel [50] seems to be easier than for the de-polarizing channel. Indeed, one such bound for the depolarizing channel can be found in the previous chapter, and this result makes very strong assumptions on the structure of the code.

There are phyiscal motivations for quantum erasure coding as well. Experimental quantum computing normally works by confining the evolution to very particular quantum states [63, 144]. This is important because often these states are easy to manipulate, or are more resilient to noise. "Loss" or "leakage" [164] occurs when the quantum state leaves these particular states. This effect is detectable in many cases, since we can just measure an observable as above to decide erasures. Generically, quantum systems are subject to additional error which may inhibit using erasure coding techniques directly. However, understanding erasure is an important first step it tackling more realistic noise models.

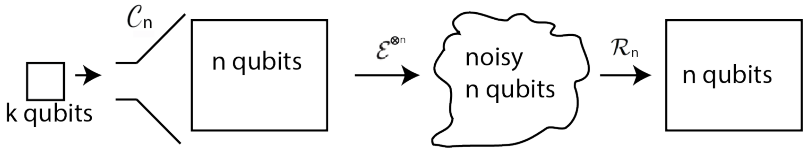


Figure 4.4: The capacity is the largest achievable size of the left hand box with respect to the right hand box, provided we have a scheme to get a "clean" right hand box.

4.1.3 Capacity of Erasure Channel

The *capacity* [146] of a channel (classical or quantum) is the largest rate of reliable information transfer through many indpendent uses of the channel. For this, we imagine we have some information we are trying to send to another party. In the classical case we would have some k bit string, while in the quantum case we would have some k qubit state. We encode it into some large subspace and send it through many copies of some channel. On the other end, we hope that the user is able to decode the information we have sent with high probability. To fix notations, suppose we encode k qubits into n qubits using some encoding scheme $\mathcal{C}_n$. They are transferred through n independent uses of the channel $\mathcal{E}$, and finally they are recovered with some recovery operation $\mathcal{R}_n$ (see Figure 4.4). We require that the recovery be successful with high probability, or

$$\forall\, |\psi\rangle \in [\mathbb{C}^2]^{\otimes k} \quad \left|\left|\mathcal{R}_n(\mathcal{E}^{\otimes n}\mathcal{C}_n(|\psi\rangle\langle\psi|)) - \mathcal{C}_n(|\psi\rangle\langle\psi|)\right|\right|_{Tr} < \varepsilon \tag{4.2}$$

for small ε.

Definition 4.1. *The capacity of a quantum channel $\mathcal{E}$ is the largest number C satisfying the following. For any $\varepsilon > 0$ and for any $\delta > 0$, for n large enough there exists an encoding (CPTP) operation that encodes $k = (C - \delta)n$ qubits into n qubits, and a decoding operation (CPTP) that satisfies:*

$$\forall |\psi\rangle \in [\mathbb{C}^2]^{\otimes k} \quad \left|\left|\mathcal{R}_n(\mathcal{E}^{\otimes n}\mathcal{C}_n(|\psi\rangle\langle\psi|)) - \mathcal{C}_n(|\psi\rangle\langle\psi|)\right|\right|_{Tr} < \varepsilon \tag{4.3}$$

Suppose we have some family of codes $\{C_n\}_{n=1}^{\infty}$, classical or quantum. We will refer to the family as *capacity achieving* if for large enough n we can find a code C_n in the family with rate arbitrarily close to the capacity. In the literature, some places codes are referred to as capacity-approaching [138]. Normally this means that the codes are very good, but cannot come arbitrarily close to the capacity. This is somewhat of a misnomer, since capacity achieving codes "approach" the capacity with large n but might not ever reach it.

For the specific channels we are interested in, the channel capacity is well known. The capacity of the classical erasure channel is $1-p$, and the capacity of the quantum erasure channel is $1-2p$. While the classical result falls under Shannon's seminal information theory paper, [146], the erasure channel has a particularly simple proof.

Theorem 4.2 ([61]). *The capacity of the classical erasure channel with probability p of erasure is $C = 1-p$.*

Proof. The upper bound is simple for the erasure channel. By Chernoff we can expect $(p \pm \delta')n$ symbols to be erased for any $\delta' > 0$ and n large enough. Hence, we can expect $n - (p \pm \delta')n$ many message bits remain. If we had tried to encode at a rate $(p+\delta)$, then there are simply not enough message bits to contain all the information for small enough δ' since $(1-p+\delta)n > (1-p+\delta')n$. Formally, for most erasures (typical ones) we are able to find a word which is not recoverable.

The lower bound can be seen by sampling random linear codes. Let us suppose we are given numbers $\delta, \varepsilon > 0$. Sample a linear code by sampling a random $n(1-p-\delta) \times n$ binary

matrix. The erasure (say it is some set of bits K) corresponds to choosing some set of columns of the generator matrix:

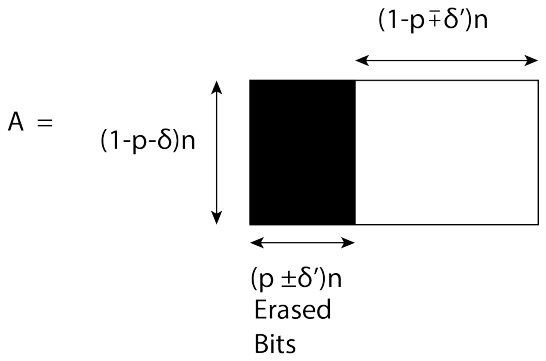


Figure 4.5: Some bits are erased, this corresponds to erasing a portion of the generator matrix.

and erasing them. If the remaining matrix is still full rank (i.e. its rank is equal to $(1-p-\delta)n$, then we can in principle determine the original code word by examining only the bits we have access to. The probability of failure, given our random choice of code can be written as:

$$p_e = \mathbb{E}_{A \sim U[\mathbb{F}_2^{n(1-p-\delta)\times n}]} \mathbb{E}_{K \sim \mathcal{E}} f(A) \tag{4.4}$$

where $f(A)$ is 1 if the rank of the columns of A corresponding to the non erased bits equals $(1-p+\delta)n$ and 0 otherwise. Since our sampled code is independent of the erased bits, we can reverse the order of the expectations, and condition on a fixed erasure K. By Chernoff, we can expect $(p \pm \delta')n$ bits are erased with high probability for any $\delta' > 0$, then the problem reduces to calculating the expected rank of a portion of a random binary matrix. The probability that this matrix fails to be full rank is upper bounded as $O\left(\frac{poly(n)}{2^{(\delta \pm \delta')n}}\right)$[2]. For small enough δ' we should have exponentially small failure probability. □

[2]We can calculate the probability that a uniform matrix is full rank by examining it row by row. The first row must be nonzero, the second row must be nonzero and not equal to the first, etc.

Theorem 4.3 ([17]). *The capacity of the quantum erasure channel $\mathcal{E}^p$ is $1-2p$*

Proof. Generate a uniform random stabilizer code by selecting the first stabilizer at random, the second so that it commutes with the first, etc. Denote the stabilizers chosen this way as $g_1, g_2, ..., g_{n-k}$. Suppose some subset $|K| = pn$ of the qubits have been erased, and let us fix some Pauli operator supported on K, g_0. If all such Paulis lie outside the normalizer of S, $N(S)$, then this erasure is correctable. We can calculate an upper bound on the probability that the erasure is non-correctable using the union bound.

$$\mathbb{P}\Big[g_0 \in N(\langle g_1, g_2, ..., g_{n-k}\rangle\Big] = \mathbb{P}\Big[g_0 \in N(g_1)\Big]\mathbb{P}\Big[g_0 \in N(g_1, g_2)\Big|g_0 \in N(g_1)\Big] \\ \mathbb{P}\Big[g_0 \in N(g_1, g_2, g_3)\Big|g_0 \in N(g_1, g_2)\Big]... \tag{4.5}$$

In each step, we sample g_{i+1} from the normalizer of $\langle g_1, ..., g_i\rangle$. It is easily seen that the normlizer of $\langle g_1, ..., g_k\rangle$ is of size 2^{n-k} if the individual Pauli operators are independent. Hence,

$$\mathbb{P}\Big[g_0 \in N(\langle g_1, ..., g_{i+1}\rangle)\Big|g_0 \in N(\langle g_1, ..., g_i\rangle)\Big] = \frac{|N(\langle g_0, ..., g_{i+1}\rangle)|}{|N(\langle g_0, ..., g_i\rangle)|} = \frac{1}{2} \tag{4.6}$$

Hence, $\mathbb{P}\Big[g_0 \in N(S)\Big] = 1/2^{n-k}$. If we substitute $k = (1-2p-\delta)n$, the probability that any of these fall into the normlizer is upper bounded as

$$\frac{4^n}{2^{n-k}} = \frac{1}{2^{\delta n}} \tag{4.7}$$

The general upper bound follows from [17, 18]. Introducing the proof of this in detail would require a great deal of machinery that will be irrelevant for the current discussion. A proof of the upper bound valid for quantum stabilizer codes is given in the proof of Theorem 4.5. □

4.1.4 Capacity Under Locality

While the above rates are achievable using *uniform random* codes, we may be interested in communicating over then channel using codes with sparse checks. In many decoding schemes codes constructed from sparse random checks are efficiently correctable, so we may be restricted to this case practically [72, 115, 152]. A natural question is: what effect does locality have on our achievable rate?

There are many results which place upper bounds on the achievable rates of codes given *locality* of the checks [12, 34, 72, 141, 142]. Burshtein et al. showed [34] that any ensemble with bounded average degree (regular or not) cannot possibly achieve the capacity. Their analysis operators by relating the "decoding uncertainty" to the entropy of the syndrome. Then this quantity can be bounded by the locality of the parity check matrix.

Theorem 4.4. *Let $A_{L\times N}$ be some parity check matrix with rate R over the erasure channel with erasure probability p. Let p_d be the fraction of rows with weight d. Suppose that k is the median Hamming weight of the rows, so $\sum_{d=1}^{k} p_d = 1/2$. Then,*

$$R \leq 1 - \frac{p}{1 - \frac{1}{2}\left(1 - h\left[\frac{1}{2}\left(1 - \left(1 - \frac{p}{2}\right)^k\right)\right]\right)} \tag{4.8}$$

As k approaches infinity, the entropy term approaches 1 and so does the denominator. Hence, the bound trivializes as we increase k. However, for any fixed k the achievable rate is separated away from the capacity by some constant amount. In the limit of large block size, divergence of the size of check terms is a necessity for capacity achieving ensembles.

To my knowledge, there is only one known strictly analogous bound for the quantum case [50]. There are bounds on the parameters of quantum codes [10, 35, 50, 60, 140], but

very few address the parameters of quantum locally generated codes [50, 60]. Like [34], the Delfosse result is a kind of "syndrome counting" argument.

Theorem 4.5 ([50] Theorem 3.8)**.** *Let $\mathcal{C}$ be a family of stabilizer codes of rate R achieving vanishing error probability for communication over the erasure channel with probability p of erasure ($\mathcal{E}^p$). If the locality of $\mathcal{C}$ is upper bounded by m,*

$$R \leq (1-2p)\frac{1-(1-p)^{m-1}}{1-(1-2p)(1-p)^{m-1}} \tag{4.9}$$

Proof. (sketch) Suppose the set of bits K is erased, and suppose S is the stabilizer of the quantum code $[[n, Rn, \sim, m]]$ with n sufficiently large. Let H be the stabilizer matrix corresponding to S (we write a basis for the generators of S on the rows of H). Given some subgroup $G \subseteq \mathcal{P}_n$, we define $rank(G)$ to be the size of the minimal generating set for G. Let us define:

$$N(S)_K = \{s \in S : s \subseteq K\} \tag{4.10}$$
$$S_K = \{s \in S : s \subseteq K\} \tag{4.11}$$

The erasure is correctable iff $rank(N(S))_K \leq rank(S_K)$. Since $S_K \subseteq N(S)_K$ if $rank(N(S))_K \leq rank(S_K)$ then $S_K = N(S)_K$ and there are no logical operators contained in the erasure. Alternatively, if $rank(N(S))_K > rank(S_K)$ then there must be some undetectable error covered by the erasure (hence uncorrectable).

It is easy to see that:

$$rank(S_K) = rank(H) - rank(H_{[n]\setminus K}) \tag{4.12}$$
$$rank(N(S)_K) = 2|K| - rank(H_K) \tag{4.13}$$

So we derive the inequality:

$$2|K| \leq rank(H) + rank(H_K) - rank(H_{[n]\setminus K}) \tag{4.14}$$

We can rearrange:

$$2\frac{|K|}{n} \leq (1-R) + \frac{1}{n}\Big[rank(H_K) - rank(H_{[n]\setminus K})\Big] \tag{4.15}$$

If we sample a random K according to the distribution induced by the erasure channel, the above equation must be satisfied in expectation in the limit of large block size. If this is not the case, then the Paley-Zygmund inequality implies that we have (at least) some constant probability of failing to satisfy the inequality, and hence $\Omega(1)$ probability of not correcting for the erasure channel. Hence, defining

$$\phi(p) = \lim_{n\to\infty} \mathbb{E}_{K\sim\mathcal{E}^p}\left[\frac{rank(H_K)}{n}\right]$$

we can achieve an inequality of the form:

$$2p \leq 1 - R + \phi(p) - \phi(1-p) \tag{4.16}$$

The main technical hurdles of the result (which we will not prove here) is that the function ϕ is concave and increasing as a function of p. We can see immediately from this that $R \leq 1-2p$ (this is a proof of Theorem 4.3 for stabilizer codes). Concavity implies:

$$\phi(1-p) \geq \phi(p) + (1-2p)\left(\frac{\phi(1)-\phi(p)}{1-p}\right) \tag{4.17}$$

so

$$R \leq 1 - 2p - \frac{1-2p}{1-p}(\phi(1)-\phi(p)) \tag{4.18}$$

Rearranging the above yields the result with the observation $\phi(1) = 1-R$. To see how locality comes into play, observe that

$$\phi(p) \leq \frac{1}{n}(rank(H) - \mathbb{E}_{K\sim\mathcal{E}}(h_K^0)) \tag{4.19}$$

where h_K^0 is the number of rows that are identically zero in the submatrix H_K. This quantity can easily be lower bounded by $rank(H)(1-p)^m$ using a indicator random variable argument.

□

The above proof finds an upper bound on the rate by finding an upper bound on the expected stabilizer rank of the erased portion of the stabilizer matrix. If we wanted to use similar methods on our ensemble, we would need to analyze random H_K from our ensemble. This does not seem like a simple task directly, we provide an interesting way to think about the erasure channel that *effectively* does this. We can see that *acheiving* the capacity is impossible for classical and quantum codes. For reference, we include here plots of the upper bounds from Theorem 4.4 and Theorem 4.5 (Figure 4.6).

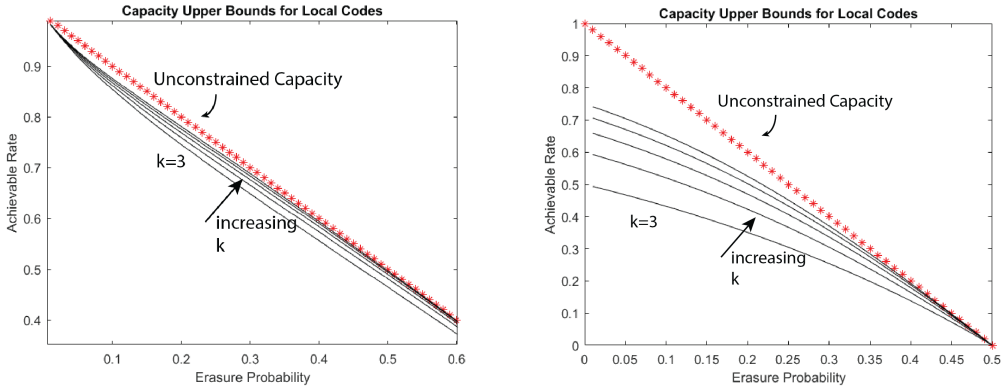


Figure 4.6: Bounds from Theorem 4.4 (left) and Theorem 4.5 (right)

4.1.5 'Barely' Non-Local Codes can Achieve the Classical Capacity

If we cannot acheive the capacity with local codes, it is natural to ask about ensembles which achieve the capacity, but are 'barely' non-local. The best possible behavior we could hope to see is an ensemble with locality that scales as a function of it's gap to optimiality. As we approach the capacity, such a family would have diverging locality, but for any fixed rate, the locality would be constant. We will refer to this as *behavior* 1. The optimal divergence of the checks as a function of the distance to optimality was studied in [142], where it was shown that the results [115] and [147] are essentially optimal. Alternatively, we could ask for something weaker. Say the locality blows up logarithmically with the block size independent of the rate, we will refer to this as *behavior* 2.

Behavior 1 is known to exist among classical codes [115, 147], the first of these [115] are normally referred to as *Tornado Codes*[3]. Suppose we are sampling a bipartite graph according to some distribution of degrees along it's *edges*. Let λ_i be the fraction of edges connected to vertices of degree i on the left, and let ρ_i be the fraction of edges connected to degree i vertices on the right.

Definition 4.6 ([115]). *Define* $Ha(d) = \sum_{i=1}^{d} 1/i$ *as the Harmonic sum truncated at* d. *Suppose there are* βn *check nodes and* n *variable nodes. Define the degree sequences as*

$$\lambda_i = \frac{1}{Ha(d)(i-1)} \;\; \forall i \in \{1, \dots d\} \text{ and } \rho_j = \frac{e^{\alpha}\alpha^{j-1}}{(j-1)!} \tag{4.20}$$

where α *is chosen so that* $\alpha e^{\alpha}/(e^{\alpha}-1) = Ha(d)(d+1)/(\beta d)$ *and we truncate the sequence* ρ *at some sufficiently high* j.

[3]They are referred to as 'Tornado Codes' because there is an iterative decoding procedure which "looks like" a Tornado. The first few iterations make little progress on decoding, but after a certain point is reached the algorithm converges very fast to the decoded codeword

It is shown in [115] that the Tornado sequence corrects for

$$p = \frac{\beta}{1 + \frac{1}{d}} \tag{4.21}$$

erasures with high probability [4]. The general idea is to relate a simple iterative decoding algorithm to a set of differential equations, and to derive these degree distributions as discretized solutions to the differential equations satisfying $error \to 0$. Note that at the capacity $p = \beta$.

For large d we can approximate:

$$p \approx \beta - \frac{\beta}{d} \tag{4.22}$$

The parameter d is used to control how close the code gets to the capacity, while at the same time increasing the locality of the code. The average degree for a node on the right (the average locality) can be calculated as $Ha(d) \cdot (d+1)/(\beta d) \sim \ln(d)/\beta$.

For quantum codes, no such behavior is known. There are not many constructions which yield quantum code from classical ones [43, 106, 158], and it is not obvious how to harness these for high rate local quantum codes. The Tillich-Zemor construction [158] is one of the more prolific ones. Naively "plugging in" Tornado codes to this construction, for instance, yields codes with rate above the capacity (and hence they can not possibly be correctable).

4.2 Result and Proof Ideas

We give quantum codes that achieve the capacity and satisfy behavior 2. These are families of randomly generated stabilizer codes with stabilizers that are polylog local and achieve the

[4] Actually, the authors need to make slight modifications to the sequence above to show this, but it does not effect the locality asymptotically, so we can ignore this.

capacity of the quantum erasure channel. This is weaker than behavior 1, since the Pauli weight diverges for any rate below the capacity, however it does not diverge *as badly* as, say, for uniform random stabilizer codes.

Theorem 4.7. *For any* $0.33 < p < \frac{1}{2}$, *let* $R < 1 - 2p$. *There exist quantum stabilizer codes* $[[n, Rn, \sim, w]]$ *with* $w = O(log^3(n))$ *such that the probability of error when communicating over the erasure channel* $\mathcal{E}^p$ *satisfies:*

$$\textit{error probability} = O\left(\frac{1}{n^{w-1}} + \frac{1}{n^{2(pw-1)}}\right) \tag{4.23}$$

Note that the distance of the quantum code does not matter here. The important point is that the code will survive random errors, rather than worst-case errors. Since local quantum codes with linear distance are unknown, this is a common feature in many results [48, 79].

We use one of the classical to quantum constructions [106] to achieve this. A quantum graph state is a special kind of stabilizer state, associated with a graph G, where the entanglement structure is associated with the cut matrices of the graph (see Section 4.4). We can sample such a state in a "log-sparse" fashion (i.e. each vertex will be connected to $O(\log(n))$ many others with high probability) so that each cut is maximally entangled with high probability. On top of this we add some classical code C with log-sparse parity check matrix. The stabilizer code is then defined as the span of states of the form $Z_{\mathbf{c}} |G\rangle$ where $\mathbf{c} \in C$ and $|G\rangle$ is the graph state. A generic state can be written as:

$$|\psi\rangle = \alpha_{\mathbf{c}_1} Z_{\mathbf{c}_1} |G\rangle + \alpha_{\mathbf{c}_2} Z_{\mathbf{c}_2} |G\rangle + ... \tag{4.24}$$

We demonstrate that the effect of the erasure is to "send" some string to the remaining state, and that this added string is correctable:

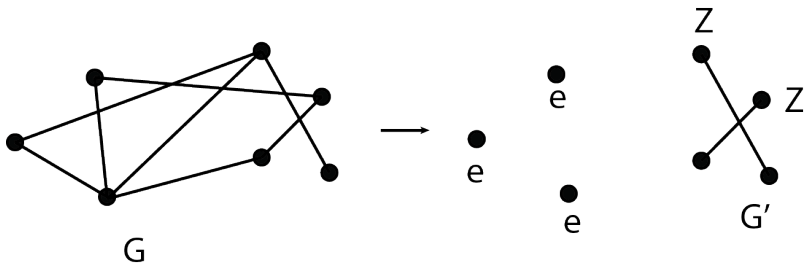


Figure 4.7: Genertic Erasure for Graph State

$$|\psi\rangle \to \alpha_{\mathbf{c}_1} Z_{error} \left(Z_{\mathbf{c}'_1} |G'\rangle + \alpha_{\mathbf{c}_2} Z_{\mathbf{c}'_2} |G'\rangle + ... \right) \tag{4.25}$$

Showing that erasure "sends" some string to the residual state is a well known phenomenon [90], the contribution here is the study of our specific model of random graph code. In order to do this, we establish some properties of sparse classical codes:

Theorem 4.8. *Let H be a $\alpha n \times n$ binary matrix with $\alpha < 1$, where each entry is chosen uniformly at random according to i.i.d. Bern(q) where $q = \frac{w \ln(n)}{n}$. Let C be the code with H as its parity check matrix. Further, let d_C be the distance of the code C. For any constant $\varepsilon > 0$ satisfying $h(\varepsilon) < \alpha$*

$$d_C > \varepsilon n \tag{4.26}$$

with probability at least:

$$1 - O\left(\frac{1}{n^{w\alpha - 2}}\right) \tag{4.27}$$

Compare this to the asymptotic version of the Gilbert-Varshamov bound:

Theorem 4.9 ([117]). *Let H be a uniform random (each bit is 1 with probability 1/2 independently) $\alpha n \times n$ parity check matrix with $\alpha < 1$. For any constant ε satisfying $h(\varepsilon) < \alpha$, the distance of the code satisfies $d > \varepsilon n$ with high probability.*

The GV bound is 'tight' on uniform random instances, i.e. if we sample a random code as described above the distance will not exceed εn with high probability. This is easily calculated from the expected number of codewords of weight less than d:

$$\mathbb{E}\Big[\text{Number of words } \mathbf{c} \in C \text{ with } |\mathbf{c}| \leq d\Big] \approx \binom{n}{d}\frac{1}{2^{\alpha n}} \approx 2^{\wedge}\Big[nH(d/n) - \alpha n\Big] \tag{4.28}$$

If d/n exceeds αn, the mean is exponentially large, while if it is less than αn the mean is exponentially small. In the second case, Markov's inequality implies that there are no codewords of weight less than d with high probability. In the first case, Markov's inequality implies that there must be codewords of size $d + \delta n$ for any $\delta > 0$ with high probability. Our results demonstrate that out log-sparse parity check matrices are 'as good as' uniform random parity check matrices with regards to distance (with high probability). To my knowledge, this is a new result and may be of independent interest, although the same behavior is known for Gallager's LDPC codes (Theorem 2.2 in [72]). Note that this is different from the results established in [98] as they consider sparse generator matrices, not sparse parity check matrices.

The next technical fact concerns the distance of these codes after the erasure channel:

Theorem 4.10. *Let H be a $\alpha n \times n$ binary matrix with $\alpha < 1$, where each entry is chosen uniformly at random according to i.i.d. Bern(q) where $q = \frac{w\ln(n)}{n}$. Let C be the code with H as its parity check matrix. Further, let d_C be the distance of the code C.*

Let K be the first βn bits (corresponding to the first βn columns of H) and assume that $\alpha > \beta$. Denote the code C restricted to the bits $V \setminus K$ as $C_{V\setminus K}$. If ε' satisfies:

$$(1-\beta)h\left(\frac{\varepsilon'}{1-\beta}\right) < \alpha - \beta \quad \text{and} \quad h(\varepsilon') < \alpha \tag{4.29}$$

then,

$$d_{C_{V\setminus K}} > \varepsilon' n \tag{4.30}$$

with probability at least

$$1 - O\left(\frac{1}{n^{w\alpha-2}}\right) \tag{4.31}$$

In the analysis we will be considering the probability that the rows of a random log sparse matrix adds to some word in $C_{V\setminus K}$. Suppose we have some binary word $\mathbf{x} \in \mathbb{F}_2^n$ and some other binary word $\mathbf{y} \in \mathbb{F}_2^n$ where $y_i \sim Bern\left(\frac{w\ln(n)}{n}\right)$. If $\mathbf{x}$ has very small weight (i.e. it's Hamming weight is $O(1)$), then the probability that $\mathbf{x} = \mathbf{y}$ is

$$\mathbb{P}[\mathbf{x} = \mathbf{y}] = \left(1 - \frac{w\ln(n)}{n}\right)^{n-|\mathbf{x}|}\left(\frac{w\ln(n)}{n}\right)^{|\mathbf{x}|} = \frac{1}{poly(n)} \tag{4.32}$$

This is only *polynomially small* with n, hence any uniform union bound would diverge if we are considering the probability $\mathbf{x} = \mathbf{y}$ for any $\mathbf{y}$ inside some subspace C with $|C| = 2^{\Omega(n)}$. The above theorem allows us to consider C with minimum distance $\Omega(n)$, which makes the above exponentially small.

4.3 Mathematical Preliminaries

We will need the Gamma and Polygamma functions to prove our result. The Gamma function is defined as:

$$\Gamma(y) := \int_0^\infty x^{y-1}e^{-x}dx \tag{4.33}$$

The Digamma functions is defined as:

$$\psi^{(0)}(y) := \frac{\Gamma'(y)}{\Gamma(y)} \tag{4.34}$$

and the mth order Polygamma function is defined as:

$$\psi^{(m)}(y) := \frac{d^m}{dy^m}\psi^{(0)}(y) \tag{4.35}$$

The Harmonic numbers (defined as functions here) are defined as:

$$H_{x-1} := \psi^{(0)}(x) + \gamma \qquad H^{(2)}_{x-1} := \frac{\pi^2}{6} - \psi^{(1)}(x) \tag{4.36}$$

where γ is the Euler Mascheroni constant. At integer values, the Harmonic numbers have their usual expression:

$$H_k = \sum_{j=1}^{k} \frac{1}{j} \qquad H^{(2)}_k = \sum_{j=1}^{k} \frac{1}{j^2} \tag{4.37}$$

There are many important properties of these definitions, we list the properties we will need for the proofs:

Fact 4.11. *1. The Gamma function is equal to the factorial at positive integer arguments:*

$$\forall j \in \mathbb{Z},\ > 0 :\ \Gamma(j) = (j-1)! \tag{4.38}$$

2. We can approximate $\psi^{(1)}(k)$ as $\Theta(1/k)$:

$$\forall k > 0,\ \textit{asymptotically large}\ :\quad \frac{1}{2k} \leq \psi^{(1)}(k) \leq \frac{2}{k} \tag{4.39}$$

which implies by definition:

$$\frac{\pi^2}{6} - \frac{2}{k} \leq H^{(2)}_{k-1} \leq \frac{\pi^2}{6} - \frac{1}{2k} \tag{4.40}$$

3. For positive k:

$$\gamma + \ln(k) < H_k < \gamma + \ln(k+1) \tag{4.41}$$

Proof. Item 1 is standard, item 2 follows simply from evaluating the limit:

$$\lim_{k\to\infty} k\psi^{(1)}(k) = 1 \tag{4.42}$$

and item 3 is also standard. It follows from considering the functions $H_k - \ln(k)$ and $H_k - \ln(k+1)$. One can show that they both converge to the same limit, and that $H_k - \ln(k)$ is strictly decreasing while $H_k - \ln(k+1)$ is strictly increasing.

□

In addition we will also need the following technical result on the rank of randomly chosen matrices:

Theorem 4.12. *[Rank of Sparse Matrices, [105]] Let A be a binary $\alpha n \times n$ matrix with $\alpha < 1$, a constant. Suppose each entry of A is sampled independently according to Bern $\left(\frac{w\ln(n)}{n}\right)$. The expected number of linear combinations of rows that add to $\mathbf{0}$, or the expected number of critical sets is*

$$1 + O\left(\frac{1}{n^{w-1}}\right) \tag{4.43}$$

Excluding the "all zeros" linear combination or the empty set, the number of critical sets is:

$$O\left(\frac{1}{n^{w-1}}\right) \tag{4.44}$$

Proof. Slight modifications of the proof of Lemma 3.3.2 in [105] yield the result. Note for reference that

$$1 - \frac{2w\ln(n)}{n} \leq 1 - \frac{2\ln(n)}{n} \tag{4.45}$$

so the upper bounds present in the proof follow immediately. □

Hence, by Markov's inequality, if a matrix A is sampled as described above, then the probability that A is not full rank is upper bounded by $O\left(\frac{1}{n^{w-1}}\right)$. Combined with the Rank Nullity theorem from linear algebra, this theorem says that if we randomly sample a parity check matrix from this ensemble, then the rows will be linearly independent with high probability.

We will also require the following simple result on the sum of binary random variables:

Lemma 4.13 ([72] Lemma 4.1)**.** *Let $\{B_i\}$ be a set of k independent Bern(p) random variables. Then we have that*

$$\mathbb{P}\left[\sum_{i=1}^{k} B_i = 1 \mod 2\right] = \frac{1-(1-2p)^k}{2} \tag{4.46}$$

This establishes the following corollary:

Corollary 4.14. *Let $\{\mathbf{b}_i\}$ be a set of k vectors of random variables such that all entries of each $\mathbf{b}_i$ are independent Bern(p) random variables. For any fixed word $\mathbf{c} \in \mathbb{F}_2^n$ such that $|\mathbf{c}| = g$,*

$$\mathbb{P}\left[\sum_i \mathbf{b}_i = \mathbf{c}\right] = \frac{1}{2^n}\left[1+(1-2p)^k\right]^{n-g}\left[1-(1-2p)^k\right]^g$$

4.4 Graph States

For us, perhaps the most important definition is that of a graph state. We will only state the definition, for examples we invite the reader to examine any of several comprehensive reviews [89, 90]. It may not be clear that the following definitions are equivalent a priori, proofs of equivalence can be found in the stated references.

Definition 4.15. *[Graph state] Given some graph $G = (V, E)$ with no self loops, associate the vertices of the graph G to the numbers $[1, 2, \ldots n]$, we define the graph state $|G\rangle$ according to three equivalent definitions:*

1. *Let A_{top} be the top half of the adjacency matrix for the graph G. This means in the adjacency matrix we set the entries above the main diagonal to 0. We define:*

$$|G\rangle := \frac{1}{2^{n/2}} \sum_{\mathbf{x} \in \mathbb{F}_2^n} (-1)^{\mathbf{x}^T A_{top} \mathbf{x}} |\mathbf{x}\rangle \tag{4.47}$$

2. *Define the following Pauli group elements:*

$$\forall i \in V \;\; S_i := X_i \prod_{j \in N(i)} Z_j \tag{4.48}$$

   The graph state $|G\rangle$ is defined as the unique state stabilized by all S_i.

3. *Let CP_{ij} be the standard controlled phase operation between qubits i and j. Let $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. $|G\rangle$ can be defined as:*

$$|G\rangle = \prod_{(i,j) \in E} CP_{ij} |+\rangle \otimes \ldots \otimes |+\rangle \tag{4.49}$$

Graph states satisfy an important orthogonality property:

Lemma 4.16. *[91] Let $\mathbf{x} \in \mathbb{F}_2^n$ be some nonzero binary string. Then,*

$$\langle G| Z_{\mathbf{x}} |G\rangle = 0 \tag{4.50}$$

Proof. Since Z operators commute with controlled phase operators, according to Definition 4.15 we calculate:

$$\langle G| Z_{\mathbf{x}} |G\rangle \tag{4.51}$$

$$= \langle +| \dots \langle +| \prod_{ij} CP_{ij} Z_{\mathbf{x}} \prod_{ij} CP_{ij} |+\rangle \dots |+\rangle \tag{4.52}$$

$$= \langle +| \dots \langle +| Z_{\mathbf{x}} |+\rangle \dots |+\rangle = 0 \tag{4.53}$$

Where we used $CP_{ij}CP_{ij} = \mathbb{I}$ to get from Equation (4.52) to Equation (4.53). □

With this definition in hand we can define a quantum graph code.

Definition 4.17 ([145]). *Given a graph G and a $[n, k, d]$ classical code C over $\mathbb{F}_2$, we define a graph code (G, C) as the linear span of quantum states of the form:*

$$Z_{\mathbf{c}} |G\rangle \tag{4.54}$$

where $\mathbf{c}$ *is any binary code word in C.*

Now we will present a few simple facts regarding this definition.

Lemma 4.18. *Let (G, C) be as defined in Definition 4.17 and suppose it has parameters $[[n, k_Q, d_Q, w_Q]]$. Suppose the code C has parameters $[n, k, d, w]$. Denote the maximum vertex degree in G as K_{max} and the minimum vertex degree as K_{min}.*

1. $k = k_Q$

2. *[106] Let* $\{\mathbf{h}_1, \ldots, \mathbf{h}_{n-k}\}$ *be some minimal weight generating set for the code* $C^\perp$*, and define:*

$$g_j := \prod_{i \in supp(\mathbf{h}_j)} S_i \tag{4.55}$$

where S_i *is defined in Equation (4.48). The code* (G, C) *is a stabilizer code code with stabilizer generators* $\{g_j\}$ *for all* j*.*

3. *[106]* $w_Q \leq wK_{max}$

4. *[106]* $d_Q \leq K_{min}$

Proof. We can see Item 1 immediately from Lemma 4.16 and Definition 4.17.

We can prove 2 as follows. We claim that the set $\{g_j\}$ form a complete, minimal set of generators for the stabilizer group for the code defined as the span of all $Z_{\mathbf{c}} |G\rangle$. Independence of these operators follows from independence of the binary vectors $\mathbf{h}_i$. They also all clearly commute since $[S_i, S_j] = 0$ for all i and j. It remains to show that these operators stabilize the code. The code (G, C) is the span of states of the form $Z_{\mathbf{c}} |G\rangle$ where $\mathbf{c} \in C$. Suppose g_j has the form:

$$\prod_{i \in supp(\mathbf{h}_j)} S_i = (-1)^\phi Z_{\mathbf{w}} \left(\prod_{j \in supp(\mathbf{h}_k)} X_j \right) \tag{4.56}$$

for some binary vector $\mathbf{w}$. We have just rewritten the operator so that the X operators are on the right and the Z operators on the left. Potentially we have introduced a phase $\phi \in \{0, 1\}$.

$$\prod_{i \in supp(\mathbf{h}_j)} S_i Z_c |G\rangle = (-1)^\phi Z_{\mathbf{w}} \left(\prod_{i \in supp(\mathbf{h}_j)} X_i \right) Z_{\mathbf{c}} |G\rangle \tag{4.57}$$

Since $\mathbf{h}_j \in C^\perp$,

$$\left[\prod_{i \in supp(\mathbf{h}_j)} X_i, Z_{\mathbf{c}}\right] = 0 \tag{4.58}$$

so

$$(-1)^\phi Z_{\mathbf{w}} \left(\prod_{i \in supp(\mathbf{h}_j)} X_i\right) Z_{\mathbf{c}} |G\rangle = \tag{4.59}$$
$$Z_{\mathbf{c}} (-1)^\phi Z_{\mathbf{w}} \left(\prod_{i \in supp(\mathbf{h}_j)} X_i\right) |G\rangle = Z_{\mathbf{c}} |G\rangle$$

Item 3 follows from Item 2. The stabilizers are given, and the weight of each stabilizer is upper bounded (by definition) by the maximum vertex degree times the maximum weight of $\mathbf{h}_j$.

We sketch the proof of Item 4. Focusing on the bit in the graph with smallest degree, an adversary can "disconnect" this vertex from the rest of the graph by enacting a unitary on this bit and its neighbors. Then, the adversary can induce undetectable phase on the code by acting a Pauli on the disconnected bit. Hence the adversarial distance is upper bounded by the minimal vertex degree.

□

Graph states correspond to a "standard form" for stabilizer states [78]. They have a simple bi-partite entanglement structure which makes many of their entanglement properties intuitive.

Theorem 4.19. *Let $G = (V, E)$ be a graph on n vertices. Further, let K be some subset of the bits with complement $V \setminus K$. Let A_{cut} be the cut matrix across the cut $(K, V \setminus K)$, let A_K be the upper half of the adjacency matrix restricted to the bits K, and let G' be the subgraph*

induced by the vertices $V \setminus K$.

1. *[91]Suppose a user measures the bits* K *in the computational basis. Every bit string* y *is equally likely and after the user measures and gets bit string* $\mathbf{y}$*, the resulting quantum state is:*

$$\langle \mathbf{y}|_K |G\rangle = \frac{1}{2^{|K|/2}} (-1)^{\mathbf{y}^T A_K \mathbf{y}} Z_{A_{cut}\mathbf{y}} |G'\rangle \tag{4.60}$$

2. *[90] Denote the entanglement entropy of the graph state across the cut* $(K, V \setminus K)$ *as* $E^{(K,V\setminus K)}(|G\rangle)$*. Then,*

$$E^{(K,V\setminus K)}(|G\rangle) = \text{rank}_{\mathbb{F}_2}(A_{cut}) \tag{4.61}$$

Proof. Item 1 follows from item 3 in Definition 4.15. If a particular bit measures $|1\rangle$, then the result is a Z gate on it's neighbors, since each of the controlled phase gates are "active" in this case. The resulting Z string is simply the sum of all the activated Z gates $\mod 2$. The additional factor of $(-1)^{\mathbf{y}^T A_K \mathbf{y}}$ results from controlled phase gates evaluated on the string $\mathbf{y}$.

For item 2, the stabilizers of a graph state are all Pauli operators of the form $X_i \prod_{j\in N(i)} Z_j$. By [67] we can calculate the entanglement entropy by breaking up the stabilizer into three sets:

1. Stabilizers supported only on K
2. Stabilizers supported only on $V \setminus K$
3. Stabilizers not strictly supported on either set

The "rank" or the size of the minimal generating set equals the entanglement entropy. WLOG, assume that the set K is smaller. Let B be some subset of K. The third set is generated by stabilizers of the form:

$$(-1)^{\phi} \prod_{i \in B} X_i \prod_{j \in N(i)} Z_j \tag{4.62}$$

where the Z string is nonzero. It is clear that this corresponds to some nonzero linear combination of the rows of the cut matrix. Hence the rank of the cut matrix determines the rank of the third set.

□

Theorem 4.19 implies that after one measures part of a labeled graph state, the residual state is as described in Section 4.2. The effect is to "send" some string $Z_{A_{cut}\mathbf{j}}$ to the rest of the graph state. For a graph code, we would get the added string $Z_{A_{cut}\mathbf{j}}$ along with some phase that depends on the the measured string:

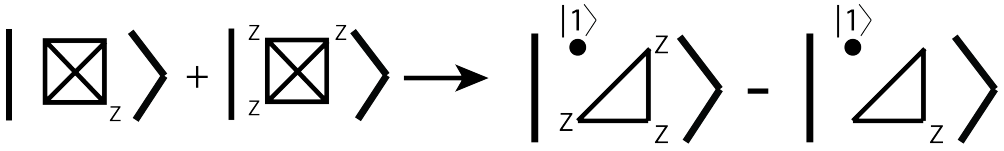


Figure 4.8: Typical Erasure for Graph Code

4.4.1 Connectivity Threshold is also an "Entanglement Threshold"

We are in a position not to state an interesting fact, which points to the "engine" of our result. Imagine we sample a 'barely-connected' Erdos Renyi graph $G \sim ER(n, w \ln(n)/n)$. If $w > 1$ [64], then as $n \to \infty$, G will be connected with high probability (the entire graph is one giant component), while if $w < 1$ then G will be disconnected with high probability

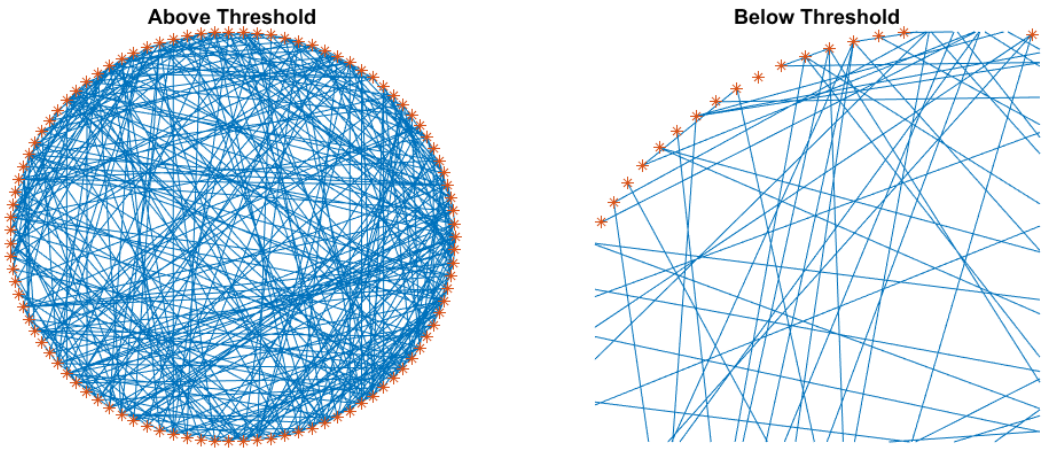


Figure 4.9: Erdos Renyi Connectivity Threshold

(there will be an isolated point). $w = 1$ is a sharp threshold for the connectivity property (Figure 4.9).

We obtain an interesting quantum analog from this fact alone. Let $G_1 \sim ER(n, (1+\varepsilon)\ln(n)/n)$, $G_2 \sim ER(n, (1-\varepsilon)\ln(n)/n)$, and suppose we construct the graph states $|G_1\rangle$ and $|G_2\rangle$. Let us fix some partition of the graph (A, B) with $A = \alpha n$ and $\alpha \in (0, 1/2)$. With very high probability, according to Theorem 4.19 and Theorem 4.12 $|G_1\rangle$ will be maximally entangled across the cut with high probability. $|G_2\rangle$, however, fails to be maximally entangled across the cut with at least constant probability. This is easily seen appealing to [64]. There will be a disconnected point with high probability, that point will be contained in A with at least constant probability. If that point happens to be in A, then the cut matrix contains a row of zeros and hence the quantum state is not maximally entangled across the cut. The ER connectivity threshold also corresponds to an 'entanglement threshold' in graph states.

The important point here is that fixed sets happen to be maximally entangled with high probability. Alternatively, random sets are maximally entangled with high probability as

long as the set is not correlated with the graph. Informally, this is the *reason* that the construction works. An erasure channel will randomly "pick " sets of qubits to erase. If it sees a maximally mixed state, then it cannot get information about the quantum code and it should be correctable in principle. Then, if we build quantum codes from these states, we can expect them to be resilient to erasures while still being 'local'. Indeed, the expected node degree is only $O(\log(n))$.

4.5 Proofs

We have already given an informal description of the construction in Section 4.2, now we are in a position to formally define our construction. Let p be the erasure probability, and suppose we are interested in coding at a rate of $(1-2p-\delta)$ for any $\delta > 0$. We will sample a graph and a code sparsely to construct a quantum graph code. Let $G = ER\left(n, \frac{w\ln(n)}{n}\right)$ be some Erdos-Renyi graph for some constant w TBD. Let C be a code constructed by sampling a random $(2p+\delta)n \times n$ parity check matrix H where each $H_{ij} \sim Bern\left(\frac{w\ln(n)}{n}\right)$ and all H_{ij} are i.i.d. The construction is the quantum graph code (G, C). The end result of this section will be that the code we have described has vanishing probability of decoding error over the erasure channel with probability p of erasure.

4.5.1 Conditions for Successful Decoding

4.5.2 Coset Measurement

For our recovery scheme, we will require the notion of a "coset" measurement. Such a measurement allows us to distinguish between different labeled graph states that are in different cosets of C. Consider C as a subgroup of $\mathbb{F}_2^n$. For each vector $\mathbf{e} \in \mathbb{F}_2^n$ we have the coset $C + \mathbf{e}$. Let $\{C + \mathbf{e}\}$ be the set of all cosets of the code C. For each coset $C + \mathbf{e}$, define the subspace:

$$V_{\mathbf{e}} = \text{span}_{\mathbf{c} \in C} Z_{\mathbf{c}+\mathbf{e}} |G\rangle \tag{4.63}$$

and let $P_{\mathbf{e}}$ be the projector onto this subspace. Let $M = \{P_{\mathbf{e}}\}$ be a measurement with projectors $P_{\mathbf{e}}$. It is well known that distinct cosets are disjoint, so $P_{\mathbf{e}} P_{\mathbf{e}'} = 0$ for $\mathbf{e} \neq \mathbf{e}'$ (modulo C). Hence, such an observable is well defined and serves to distinguish cosets. The idea here will be to use the coset measurement to determine the 'error string' as in Section 4.2.

4.5.3 Recovery Operation

Suppose we send the state $|\psi\rangle \in (G, C)$ through an erasure channel and pn bits are erased. Denote the dimension of the code C as Rn, and the set of erased bits K. Construct the following $[(R + p)n] \times (1 - p)n$ matrix F. Let the first Rn rows of F be the generators of the code C restricted to the non erased bits, and let the remaining rows be the transpose of the cut matrix between the erased bits and the non-erased bits:

$$F = \quad (R+p)n \left\updownarrow \left[\begin{array}{c} \overset{(1-p)n}{\longleftrightarrow} \\ C_{V\setminus K} \\ A_{cut}^T \end{array} \right] \right. \tag{4.64}$$

It is important to note here that the top portion of the matrix is the generators of C restricted to $V \setminus K$, and not the generators of $C_{V\setminus K}$. Further, let G' be the subgraph of G induced by the vertex set $V \setminus K$.

Lemma 4.20. *If the matrix F is full rank over $\mathbb{F}_2$, then there is a quantum operation $\mathcal{R}$ which recovers from the erasure.*

Proof. Denote the quantum state before the channel as:

$$|\psi\rangle = \sum_{\mathbf{c}\in C} b_{\mathbf{c}} Z_{\mathbf{c}} |G\rangle \tag{4.65}$$

Suppose without loss of generality that the erased bits, K, are the first pn bits (or rearrange the bits so that this is the case). For each codeword $\mathbf{c} \in C$, decompose $\mathbf{c}$ as the concatenation of its value on the erased bits with its value on the non erased bits: $\mathbf{c} = (\mathbf{c}_K, \mathbf{c}_{V\setminus K})$. By the principle of implicit measurement and by Theorem 4.19, after the erasure channel we are left with the following density matrix on the subsystems $V \setminus K$:

$$\rho = \sum_{\mathbf{j}\in\mathbb{F}_2^{|K|}} \rho_{\mathbf{j}} |\phi_{\mathbf{j}}\rangle \langle\phi_{\mathbf{j}}| \tag{4.66}$$

where:

$$\sum \rho_{\mathbf{j}} = 1 \quad \text{and} \quad |\phi_{\mathbf{j}}\rangle = \sum_{\mathbf{c}\in C} b_{\mathbf{c}} (-1)^{\mathbf{j}\cdot\mathbf{c}_K} Z_{\mathbf{c}_{V\setminus K} + A_{cut}\mathbf{j}} |G'\rangle_{V\setminus K} \tag{4.67}$$

If F is full rank, then the coset measurement of $C_{V\setminus K}$ can be used to determine the string $\mathbf{j}$. Indeed, if F is full rank then each element of the range of A_{cut} belongs to a different coset. So, measuring which coset the graph state falls into will determine the string $\mathbf{j}$. Note further that such a measurement will not disturb the encoded information since for fixed $\mathbf{j}$ the states

$\{Z_{\mathbf{c}_{V\setminus K}+A_{cut}\mathbf{j}}\}$ all lie in a particular coset of $C_{V\setminus K}$.

Let us describe this more formally. Suppose that $A_{cut}\mathbf{j}$ falls into a particular coset of $C_{V\setminus K}$:

$$A_{cut}\mathbf{j} = \mathbf{c}''_{V\setminus K} + \mathbf{e} \tag{4.68}$$

for some fixed $\mathbf{c}''_{V\setminus K} \in C_{V\setminus K}$. If $P_{\mathbf{e}'}$ is the projector onto a different coset, then $P_{\mathbf{e}'} |\phi_{\mathbf{j}}\rangle = 0$:

$$P_{\mathbf{e}'} |\phi_{\mathbf{j}}\rangle = \sum_{\mathbf{c},\mathbf{c}'\in C} \sigma_{(\mathbf{c},\mathbf{c}',\mathbf{e}',\mathbf{j})} \langle G'| Z_{A_{cut}\mathbf{j}+\mathbf{c}_{V\setminus K}+\mathbf{c}'_{V\setminus K}+\mathbf{e}'} |G'\rangle \tag{4.69}$$

for some operators $\sigma_{(\mathbf{c},\mathbf{c}',\mathbf{e}',\mathbf{j})}$. Since $\mathbf{e}'$ corresponds to a distinct coset, the string $A_{cut}\mathbf{j} + \mathbf{c}_{V\setminus K} + \mathbf{c}'_{V\setminus K} + e'$ is always in some nonzero coset, so by Lemma 4.16, $P_{\mathbf{e}'} |\phi_{\mathbf{j}}\rangle = 0$.

The projector $P_{\mathbf{e}}$ has the following effect on the state $|\phi_{\mathbf{j}}\rangle$:

$$\begin{aligned} P_{\mathbf{e}} |\phi_{\mathbf{j}}\rangle &= \left[\sum_{\mathbf{c}'_{V\setminus K}} Z_{\mathbf{c}'_{V\setminus K}+\mathbf{e}} |G'\rangle \langle G'| Z_{\mathbf{c}'_{V\setminus K}+\mathbf{e}} \right] |\phi_{\mathbf{j}}\rangle \\ &= \sum_{\substack{\mathbf{c}'_{V\setminus K},\mathbf{c}_{V\setminus K}: \\ \mathbf{c}'_{V\setminus K}+\mathbf{e}=\mathbf{c}_{V\setminus K}+A_{cut}\mathbf{j}}} b_{\mathbf{c}}(-1)^{\mathbf{j}\cdot\mathbf{c}_K} Z_{\mathbf{c}'_{V\setminus K}+\mathbf{e}} |G'\rangle \end{aligned} \tag{4.70}$$

The relation $\mathbf{c}'_{V\setminus K} + \mathbf{e} = \mathbf{c}_{V\setminus K} + A_{cut}\mathbf{j}$ fixes $\mathbf{c}'_{V\setminus K}$ given $\mathbf{c}_{V\setminus K}$. It is easy to see that we get back exactly the state $|\phi_{\mathbf{j}}\rangle$.

To complete our description of the recovery operation $\mathcal{R}$ we need to give the unitary that can recover the original quantum state $|\psi\rangle$ given the state $|\phi_{\mathbf{j}}\rangle$ and given the string $\mathbf{j}$. By appending extra copies of the state $|+\rangle$, we can achieve the state:

$$\sum_{\mathbf{c}\in C} b_{\mathbf{c}}(-1)^{\mathbf{j}\cdot\mathbf{c}_K} Z_{\mathbf{c}_{V\setminus K}+A_{cut}\mathbf{j}} |+\ldots+\rangle_K |G'\rangle_{V\setminus K} \tag{4.71}$$

Now we an apply controlled phase operations (using Definition 4.15) to transform the state to:

$$\sum_{\mathbf{c}\in C} b_{\mathbf{c}}(-1)^{\mathbf{j}\cdot\mathbf{c}_K} Z_{\mathbf{c}_{V\setminus K}+A_{cut}\mathbf{j}} |G\rangle_V \tag{4.72}$$

Now we can apply the unitary $Z_{A_{cut}\mathbf{j}}$ to transform the state to:

$$\sum_{\mathbf{c}\in C} b_{\mathbf{c}}(-1)^{\mathbf{j}\cdot\mathbf{c}_K} Z_{\mathbf{c}_{V\setminus K}} |G\rangle \tag{4.73}$$

Since

$$\langle G|Z_{\mathbf{c}_{V\setminus K}} Z_{\mathbf{c}'_{V\setminus K}}|G\rangle = 0 = \langle G|Z_{\mathbf{c}} Z_{\mathbf{c}'}|G\rangle \tag{4.74}$$

the unitary that sends $Z_{\mathbf{c}_{V\setminus K}} |G\rangle \to Z_{\mathbf{c}} |G\rangle$ is well defined. We can apply it to obtain:

$$\sum_{\mathbf{c}\in C} b_{\mathbf{c}}(-1)^{\mathbf{j}\cdot\mathbf{c}_K} Z_{\mathbf{c}} |G\rangle \tag{4.75}$$

Finally we can apply a unitary that is diagonal in the $Z_{\mathbf{c}} |G\rangle$ basis to remove the phase and recover $|\psi\rangle$ □

4.5.4 The Classical Codes we Sample are as Good as Totally Random Codes

Throughout this section we will have some n and w in mind. We will denote:

$$a := 1 - \frac{2w\ln(n)}{n} \tag{4.76}$$

The goal is to show that codes from our ensemble have linear distance after the erasure channel. We find it useful to first prove that the code itself has linear distance with high probability:

Lemma 4.21. *Let H be a $\alpha n \times n$ binary matrix with $\alpha < 1$, where each entry is chosen uniformly at random according to i.i.d. Bern(q) where $q = \frac{w\ln(n)}{n}$. Let C be the code with H as its parity check matrix. Further, let d_C be the distance of the code C. For any constant $\varepsilon > 0$ satisfying $h(\varepsilon) < \alpha$*

$$d_C > \varepsilon n \tag{4.77}$$

with probability at least:

$$1 - O\left(\frac{1}{n^{w\alpha-2}}\right) \tag{4.78}$$

Proof. We use first moment methods. Let X be a random variable equal to the number of subsets of columns of H with size at most εn that sum to zero. X can equivalently be defined as the number of words of C with weight less than εn. We calculate using Corollary 4.14

$$\mathbb{E}(X) = \sum_{k=1}^{\varepsilon n} \binom{n}{k} \left(\frac{1+a^k}{2}\right)^{\alpha n} \tag{4.79}$$

Let us define the function:

$$f(k) := \binom{n}{k} \left(1+a^k\right)^{\alpha n} \tag{4.80}$$

We can make this function continuous and differientiable by substituting Gamma functions for factorials:

$$\binom{n}{k} = \frac{\Gamma(n+1)}{\Gamma(k+1)\Gamma(n-k+1)} \tag{4.81}$$

We will show that we can find an upper bound for $f(k)$ by examining the endpoints $k = 1$ and $k = \varepsilon n$. Define two intervals $I_1 = \left[1, \frac{zn}{\ln(n)}\right]$ and $I_2 = \left[\frac{zn}{\ln(n)}, \varepsilon n\right]$ where z is some large constant to be determined. The property we claim follows if we can show that $f'(k) < 0$ in the interval I_1 and that the function $f'(k)$ has exactly one zero in the interval I_2. The remainder of the proof will fall into two parts. In part a we will demonstrate that $f'(k) < 0$ in I_1 and in part b we will demonstrate that $f'(k)$ has exactly one zero in I_2.

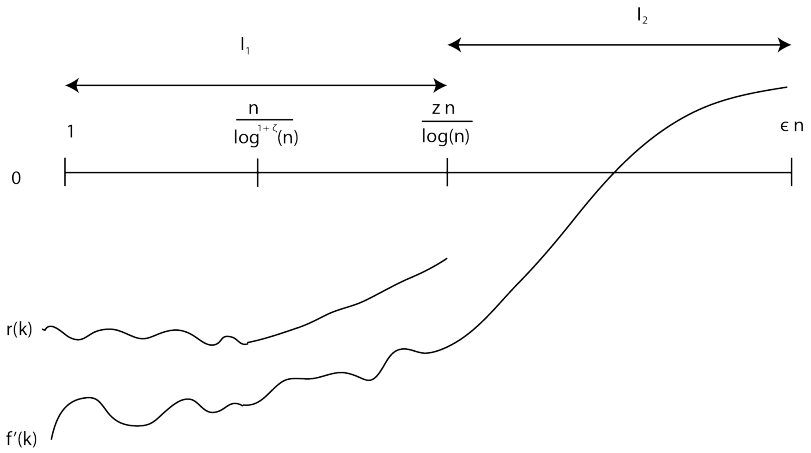


Figure 4.10: An illustration of our proof method. We demonstrate that f' has the above form, which implies exactly one local minimum. Therefore, f must be maximized at one of the endpoints (either at $k = 1$ or at $k = \varepsilon n$.

Part a

We will further divide interval I_1 into two other intervals: $\left[1, \frac{n}{\ln^{1+\zeta}(n)}\right]$ and $\left[\frac{n}{\ln^{1+\zeta}(n)}, \frac{zn}{\ln(n)}\right]$ where ζ is some small positive constant. We will provide another function $r(k)$ which upper bounds f' in both of these intervals. We first demonstrate that $r(k) < 0$ in the interval $\left[1, \frac{n}{\ln^{1+\zeta}(n)}\right]$. Then we will show that $r'(k)$ has a positive slope in the interval $\left[\frac{n}{\ln^{1+\zeta}(n)}, \frac{zn}{\ln(n)}\right]$ and has a negative endpoint at $\frac{zn}{\ln(n)}$, implying $f'(k) < 0$ throughout I_1.

We calculate:

$$f'(k) = v(k)\left(a^k\left(\alpha n \ln(a) + H_{n-k} - H_k\right) + H_{n-k} - H_k\right) \tag{4.82}$$

where $v(k) > 0$ We are interested in the sign of $f'(k)$, so it is sufficient to study $\frac{f'(k)}{v(k)}$. Define:

$$b(k) := a^k\left(\alpha n \ln(a) + H_{n-k} - H_k\right) + H_{n-k} - H_k \tag{4.83}$$

For large enough n we can take limits to show:

$$\frac{-3w\ln(n)}{n} \leq \ln(a) \leq -\frac{2w\ln(n)}{n} \tag{4.84}$$

So,

$$b(k) \leq a^k \left(H_{n-k} - H_k - 2w\alpha \ln(n)\right) + H_{n-k} - H_k \tag{4.85}$$

By Fact 4.11:

$$H_{n-k} \leq \ln(n) + \gamma \tag{4.86}$$

so we have $b(k) \leq r(k)$ where:

$$r(k) := a^k \left(-2w\alpha + 2\right) \ln(n) + H_{n-k} - H_k \tag{4.87}$$

We need to show that the function $r(k) < 0$ for all k in the interval $\left[1, z\frac{n}{\ln(n)}\right]$.

For any $k \in \left[1, \frac{n}{\ln^{1+\zeta}(n)}\right]$ the function $r(k)$ is negative for sufficiently large n. Indeed for $k = \frac{n}{\ln^{1+\zeta}(n)}$:

$$\lim_{n\to\infty} a^k = \lim_{n\to\infty} \left(1 - \frac{2w\ln(n)}{n}\right)^{\frac{n}{\ln^{1+\zeta}(n)}} = 1 \tag{4.88}$$

The Harmonic terms are of order $\ln(n)$ at most, so the first term dominates if $w\alpha > 2$.

Now we will show that the term $r'(k) \geq 0$ for all k in the interval $\left[\frac{n}{\ln^{1+\zeta}(n)}, z\frac{n}{\ln(n)}\right]$. We calculate:

$$r'(k) = a^k(-2\alpha w + 2)\ln(n)\ln(a) + H^{(2)}_{n-k} + H^{(2)}_k - \frac{\pi^2}{3} \tag{4.89}$$

By Equation (4.84):

$$r'(k) \geq \frac{4w^2\alpha \ln^2(n)}{n}(a)^k + H^{(2)}_{n-k} + H^{(2)}_k - \frac{\pi^2}{3} \tag{4.90}$$

Now we can apply Fact 4.11, and the fact that $\frac{1}{n-k} \leq \frac{1}{k}$ to obtain:

$$r'(k) \geq \frac{4w^2\alpha \ln^2(n)}{n}\left(1 - \frac{2w\ln(n)}{n}\right)^k - \frac{4}{k} \tag{4.91}$$

By definition of our interval, we obtain:

$$k \geq \frac{n}{\ln^{1+\zeta}(n)} \Rightarrow -\frac{1}{k} \geq -\frac{\ln^{1+\zeta}(n)}{n} \tag{4.92}$$

For large enough n, we have:

$$k \leq z\frac{n}{\ln(n)} \Rightarrow \left(1 - \frac{2w\ln(n)}{n}\right)^k \geq e^{-2wz-1} \tag{4.93}$$

Hence,

$$r'(k) \geq \frac{4w^2\alpha e^{-2wz-1}\ln^2(n)}{n} - \frac{4\ln^{1+\zeta}(n)}{n} > 0 \tag{4.94}$$

Once we demonstrate that $r\left(\frac{zn}{\ln(n)}\right) < 0$, we will have shown that $r(k) < 0$ for all $k \in \left[1, \frac{zn}{\ln(n)}\right]$. Indeed:

$$r\left(\frac{zn}{\ln(n)}\right) = (a)^{\frac{zn}{\ln(n)}}\left(-2w\alpha + 2\right)\ln(n) + H_{n-\frac{zn}{\ln(n)}} - H_{\frac{zn}{\ln(n)}} \tag{4.95}$$

It is not hard to see that:

$$H_{n-\frac{zn}{\ln(n)}} - H_{\frac{zn}{\ln(n)}} = O(\ln(\ln(n))) \tag{4.96}$$

from Fact 4.11. In addition, the term:

$$\left(1 - \frac{2w\ln(n)}{n}\right)^{\frac{zn}{\ln(n)}} = \Theta(1) \tag{4.97}$$

So, for large enough n, $r\left(\frac{zn}{\ln(n)}\right) < 0$.

We have shown that $r(k) < 0$ for all $k \in \left[1, \frac{zn}{\ln(n)}\right]$. This implies $b(k) < 0$ for all $k \in I_1$ which in turn implies that $f'(k) < 0$ for these k.

Part b

Now we will show that inside the interval $k \in I_2$ the function $f'(k)$ has exactly one zero. By

rearranging, we can see that $f'(k) = 0$ if and only if

$$(H_{n-k} - H_k)\left(\frac{1}{a^k} + 1\right) = -\alpha n \ln(a) \tag{4.98}$$

We define:

$$g(k) := (H_{n-k} - H_k)\left(\frac{1}{a^k} + 1\right) \tag{4.99}$$

and calculate:

$$g'(k) = \frac{1}{a^k}\left(\ln(a)(H_k - H_{n-k}) - (1 + a^k)(\psi^{(1)}(n-k+1) + \psi^{(1)}(k+1))\right) \tag{4.100}$$

So we can lower bound:

$$\begin{aligned} a^k g'(k) &\geq \frac{2w\ln(n)}{n}(H_{n-k} - H_k) - \frac{4}{k} \\ &\geq \frac{2w\ln(n)}{n}(H_{n-k} - H_k) - \frac{4\ln(n)}{zn} > 0 \end{aligned} \tag{4.101}$$

which is positive for z large enough since

$$H_{n-k} - H_k \geq H_{n(1-\varepsilon)} - H_{\varepsilon n} \geq \ln\left[\frac{1-\varepsilon}{\varepsilon + 1/n}\right] = \Omega(1)$$

in this interval by Fact 4.11. Note that we used the fact that a^k is negligible compared to 1 in this interval. Since $g(k)$ is strictly increasing with k and the RHS of Equation (4.98) is fixed, there can be at most one place where $f'(k) = 0$. We have already shown

$$f'\left(\frac{zn}{\ln(n)}\right) < 0 \tag{4.102}$$

and it is not hard to see:

$$f'(\varepsilon n) > 0 \tag{4.103}$$

for large enough n so f' has exactly one zero in this interval.

Now back to the problem at hand, we want an upper bound on $\mathbb{E}(X)$. Our analysis

implies that we can use the largest endpoint as an upper bound on $f(k)$:

$$\begin{aligned}\mathbb{E}(X) &\leq \max\left\{\varepsilon n\binom{n}{1}\left(1-\frac{2w\ln(n)}{n}\right)^{\alpha n}, \varepsilon n\binom{n}{\varepsilon n}\left(\frac{1+a^k}{2}\right)^{\alpha n}\right\} \\ &\leq \max\left\{\frac{1}{n^{w\alpha-2}}, 2^{o(n)+(h(\varepsilon)-\alpha)n}\right\}\end{aligned} \tag{4.104}$$

By hypothesis the second term is exponentially small. To complete the proof we use Markov's inequality. Let us define the event A_1 as the event '$d_C \leq \varepsilon n$'. Then, we have:

$$\mathbb{P}(A_1) = \mathbb{P}(X \geq 1) \leq \mathbb{E}(X) = O\left(\frac{1}{n^{w\alpha-2}}\right) \tag{4.105}$$

□

4.5.5 If we Delete the Nodes K, We Still Have Linear Distance

Theorem 4.22. *Let H be a $\alpha n \times n$ binary matrix with $\alpha < 1$, where each entry is chosen uniformly at random according to i.i.d. Bern(q) where $q = \frac{w\ln(n)}{n}$. Let C be the code with H as its parity check matrix. Further, let d_C be the distance of the code C.*

Let K be the first βn bits (corresponding to the first βn columns of H) and assume that $\alpha > \beta$. Denote the code C restricted to the bits $V \setminus K$ as $C_{V\setminus K}$. If ε' satisfies:

$$(1-\beta)h\left(\frac{\varepsilon'}{1-\beta}\right) < \alpha - \beta \quad \text{and} \quad h(\varepsilon') < \alpha \tag{4.106}$$

then,

$$d_{C_{V\setminus K}} > \varepsilon' n \tag{4.107}$$

with probability at least

$$1 - O\left(\frac{1}{n^{w\alpha-2}}\right) \tag{4.108}$$

Proof. We will use the previous lemma to show that we can expect the code C to have linear distance, and condition on this event to show that the code $C_{V\setminus K}$ satisfies the same property with a weaker constant.

Again let A_1 be the event that $d_C \leq \varepsilon n$ where $\varepsilon > \varepsilon'$ and $h(\varepsilon) < \alpha$. Further, let A_2 be the event that $d_{C_{V\setminus K}} \leq \varepsilon' n$. We will use the negation symbol $\neg$ for the complement. So $\neg A_1$ is the event that $d_C > \varepsilon n$. We can write:

$$\mathbb{P}(A_2) = \mathbb{P}(A_2 \cap A_1) + \mathbb{P}(A_2 \cap \neg A_1) \tag{4.109}$$

By hypothesis $h(\varepsilon) < \alpha$, so by Lemma 4.21, we can upper bound:

$$\mathbb{P}(A_2 \cap A_1) \leq \mathbb{P}(A_1) = O\left(\frac{1}{n^{w\alpha-2}}\right) \tag{4.110}$$

Recall from the previous lemma that the code C is defined through the parity check matrix H as the set of all subsets of the columns of H which sum to zero. A 'bad' event in the current context is the existence of a set of columns which simultaneously sum to zero and has small weight outside the set K. Such an event implies the existence of a codeword which is "nearly covered up" by the erasure. We proceed by bounding the probability that such a set exists.

Let $s \subset [n]$ be some subset of the columns containing fewer than $\varepsilon' n$ many columns outside the erased set K, and let Q be the class of all sets with this property. Let us define the event B_s as 'the sum of the columns in the set s is zero' (or equivalently that the membership vector of the set s forms a word in the code). It is easy to see that:

$$A_2 \Rightarrow \bigcup_{s\in Q} B_s \tag{4.111}$$

so we have immediately that:

$$A_2 \cap \neg A_1 \Rightarrow \left(\bigcup_{s \in Q} B_s \right) \cap \neg A_1 \tag{4.112}$$

which implies the upper bound:

$$\mathbb{P}(A_2 \cap \neg A_1) \leq \mathbb{P}\left(\left(\bigcup_{s \in Q} B_s \right) \cap \neg A_1 \right) \tag{4.113}$$

Now we will compute an upper bound on $\mathbb{P}((\cup_{s \in Q} B_s) \cap \neg A_1)$. Let s be some subset with l_1 many elements in K and l_2 many elements in $V \setminus K$.

By Corollary 4.14, the probability is the same as before:

$$\mathbb{P}(B_s) = \left(\frac{1 + \left(1 - \frac{2w \ln(n)}{n}\right)^{l_1 + l_2}}{2} \right)^{\alpha n} \tag{4.114}$$

except that under our assumptions (namely that we are in the case $\neg A_1$) we have,

$$1 \leq l_2 \leq \varepsilon' n \tag{4.115}$$

and

$$(\varepsilon - \varepsilon')n \leq l_1 \tag{4.116}$$

so we have:

$$\mathbb{P}\left(B_s \cap \neg A_1\right) \leq \frac{1}{2^{\alpha n}} \left(1 + \left(1 - \frac{2w \log(n)}{n} \right)^{(\varepsilon - \varepsilon')n} \right)^{\alpha n} \leq \frac{1}{2^{\alpha n}} \left(1 + \frac{1}{n^{2w(\varepsilon - \varepsilon')}} \right)^{\alpha n} \tag{4.117}$$

Where we assumed n was very large for the final inequality. Using the union bound, we can then argue:

$$\mathbb{P}\left(\left(\bigcup_{s\in Q} B_s\right)\cap\neg A_1\right) \leq \binom{(1-\beta)n}{\varepsilon' n}\sum_{k=(\varepsilon-\varepsilon')n}^{\beta n}\binom{\beta n}{k}\frac{1}{2^{\alpha n}}\left(1+\frac{1}{n^{2w(\varepsilon-\varepsilon')}}\right)^{\alpha n} \tag{4.118}$$
$$\leq \frac{2^{o(n)+h\left(\frac{\varepsilon'}{1-\beta}\right)(1-\beta)n}}{2^{(\alpha-\beta)n}}$$

In the last step we used the standard approximation to binomial coefficients:

$$\binom{m}{\delta m} = 2^{o(m)}2^{h(\delta)m} \tag{4.119}$$

We obtain an upper bound that is exponentially small with n if

$$h\left(\frac{\varepsilon'}{1-\beta}\right)(1-\beta) < \alpha-\beta \tag{4.120}$$

□

4.5.6 Proof of Main Theorem

Theorem 4.23. *For any $0.33 < p < \frac{1}{2}$, let $R < 1-2p$. Also fix $q = \frac{w\ln(n)}{n}$ for some constant $w > 1$. Let H be a randomly sampled $(1-R)n \times n$ matrix where all H_{ij} are distributed according to i.i.d Bern(q) random variables. Denote the code with parity check matrix H as C. Let G be a randomly sampled graph where we begin with an empty graph and add each edge independently with probability q.*

In analogy to the erasure channel, suppose each column is 'erased' with probability p. I.e. we start with the empty set $K \subseteq [n]$ and add each bit independently to K with probability p. Let F be the matrix defined in Section 4.5.3 given the subset K. The probability that F is not full rank satisfies:

$$p_e = O\left(\frac{1}{n^{w-1}} + \frac{1}{n^{2(pw-1)}}\right) \tag{4.121}$$

Proof. Let K denote the set of erased bits. By the Chernoff Bound, we can assume that $|K| = p'n \in [(p-\delta')n, (p+\delta')n]$ with probability exponentially close to 1 for any $\delta' > 0$. Since the erased bits are independent of the code, we can assume without loss of generality that K consists of the first $p'n$ bits and randomly sample our code. Let us define the constant δ such that $R + \delta = 1 - 2p$. The matrix F is as defined in the previous lemma:

$$F = \overset{\overset{(1-p')n}{\longleftrightarrow}}{(R+p')n \updownarrow \left[\begin{array}{c} C_{V\setminus K} \\ A_{cut}^T \end{array}\right]} \tag{4.122}$$

where $C_{V\setminus K}$ are the generators of the code C restricted to the non-erased bits, and A_{cut} is the $|V \setminus K| \times |K|$ cut matrix across the cut $(K, V \setminus K)$. The matrix F fails to to full rank if and only if one the the following events occurs:

1. A_1- The rows of $C_{V\setminus K}$ are linearly dependent
2. A_2- The rows of A_{cut}^T are linearly dependent
3. A_3- A linear combination of the rows of A_{cut}^T produce some word in $C_{V\setminus K}$

We will produce upper bounds on the probabilities of each of these events, and use the union bound to find an upper bound on the probability that F is not full rank.

For A_1, we will argue using the randomly generated parity check matrix of the classical code. Recall that the parity check matrix H is a $[(1-R)n] \times n$ matrix such that each entry is 1 with probability q, and 0 otherwise. Denote the codewords $c \in C$ as $c = (a, b)$ where a is supported on the erased bits and b is supported on the non-erased bits. Further, let $\{(a_i, b_i)\}$ be a minimal set of generators for the code. Now observe the following equivalence: The set of vectors $\{\mathbf{b}_i\}$ is linearly dependent if and only if there is some nonzero $\mathbf{c} = (\mathbf{a}, \mathbf{b}) \in C$ with

$\mathbf{b} = 0$. Observe that, if H is full rank on the first $p'n$ bits, then there is no non-zero $c \in C$ such that $\mathbf{c} = (\mathbf{a}, \mathbf{b})$ with $\mathbf{b} = 0$. Hence, the probability $\mathbb{P}(A_1)$ is less than or equal to the probability that the first $p'n$ columns of H are linearly dependent.

The first $p'n$ columns of H correspond to a random $[(1-R)n] \times p'n = (2p+\delta)n \times p'n$ matrix where each entry is 1 with probability q. Note that, conditioned on the erased bits, we can treat this submatrix as independently sampled as in Theorem 4.12. If δ' is small compared to p, then this submatrix has more rows than columns by some constant fraction of n. Hence, we derive the following upper bound using Theorem 4.12

$$\mathbb{P}(A_1) = O\left(\frac{1}{n^{w-1}}\right) \tag{4.123}$$

We can bound $\mathbb{P}(A_2)$ directly using Theorem 4.12. A_{cut}^T is a randomly sampled $p'n \times (1-p')n$ binary matrix where each entry is 1 independently with probability $\frac{w\ln(n)}{n}$. Since $p < \frac{1}{2}$, there are more rows of A_{cut}^T than there are columns by some constant fraction of n if we take δ' small enough. Hence:

$$\mathbb{P}(A_2) = O\left(\frac{1}{n^{w-1}}\right) \tag{4.124}$$

Bounding $\mathbb{P}(A_3)$ requires Theorem 4.10. Let B_1 be the event '$d_{C_{V\setminus K}} \leq \varepsilon' n$' where $\varepsilon' = H^{-1}(p)$. We write:

$$\mathbb{P}(A_3) = \mathbb{P}(A_3 \cap B_1) + \mathbb{P}(A_3 \cap \neg B_1) \tag{4.125}$$

It is easy to check (computationally) that we have met the conditions required for Theorem 4.10 for small enough δ':

$$(1-p')h\left(\frac{h^{-1}(p)}{1-p'}\right) < 2p + \delta - p - \delta' \tag{4.126}$$

and

$$h(\varepsilon') = p < 2p + \delta \tag{4.127}$$

So, we can upper bound:

$$\mathbb{P}(A_3 \cap B_1) \leq \mathbb{P}(B_1) = O\left(\frac{1}{n^{(2p+\delta)w-2}}\right) = O\left(\frac{1}{n^{2(pw-1)}}\right) \tag{4.128}$$

Now we need to find an upper bound on $\mathbb{P}(A_3 \cap \neg B_1)$. Let $\mathbf{c}_{V\setminus K}$ be some word in $C_{V\setminus K}$ of weight g, and let $\mathbf{v}$ be any word in $\mathbb{F}_2^{p'n}$ of weight k. By Corollary 4.14, we write:

$$\frac{b(k)}{2^{(1-p')n}} := \mathbb{P}\left(A_{cut}\mathbf{v} = \mathbf{c}_{V\setminus K}\right) = \frac{1}{2^{(1-p')n}}\left(1 + \left(1 - \frac{2w\ln(n)}{n}\right)^k\right)^{(1-p')n-g} \cdot \left(1 - \left(1 - \frac{2w\ln(n)}{n}\right)^k\right)^g \tag{4.129}$$

It is sufficient for an upper bound to analyze the word $\mathbf{c}_{V\setminus K}$ of smallest weight, since clearly this expression is decreasing with g increasing. Since $d_{C_{V\setminus K}} > \varepsilon' n$ we can assume $g = \varepsilon' n$. Let us define the intervals $I_1 = \left[1, \frac{zn}{\ln(n)}\right]$ and $I_2 = \left[\frac{zn}{\ln(n)}, p'n\right]$ for some large constant z to be determined. For fixed g, we will first provide an upper bound for this function inside the interval I_1. Let us analyze the derivative of $b(k)$ with respect to k:

$$b'(k) = \left[1 + \left(1 - \frac{2w\ln(n)}{n}\right)^k\right]^{(1-p')n-g-1}\left[1 + \left(1 - \frac{2w\ln(n)}{n}\right)^k\right]^{g-1}\ln\left(1 - \frac{2w\ln(n)}{n}\right) \times \left\{-\left(1 + \left(1 - \frac{2w\ln(n)}{n}\right)^k\right)g + ((1-p')n - g)\left(1 - \left(1 - \frac{2w\ln(n)}{n}\right)^k\right)\right\} \tag{4.130}$$

We can set this expression equal to zero and solve. We obtain:

$$k^1_{max} = \frac{\ln\left(\frac{(1-p')n-2g}{(1-p')n}\right)}{\ln\left(1 - \frac{2w\ln(n)}{n}\right)} = \frac{\ln\left(1 - \frac{2\varepsilon'}{1-p'}\right)}{\ln\left(1 - \frac{2w\ln(n)}{n}\right)} \tag{4.131}$$

The function $b'(k)$ is peaked around k^1_{max}, or $b'(k) \geq 0$ for $k \leq k^1_{max}$ and $b'(k) \leq 0$ for

$k \geq k^1_{max}$. Expanding the expression for k^1_{max}, we can see that $k^1_{max} \in I_1$, so $b(k^1_{max})$ provides an upper bound for $b(k)$ in this interval. We calculate:

$$\frac{b(k^1_{max})}{2^{(1-p')n}} = \frac{1}{2^{(1-p')n}} \left[2 - 2\frac{\varepsilon'}{1-p'}\right]^{(1-p')n-\varepsilon' n} \left[\frac{2\varepsilon'}{1-p'}\right]^{\varepsilon' n} \tag{4.132}$$
$$= 2^{\wedge} \left[-\varepsilon' n + \log\left(1 - \frac{\varepsilon'}{1-p'}\right)((1-p')n - \varepsilon' n) + \log\left(\frac{2\varepsilon'}{1-p'}\right)\varepsilon' n\right]$$

In the second interval I_2, the best upper bound we can obtain is:

$$\frac{b\left(\frac{zn}{\ln(n)}\right)}{2^{(1-p')n}} \leq \frac{1}{2^{(1-p')n}} \left[1 + \left(1 - \frac{2w\ln(n)}{n}\right)^{\frac{zn}{\ln(n)}}\right]^{(1-p')n} \tag{4.133}$$

For large enough n,

$$\leq \frac{1}{2^{(1-p')n}} \left[1 + e^{-2wz}\right]^{(1-p')n} =: \frac{b(k^2_{max})}{2^{(1-p')n}} \tag{4.134}$$

We are interested in finding an upper bound for the probability that any $\mathbf{v} \in \mathbb{F}_2^{p'n}$ maps to any $\mathbf{c}_{V\setminus K} \in C_{V\setminus K}$ under A_{cut}. For this we can employ the union bound:

$$\mathbb{P}(\exists \mathbf{v} \in \mathbb{F}_2^{p'n}, \exists \mathbf{c}_{V\setminus K} \in C_{V\setminus K} : A_{cut}\mathbf{v} = \mathbf{c}_{V\setminus K} \cap \neg B_1) \leq \sum_{\substack{\mathbf{v}\in\mathbb{F}_2^{p'n} \\ \mathbf{c}_{V\setminus K}\in C_{V\setminus K}}} \mathbb{P}[A_{cut}\mathbf{v} = \mathbf{c}_{V\setminus K} \cap \neg B_1]$$

Under our assumptions, $\mathbf{c}_{V\setminus K} > \varepsilon' n$, we can further upper bound this expression by:

$$\leq 2^{Rn} \left(\sum_{|v|\in I_1} \frac{b(k^1_{max})}{2^{(1-p')n}} + \sum_{|v|\in I_2} \frac{b(k^2_{max})}{2^{(1-p')n}}\right) \tag{4.135}$$

where we used the fact that the code $C_{V\setminus K}$ has at most 2^{Rn} many words. We then note that there are at most $2^{o(n)}$ many terms in the first sum. The upper bound we obtain is:

$$\frac{2^{Rn}2^{o(n)}b(k^1_{max})}{2^{(1-p')n}} + \frac{2^{Rn}2^{p'n}}{2^{(1-p')n}} \left[1 + e^{-2wz}\right]^{(1-p')n} \tag{4.136}$$

For large enough z and small enough δ' the second term is exponentially small with n. The first term is exponentially small if:

$$R - \varepsilon' + \log\left(1 - \frac{\varepsilon'}{1-p'}\right)((1-p') - \varepsilon') + \log\left(\frac{2\varepsilon'}{1-p'}\right)\varepsilon' < 0 \tag{4.137}$$

If

$$g(p) := 1 - 2p - \varepsilon' + \log\left(1 - \frac{\varepsilon'}{1-p}\right)((1-p) - \varepsilon') + \log\left(\frac{2\varepsilon'}{1-p}\right)\varepsilon' < 0 \tag{4.138}$$

Then we can make δ' small enough that Equation (4.137) holds. We have found computationally $g(p) < 0$ for $0.33 < p < \frac{1}{2}$ (recall that we set $\varepsilon' = h^{-1}(p)$). □

4.5.7 Size Bounds on the code (G, C)

Now to provide a probabilistic estimate on the weight of the randomly chosen stabilizer code. For $i \in [1, \ldots (1-R)n]$, define the random variable X_i to be the weight of the ith row of the parity check matrix H. By Chernoff, for each i

$$\mathbb{P}(X_i \geq \ln^{1+\zeta}(n)) \leq \frac{\mathbb{E}(e^{t \cdot X_i})}{e^{t \cdot a}} = \frac{e^{n\left(\frac{w\ln(n)}{n}\right)(e^t - 1)}}{e^{t\ln^{1+\zeta}(n)}} = \frac{n^{w(e^t-1)}}{n^{t\ln^{\zeta}(n)}} \tag{4.139}$$

So, we can calculate via the union bound:

$$\mathbb{P}\left(\cup_i \{X_i \geq \ln^{1+\zeta}(n)\}\right) \leq \frac{ne^{n\left(\frac{w\ln(n)}{n}\right)(e^t-1)}}{e^{t\ln^{1+\zeta}(n)}} = \frac{n^{w(e^t-1)+1}}{n^{t\ln^{\zeta}(n)}} \tag{4.140}$$

Now for $i \in [1, 2, \ldots n]$ define the random variable Y_i to be the number of neighbors of a vertex i in the randomly generated graph G. The same analysis yields:

$$\mathbb{P}\left(\cup_i \{Y_i \geq \ln^{1+\zeta}(n)\}\right) \leq \frac{n^{w(e^t-1)+1}}{n^{t\ln^{\zeta}(n)}} \tag{4.141}$$

If all X_i and Y_i are less than $\ln^{1+\zeta}(n)$, then the maximum weight of a generator of the

stabilizer quantum code is upper bounded by $\ln^{2+2\zeta}(n)$.

$$\mathbb{P}(\text{code } (G,C) \text{ is a } [[n,k,d,j]] \text{ code with } j \geq \ln^{2+2\zeta}(n)) \leq \frac{2n^{w(e^t-1)+1}}{n^{t\ln^{\zeta}(n)}} \tag{4.142}$$

If we take $t = O(1)$ we obtain vanishing probability with n for any $\zeta > 0$.

4.6 Conclusions and Further Directions

In this chapter we have established some results on random constructions for classical and quantum codes. We have shown that classical codes with random 'log-sparse' parity check matrices have code distance as good as uniformly random classical codes. It is possible that these codes have efficient decoding algorithms, although it is not clear that the iterative decoding procedure for LDPC codes will work. The main result is that our quantum code construction achieves the capacity of the erasure channel while only being poly-log local.

There are many open questions here. I realized while compiling this thesis that Gallager's regular LDPC codes can probably be used instead of the log-sparse codes we sample in the document. It is known that they have distance meeting the GV bound [72], and I suspect modifications to Theorem 4.10 will yield the same result with these even sparser codes. If these codes are sufficient, then we achieve the same result with checks of size $O(\ln(n))$ rather than $O(\ln^{2+\varepsilon}(n))$. Our construction still has diverging block length for every erasure probability. An obvious next step would be to look for capacity achieving quantum codes whose locality scales inversely with the gap to capacity, i.e. like the Tornado sequence. To find quantum codes of this form, however, different methods are needed than those presented in this chapter. As discussed in the introduction, graph codes with a graph of constant degree

must have at least constant probability of failure under erasure. Another open question is the study of decoding algorithms for the codes described. What I have presented implies efficient decoding using the standard approaches: measure the stabilizers and act on the quantum state with some Pauli operator in the same syndrome. However, it is possible this process for our code is more efficient, or more parallelizeable than a standard quantum error correcting code. One further open problem is the study of the effect of de-polarizing errors on our ensemble. This seems to be the biggest open question among these questions, however. The ideas we present here may not necessarily be useful for this case, and the combinatorics for this question seem to be much more difficult.

Chapter 5

$O(n \log(n))$ Scramblers from Erdos-Renyi Graph States

5.1 Some Fixed Notation

We will fix the following notation throughout the chapter. Given two distributions, we will denote the statistical distance as $\Delta(P, Q)$. We will be considering ensembles of Clifford operators $\mathcal{E} = \{p_k, C_k\}$. We denote

$$\mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_y\right] = \mathbb{P}_{U\sim\mathcal{E}}\left[UP_xU^\dagger \propto P_y\right] \tag{5.1}$$

as the probability that Pauli operator P_x conjugates to Pauli operator P_y under ensemble $\mathcal{E}$ ignoring phase. We will use the notation $\mathcal{Q}_{P_x}$ as the corresponding measure:

$$\mathcal{Q}_{P_x}[P_y] = \mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_y\right] \tag{5.2}$$

We will have need to condition on certain events, so we will define a conditioned measure as well. Let B be some event, which should be obvious from the context. We will define:

$$\widetilde{Q}_{P_x}[P_y] = \mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_y \middle| B\right] \tag{5.3}$$

as the probability that P_x conjugates to P_y assuming the event B.

Lastly, in order to work conveniently with conjugation relations we will occasionally denote P_y using its X and Z strings $P_y = X_{\mathbf{q}_1}Z_{\mathbf{q}_2}$. If we are viewing P_y as a random variable, we will denote it with $(\mathbf{Q}_1, \mathbf{Q}_2)$ where each string is some random variable over $\mathbb{F}_2^n$.

5.2 Sampling from the sphere

There are many classical contexts in which it is of interest to sample unit vectors or rotations randomly on the unit sphere [8, 22, 39, 65]. Machine learning [22], Monte Carlo simulations [39], and even numerical integration [65] can use uniform random rotations to accomplish some task quickly and easily. More generally, the field of Pseudorandomness seeks to provide

distributions that "look" uniform over some large set, but in fact require very few coin tosses to generate [13]. There are many problems that can be solved faster using randomness [4,136], and pseudorandom constructions often allow drastic resource savings for solutions to these problems.

Quantum mechanics has it's own "built in" sources of randomness, so in quantum computing generating random coin flips is not of much interest. The interesting problem here is to sample uniform random vectors from the complex sphere, or to sample uniform random rotations of the sphere. Here the problem is that producing a uniform random rotation or state is completely intractable for an exponentially large Hilbert space. For a single qubit, it is easy to sample a random state or unitary. Using the Bloch sphere [123] (Figure 5.1), we can associate quantum states to points on the sphere in $\mathbb{R}^3$. A general quantum state can be written as $|\psi\rangle = \cos(\theta/2)\,|0\rangle + e^{i\phi}\sin(\theta/2)\,|1\rangle$. We can associate this to the point on the unit sphere with spherical angles (θ, ϕ) (Figure 5.1). A random quantum state is then just a random point on the unit sphere, and a random unitary is is a random rotation of the sphere. This corresponds to two and three independent parameters respectively (a random rotation can be found by sampling the three Euler angles).

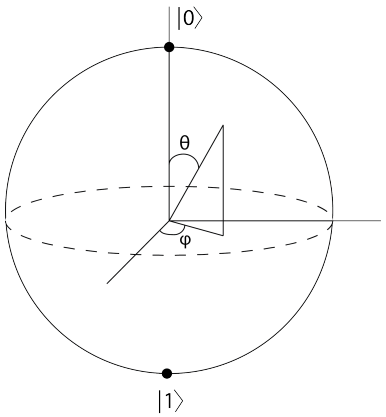


Figure 5.1: Bloch Sphere

The problem is that a randomly sampled vector from an exponentially large space will require an exponential number of random parameters and hence an exponentially large quantum circuit. For this reason, the notion of a design was developed [7]. A t design is an ensemble of states $\{p_k, |\psi_k\rangle\}$ or unitaries $\{p_k, U_k\}$ that "mimics" sampling Haar random states of unitaries in some well defined way. For the qubit case, a state design can be thought of as a discrete set of points with an efficient sampling algorithm that closely resembles a random sample to the sphere:

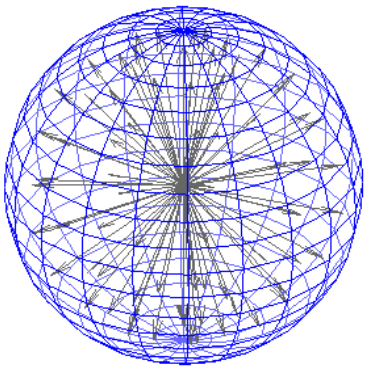


Figure 5.2: Illustration of a Design for a Single qubit

The paramter t is used to denote the "order" of approximation. Large t implies that the ensemble matches the Haar random case up to some very high order.

While these ensembles are only approximations to random unitaries they have found use in many important applications inside quantum information theory [32, 33, 45, 58, 62, 87, 88, 104, 118, 156, 160]. Decoupling is the use of random unitaries to "de-correlate" quantum systems from each other [32, 58, 87, 156]. Decoupling theorems have important applications in quantum coding theory, since they point to conditions under which good quantum codes can be constructed. To provide some intuition, consider the following picture:

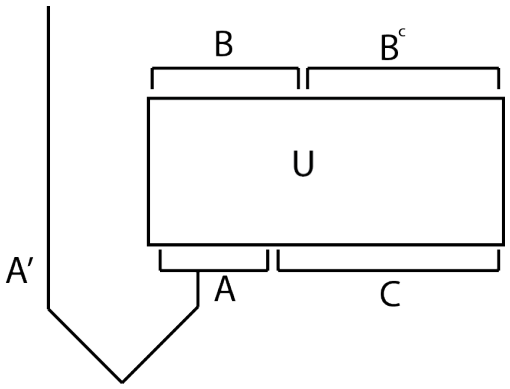


Figure 5.3: Information in A is scrambled, and hence not recoverable from B

Suppose we have a maximally entangled state $|\psi\rangle_{A,A'} = \sum_i |i\rangle_A |i\rangle_{A'}$ and we send half of it through the circuit U. The other party (Bob) has access to some portion of the output of our circuit, in the figure we would say they have access to the subsystem B. Our goal is to exchange entanglement so that we can accomplish some quantum operation, say teleportation or entanglement-assisted communication. If Bob has access to the entire output of the circuit, then he can simply invert it to obtain the original input (if he knows what the circuit was). If, however, Bob has access to only some small portion of the output he will be unable to access the input for most circuits U [126]. We say in this case that the subsystems A, A' have been decoupled of that information in A' has been scrambled. This general problem has found important applications in describing black holes [88, 110], and in quantum coding theory [32, 87]. The above picture makes this clear: if we think about "Bob" as the environment, scrambling behavior implies the environment needs access to a large number of the susbsytems to get any information about the input.

For this chapter, the relevant application for the construcion is "Randomized Benchmarking" [33, 45, 62, 104, 118, 160]. Suppose we have some set of noisy quantum operations $\mathcal{G} = \{U_1, ..., U_m\}$ that we are trying to use in a quantum computer. Each of these operations

is meant to imitate some unitary operation, U_i, but each one comes with some noise channel $\mathcal{E}_i$ which prevents an exact implementation. Assuming the noise is Markovian [33], we can effectively implement the map:

$$\hat{U}_i(\rho) := \mathcal{E}_i(U_i \rho U_i^\dagger) \tag{5.4}$$

We could equally well have assumed the noise happened *before* the clean unitary, and obtained a different (rotated) noise map $\mathcal{E}'_i$:

$$\hat{U}_i(\rho) := \mathcal{E}_i(U_i \rho U_i^\dagger) = U_i \mathcal{E}'_i(\rho) U_i^\dagger \tag{5.5}$$

Assuming the set $\mathcal{G}$ is closed under inverse, we can define the "average noise channel" by sampling one of the operations at random, applying it, and applying it's inverse:

$$\Lambda(\rho) = \frac{1}{m} \sum_{i=1}^{m} \hat{U}_i^\dagger \left(\hat{U}_i(\rho) \right) \tag{5.6}$$

If each of the operations works perfectly (all of the noise channels $\mathcal{E}_i$ are the identity), then the channel Λ acts as the identity on all inputs. If the channel is far from the identity (say in diamond norm), then these operations are very noisy and probably unsuitable for a quantum computation. Randomized benchmarking tries to efficiently measure the strength of this channel in order to determine the amount of noise present "on average". The average noise channel can be written as

$$\Lambda(\rho) = \frac{1}{m} \sum_i U_i^\dagger \mathcal{E}'_i \left(\mathcal{E}_i \left(U_i \rho U_i^\dagger \right) \right) U_i \tag{5.7}$$

Assuming further that the error channel has very weak gate dependence, we can replace each map $\mathcal{E}'_i \circ \mathcal{E}_i$ with some other map Λ' (independent of i):

$$\Lambda(\rho) \approx \frac{1}{m} \sum_i U_i^\dagger \Lambda'(U_i \rho U_i^\dagger) U_i \tag{5.8}$$

The "strength" of the noise map is defined as the average fidelity of the map Λ':

$$F(\Lambda') := Tr\left[\rho \int U^\dagger \Lambda'(U\rho U^\dagger) U dU\right] \tag{5.9}$$

where the integral is taken over the Haar measure [42] and ρ is some pure state. Unitary designs are useful in this context because they provide ensembles $\mathcal{G}$ which allow one to determine Equation (5.9) without resorting to random Haar sampling. A design is an ensemble for which $\sum_i (1/m) U_i^\dagger \Lambda'(U_i \rho U_i^\dagger) U_i \approx \int U^\dagger \Lambda'(U\rho U^\dagger) U dU$ for all input states ρ and all maps Λ'.

5.3 Unitary 2-Designs

There are many ways to formalize our notion of "approximation" to a Haar random unitary [47,112]. We will discuss only two, but depending on the context or task at hand there could be many different desirable ones. Unitary 2-Designs come in two types, exact 2-designs and approximate 2-designs. Approximate designs are ensembles that match the exact definitions up to some error ε, normally quantified using the diamond norm. Let us first give the most intuitive notion of a unitary t-design. Define the expected operators:

$$\mathbb{E}^t_{U\sim Haar}\left[U^{\otimes t}\rho(U^\dagger)^{\otimes t}\right] = \int U^{\otimes t}\rho(U^\dagger)^{\otimes t} dU \quad \text{and} \quad \mathbb{E}^t_{U\sim\mathcal{E}}\left[U^{\otimes t}\rho(U^\dagger)^{\otimes t}\right] = \sum_k p_k U_k^{\otimes t}\rho(U_k^\dagger)^{\otimes t}$$

where the first integral is calculated according to the Haar measure [42]. We will consider each of these as quantum (CPTP) maps.

Definition 5.1 (ε-Approximate Diamond t-design)**.** *Let $\mathcal{E} = \{p_k, U_k\}$ be any ensemble of unitaries on n qubits. We say that $\mathcal{E}$ is an ε-Approximate Diamond t-design if $||\mathbb{E}^t_{U\sim Haar} - \mathbb{E}^t_{U\sim\mathcal{E}}||_\diamond < \varepsilon$.*

We can obtain the exact definition by setting ε equal to 0. In this case the two maps are identical for any density matrix ρ. This is the most intuitive notion of a t-design, the definition is equivalent to the statement "Observables measured on our ensemble match observables measured on the Haar random unitary ensemble if no more than t copies of the Hilbert space are used". Let M be some matrix on t copies of the Hilbert space where we think of $Tr[M_i\rho]$ as the probability of measuring some value i for an observable $\mathcal{O}$ on density matrix ρ. Define the "true" probability as

$$p = Tr\Big[M\ \mathbb{E}^t_{U\sim Haar}[U^{\otimes t}\rho(U^\dagger)^{\otimes t}]\Big] = \mathbb{E}^t_{U\sim Haar}\Big[Tr(MU^{\otimes t}\rho(U^\dagger)^{\otimes t})\Big] \tag{5.10}$$

and the "observed " probability as

$$\widetilde{p} = Tr\Big[M\ \mathbb{E}_{U\sim\mathcal{E}}[U^{\otimes t}\rho(U^\dagger)^{\otimes t}\Big] = \mathbb{E}_{U\sim\mathcal{E}}\Big[Tr(MU^{\otimes t}\rho(U^\dagger)^{\otimes t})\Big] \tag{5.11}$$

In the exact case, $p = \widetilde{p}$, while for an ε-approximate t design, $|p - \widetilde{p}| < \varepsilon$.

The next definition we will discuss is the so-called "Twirl" Definition [47]. Suppose we are given an ensemble of unitary matrices $\mathcal{E} = \{p_k, U_k\}$. Given any super-operator $\Lambda(\rho)$, we define the following two twirling operations:

$$\Phi^{\mathcal{E}}_\Lambda(\rho) := \mathbb{E}_{U\sim\mathcal{E}}\Big[U^\dagger\Lambda(U\rho U^\dagger)U\Big] \tag{5.12}$$

and

$$\Phi^{Haar}_\Lambda(\rho) := \mathbb{E}_{U\sim Haar}\Big[U^\dagger\Lambda(U\rho U^\dagger)U\Big] \tag{5.13}$$

Definition 5.2 (ε-Approximate Twirl Design)**.** *We define $\mathcal{E}$ to be a ε-approximate Twirl design if:*

$$\forall\Lambda \quad ||\Phi^{\mathcal{E}}_\Lambda - \Phi^{Haar}_\Lambda||_\diamond < \varepsilon||\Lambda||_\diamond \tag{5.14}$$

At this point, it should be clear why this definition is useful for applications in the previous section. If we treat Λ as the noise map, then using our ensemble is an effective way

to probe the average fidelity of the map. Indeed, we should be able to determine the average fidelity up to precision ε. As before, we can also define an exact version of this design by setting $\varepsilon = 0$.

Most of the t-design definitions are equivalent in the exact case. Indeed the two definitions we have exhibited so far are equivalent when $\varepsilon = 0$. We will provide this proof here, since it provides intuition about exact 2-designs that will be useful for thinking about approximate 2-designs. A t-design "coherently mixes up" input states. We can get a better understanding of the action of our ensemble by examining it's effect on general operators, not just physical density matrices. By linearity, if $\Phi_\Lambda^{\mathcal{E}}(\rho) = \Phi_\Lambda^{Haar}(\rho)$ then $\Phi_\Lambda^{\mathcal{E}}(P_x) = \Phi_\Lambda^{Haar}(P_x)$ where P_x is some arbitrary Pauli operator[1]. Conversely, if we are able to verify $\Phi_\Lambda^{\mathcal{E}}(P_x) = \Phi_\Lambda^{Haar}(P_x)$ for all Pauli operators then $\Phi_\Lambda^{\mathcal{E}}(\rho) = \Phi_\Lambda^{Haar}(\rho)$ by linearity and completeness of the Pauli basis. We will consider here only Clifford ensembles, so in this case P_x conjugates to some distribution over Pauli operators. It turns out that designs result in a very "uniform" spread over the Pauli operators when applied via conjugation. This makes intuitive sense, a good design should scramble operators as well as states. First, let us say an ensemble of Clifford operators $\{p_k, C_k\}$ is *pauli invariant* if pre-multiplication by any distribution of Pauli operators yields the same distribution over Clifford operators. Formally, if $\{r_j, R_j\}$ is some ensemble of Pauli operators and $\{p_k, C_k\}$ is some ensemble of Clifford operators then $\{p_k r_j, C_k R_j\} = \{p_k, C_k\}$ as ensembles (for any fixed C_k, if we add the probabilities $p_{k'} r_j$ of operators $C_{k'} R_j$ equal to C_k we get exactly p_k). Now we are in a position to state and prove the equivalence. Note that this theorem holds more generally [41], we have restricted our attention to Clifford operators because it makes the proof simpler and more enlightening.

[1] It may not be obvious here why this is the case since ρ is required to be physical. For details see [40].

Theorem 5.3 ([41]). *Let $\mathcal{E}$ be an ensemble of Clifford operators on n qubits that is Pauli invariant. Then, $\mathcal{E}$ is a Twirl design $\Leftrightarrow$ $\mathcal{E}$ is a Diamond 2-design*

Proof. Fix an ensemble $\mathcal{E} = \{p_k, U_k\}$. Let us define the Pauli propagator:

$$\mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_y\right] = \mathbb{P}_{U\sim\mathcal{E}}\left[UP_xU^\dagger \propto P_y\right] \tag{5.15}$$

Let us say ensemble $\mathcal{E}$ has Property A if the Pauli propagator satisfies:

$$\forall x, y \neq 0 \;\; \mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_y\right] = \frac{1}{4^n - 1} \tag{5.16}$$

We will demonstrate that both the Twirl and Diamond definitions are equivalent to Property A.

First examine the Twirl definition. By linearity, we can consider only super-operators of the form $\Lambda(\rho) = A\rho B$. Also by linearity, we can examine the effect of $\Phi_\Lambda^{\mathcal{E}}$ on Pauli operators. If they agree for all Pauli operators, then they agree for all operators. By [47],

$$\int dU U^\dagger AUXU^\dagger BU = \frac{Tr(AB)Tr(X)}{2^n}\frac{\mathbb{I}}{2^n} + \frac{2^n Tr(A)Tr(B) - Tr(AB)}{2^n(4^n-1)}\left(X - Tr(X)\frac{\mathbb{I}}{2^n}\right)$$

Let us substitute the same nonzero Pauli matrix for A and B and some other Pauli matrix for X:

$$\begin{aligned} \forall y \neq 0 \quad \int dU U^\dagger P_x U P_y U^\dagger P_x^\dagger U &= -\frac{1}{4^n-1}P_y \\ \int dU U^\dagger P_x U \mathbb{I} U^\dagger P_x^\dagger U &= \mathbb{I} \end{aligned} \tag{5.17}$$

Now examine our ensemble $\mathcal{E}$:

$$\sum_k p_k U_k^\dagger P_x U_k P_y U_k^\dagger P_x^\dagger U_k$$

Any potential phase resulting from conjugation cancels out, since

$$U_k^\dagger P_x U_k = e^{i\phi}P_z \Rightarrow U_k^\dagger P_x^\dagger U_k = e^{-i\phi}P_z \tag{5.18}$$

Hence we can rewrite this expression using the propagator:

$$\sum_k p_k U_k^\dagger P_x U_k P_y U_k^\dagger P_x^\dagger U_k = \sum_z \mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_z\right] P_z P_y P_z^\dagger = \sum_z \mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_z\right] (-1)^{\langle P_y, P_z\rangle} P_y \quad (5.19)$$

For each P_x, we have obtained 4^n equations, one for each possible Pauli matrix P_y. Setting Equation (5.17) equal to Equation (5.19), we obtain:

$$\forall y \neq 0, \quad -\frac{1}{4^n - 1} = \sum_z \mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_z\right] (-1)^{\langle P_y, P_z\rangle}$$

$$1 = \sum_z \mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_z\right]$$

We can put these equations into a $4^n \times 4^n$ matrix. It is easy to see that the matrix is full rank over the reals since any Pauli operator commutes with exactly half of the other Pauli operators. This implies the dot product of any pair of rows is zero (over $\mathbb{R}$), and any row with itself is 4^n. We can observe that Property A is a solution to the equations and hence must be the only solution. We have demonstrated at this point that Property A is equivalent to a Twirl design.

Now to prove the second notion of design, a Diamond 2-design is also equivalent to this Property A. The same kind of analysis holds here, this definition extends by linearity to act on any Pauli operator. By [83],

$$\int U \otimes U \Big(P_x \otimes P_y\Big) U^\dagger \otimes U^\dagger dU = \delta_{xy} \frac{1}{4^n - 1} \sum_z P_z \otimes P_z \quad (5.20)$$

For our ensemble we have:

$$\sum_k p_k U_k \otimes U_k \Big(P_x \otimes P_x\Big) U_k^\dagger \otimes U_k^\dagger = \sum_z \mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_z\right] P_z \otimes P_z \quad (5.21)$$

Uniformity of the distribution can be checked with the observation $Tr(P_x P_y) = \delta_{xy}$. So far we have shown only one way implication, namely that Diamond design $\Rightarrow$ Property A. To show the other way we need to use our Pauli invariant assumption. By hypothesis, we can

add a uniform Pauli matrix before the ensemble without changing the expected operator. Hence,

$$\sum_k p_k U_k \otimes U_k \Big(P_x \otimes P_y \Big) U_k^\dagger \otimes U_k^\dagger = \frac{1}{4^n} \sum_{k,j} p_k U_k R_j \otimes U_k R_j \Big(P_x \otimes P_y \Big) R_j^\dagger U_k^\dagger \otimes R_j^\dagger U_k^\dagger \quad (5.22)$$

If x and y are different, then there exists some $R_q \in \mathcal{P}_n$ such that P_x commutes with R_q, but P_y anti-commutes. Again using Pauli invariance,

$$\begin{aligned} &\frac{1}{4^n} \sum_{k,j} U_k R_j \otimes U_k R_j \Big(P_x \otimes P_y \Big) R_j^\dagger U_k^\dagger \otimes R_j^\dagger U_k^\dagger \\ = &\frac{1}{4^n} \sum_{k,j} U_k R_j R_q \otimes U_k R_j R_q \Big(P_x \otimes P_y \Big) R_q^\dagger R_j^\dagger U_k^\dagger \otimes R_q^\dagger R_j^\dagger U_k^\dagger \\ = &-\frac{1}{4^n} \sum_{k,j} U_k R_j \otimes U_k R_j \Big(P_x \otimes P_y \Big) R_j^\dagger U_k^\dagger \otimes R_j^\dagger U_k^\dagger \end{aligned}$$

This demonstrates that Pauli invariant + Property A $\Rightarrow$ Diamond design. □

The above proof shows that if we have some Clifford ensemble which forms a 2-design, then it must result in a *totally uniform* distribution over Pauli operators when we conjugate any fixed Pauli operator. It is natural to ask, if the ensemble $\mathcal{E}$ is only "approximately mixing" on Pauli operators, does it form an approximate design? The answer is yes, depending on how well it mixes and the desired definition of a design. The definition becomes important because in the approximate case, the definitions are no longer equivalent in a strict sense. It holds that a ε-Twirl design is a $poly(2^n)\varepsilon$-Diamond design, and that a a ε-Diamond design is a $poly(2^n)\varepsilon$-Twirl design [112], but for ε at least $\omega(1/2^n)$ these do not lead to meaningful implications. In our case, ε will be inverse logarithmic with the dimension of the space (or inverse polynomial with the number of qubits) so a proof according to one definition does not translate to a proof according to another.

It seems, however, that our our construction is not strong enough to be a Diamond-design. Given some P_x and some ensemble $\mathcal{E}$, let $\mathcal{Q}$ be the measure corresponding to the distribution induced by conjugation:

$$\mathcal{Q}_{P_x}[P_y] = \mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_y\right] = \mathbb{P}_{U\sim\mathcal{E}}\left[UP_xU^\dagger \propto P_y\right] \tag{5.23}$$

and let $\mathcal{U}$ be a measure on the Pauli group that is uniform over the non-identity Pauli terms and 0 on $\mathbb{I}$.

We establish that $\Delta(\mathcal{Q}_{P_x},\mathcal{U}) = 1/poly(n)$, or that the statistical distance between conjugation by our ensemble and the uniform distribution (Property A) is only inverse polynomial in the number of qubits. Attempting to apply this in any of the standard proofs [83] for diamond designs results in trivial designs with $\varepsilon \geq 1$.

Before proceeding to a statement of results and core ideas we will give one more definition of "scrambling" It corresponds to an ensemble which conjugates local observables to non-local observables with high probability, much like "out-of-time-order" correlators [93]. This definition is implicit in several works [32, 83]. Showing that a particular model of random unitaries conjugates all small Pauli operators to large Pauli operators is a common first step to showing mixing. We will denote it here as "weak" scrambling since in the context of our results it will be a very weak definition. However, for certain parameter ranges the following definition could correspond to a very strong notion of scrambling (see Proposition 5.5).

Definition 5.4 ((α,ε) Weak Scrambler)**.** *Let $\mathcal{E} = \{p_k, U_k\}$ be some random ensemble where each $U_k \in \mathcal{C}\ell_n$. We say $\mathcal{E}$ is a (α,ε) Weak Scrambler if all Pauli matrices are conjugated to subsystems of size at least αn with all but probability ε.*

$$\forall P \in \mathcal{P}_n \quad \mathbb{P}_{U\sim\mathcal{E}}\left[|UPU^\dagger| < \alpha n\right] < \varepsilon \tag{5.24}$$

This definition is weak or strong depending on the value of ε. If ε is very small (at least $2^{-(1+c)n}$ with $c > 0$) then this definition corresponds to strong scrambling, since the circuit U will be an encoding circuit for a "good" quantum code (one with linear distance) with high probability. Note that a good code scrambles observables to a subsystem of size at least the distance of the code. If ε is not small enough, say it is larger than 2^{-n}, it is not clear that it will scramble a single observable (modulo the stabilizers of the code)! Our construction achieves only $\varepsilon = 1/poly(n)$, but I suspect if we condition on the circuit we can obtain much better parameters.

Proposition 5.5. *Let $\mathcal{E} = \{p_k, U_k\}$ be some (α, ε) weak scrambler on n qubits with $\varepsilon = 2^{-(1+c)n}$ with $c > 0$ and $\alpha = \Omega(n)$. With high probability, a randomly sampled U_k will act as an encoding circuit for an asymptotically "good" quantum code.*

Proof. Consider the action of the randomly chosen unitary on k message qubits, while setting the rest of the qubits equal to $|0\rangle$:

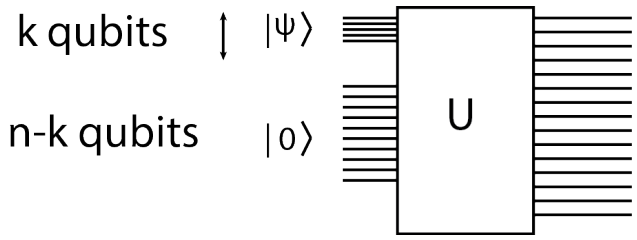


Figure 5.4: Weak Scramblers are Error Correcting codes for ε small enough.

Before the circuit, the stabilizers for the code are generated by $\{Z_i\}_{i=k+1}^{n}$ all Z operators not on the first k qubits. The logical operators consist of all Pauli operators supported on the first k qubits. After the circuit, the logical operators consist of conjugated logical operators multiplied by some stabilizer. Hence, on the other end of the circuit there are at most $2^{n-k}4^k$

non-trivial logical operators. Each of these inhabit a subsystem of size less than αn with probability less than ε. By the union bound, the probability of any small logical operator is at most $2^{n-k}4^k\varepsilon = 2^{n+k}2^{-n(1+c)} = 2^{k-cn}$. For small enough k the distance of the code will be at least αn with all but exponentially small probability. □

5.4 Statement of Results and Core Ideas

The core ideas of this design can best be motivated by an analogy. Let A be a symmetric, random $n \times n$ binary matrix with each off-diagonal A_{ij} independent and distributed according to a Bernoulli random variable with probability $w\ln(n)/n$ of being 1. Further, let each diagonal entry A_{ii} be 1 with probability $1/2$ and 0 otherwise. Let $\mathbf{e}$ be some binary vector and suppose we are interested in showing the distribution of $\mathbf{q} = A\mathbf{e}$ is 'nearly' uniform over $\mathbb{F}_2^n$ for some $\mathbf{e}$. If $\mathbf{e}$ has very small Hamming weight, say $O_n(1)$, then clearly the distribution of $A\mathbf{e}$ is not uniform since the weight of $\mathbf{q}$ will be logarithmic with high probability. If, however, $\mathbf{e}$ has large weight, say $\Omega(n)$, then the distribution of $\mathbf{q}$ will be nearly uniform up to statistical distance $1/poly(n)$. The probability is:

$$\mathbb{P}(\mathbf{q} = A\mathbf{e}) = \left(\frac{1}{2}\right)^{|\mathbf{e}|}\left(\frac{1-(1-2p)^{|\mathbf{e}|}}{2}\right)^{a_2}\left(\frac{1+(1-2p)^{|\mathbf{e}|}}{2}\right)^{a_3} \tag{5.25}$$

where a_2 is the support of $\mathbf{q}$ minus the support of $\mathbf{e}$ and $a_3 = n - |\mathbf{e}| - a_2$. Assuming w is large enough, the above is upper/lower bounded by $(1/2)^n\Big(1 \pm 1/poly(n)\Big)$.

The construction here can be understood in much the same way. Let $G = (V, E)$ be a graph and let

$$U_G := \prod_{(i,j)\in E} CP_{ij} \tag{5.26}$$

be the controlled phase operation corresponding to the edges of G. From conjugation rela-

tions of controlled phase and Hadamard (see Equation (2.28)) it is easy to verify:

$$U_G H^{\otimes n} X_{\mathbf{b}} Z_{\mathbf{c}} (H^\dagger)^{\otimes n} U_G^\dagger \;\; \propto \;\; X_{\mathbf{c}} Z_{\mathbf{b}+A\mathbf{c}} \tag{5.27}$$

where A is the adjacency matrix of G. Suppose G is is an Erdos-Renyi (ER) graph [64] with probability $w\ln(n)/n$ of edge formation. We can appeal to the above intuition which says we achieve mixing over the set of Z strings **if** the vector $\mathbf{c}$ has weight $\Omega(n)$.

Our construction is defined with this intuition in mind. It is composed of two Clifford circuits. The first is designed to increase the Pauli weight of an input string (as in the weak scrambler definition) probabilistically and the second component is a few copies of these ER controlled phase circuits designed to scramble large weight Paulis. For the following, one should recall the specific Clifford Unitaries from Chapter 2. We can write the procedure as follows:

Phase 1 ($\mathcal{E}_1$) Apply a uniform random Pauli operator.
Phase 2 ($\mathcal{E}_2$) Let $\{G_n\}_{n=0}^{\infty}$ be an infinite family of expander graphs. Repeat the following procedure $l\log(n)$ times:

1. Perform a Clifford Twirl around *each* qubit
2. Apply U_{G_n}

Phase 2.5 ($\mathcal{E}_{2.5}$) Clifford Twirl around each qubit.
Phase 3 ($\mathcal{E}_3$) Let ER_1, ER_2 and ER_3 be three independent samples to $ER\left(n, \frac{w\ln(n)}{n}\right)$. Further, let $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{e}_3$ be independent uniform samples to $\mathbb{F}_2^n$

5. Apply $H^{\otimes n}$, followed by U_{ER_1}, followed by $S_{\mathbf{e}_1}$.
6. Apply $H^{\otimes n}$, followed by U_{ER_2}, followed by $S_{\mathbf{e}_2}$.
7. Apply $H^{\otimes n}$, followed by U_{ER_3}, followed by $S_{\mathbf{e}_3}$.

The full ensemble is the composition of these operations $\mathcal{E}_3 \circ \mathcal{E}_{2.5} \circ \mathcal{E}_2 \circ \mathcal{E}_1$. We will prove separate facts about each of the operations $\mathcal{E}_i$ and combine them to show that this operation forms a Twirl design. The main technical lemma that we use is:

Lemma 5.6 ([46, 47]). *Let $\mathcal{E}$ be some Pauli invariant Clifford ensemble. Define the Pauli propagator:*

$$\mathcal{Q}_{P_x}(P_y) = \mathbb{P}\Big[P_x \to P_y\Big] = \mathbb{P}_{U\sim\mathcal{E}}\Big[UP_xU^\dagger \propto P_y\Big]$$

If

$$\forall P_x \neq \mathbb{I}\ \ \Delta(\mathcal{Q}_{P_x},\mathcal{U}) = \sum_{P_y\neq\mathbb{I}}\left|\mathbb{P}\Big[P_x \to P_y\Big] - \frac{1}{4^n-1}\right| < \varepsilon$$

Then $\mathcal{E}$ is an ε Twirl Design.

To apply the above lemma, we just need to show that the procedure above produces a distribution on Pauli matrices that is very close to uniform. The analysis proceeds by first showing that phase 2 will increase the Pauli weight of an input operator with high probability:

Lemma 5.7. *For large enough d, and l, both $O_n(1)$, $\mathcal{E}_2$ is a (β,ε) weak scrambler with $\varepsilon = 1/poly(n)$, and $\beta = 1/(2d^3)$. Denote the conjugated Pauli operator after phase 2.5 as $UP_xU^\dagger = X_{\mathbf{e}_1}Z_{\mathbf{e}_2}$, we can expect $|\mathbf{e}_2| \geq (\beta/6)n$ with all but inverse polynomial probability.*

We can then feed this into phase 3, which has the effect of scrambling large weight Paulis.

Lemma 5.8. *Let ER_1, ER_2 and ER_3 be randomly independently sampled ER graphs with probability $w\ln(n)/n$ of edge formation. Suppose $|\mathbf{e}_2| \geq \beta' n$ and let*

$$\mathcal{Q}_{\mathbf{e}_1,\mathbf{e}_2}(\mathbf{q}_1,\mathbf{q}_2) := \mathbb{P}_{U\sim\mathcal{E}_3}\Big[UX_{\mathbf{e}_1}Z_{\mathbf{e}_2}U^\dagger \propto X_{\mathbf{q}_1}Z_{\mathbf{q}_2}\Big]$$

Then, for sufficiently large $w = O_n(1)$, $\Delta(\mathcal{Q}_{\mathbf{e}_1,\mathbf{e}_2},\mathcal{U}) \leq 1/poly(n)$.

The above lemmas establish:

Theorem 5.9. *For large enough d, w, l, all $O_n(1)$, the procedure described above is an ε-Twirl design for $\varepsilon = 1/poly(n)$.*

5.4.1 Previous Work

The higher order a design is, the more highly scrambling it is, since it becomes a closer and closer approximation to the Haar random unitary. The main "jump" between a design too weak to effectively scramble and one which is highly scrambling is the transition from a 1-design to a 2-design. A 1-design effectively accomplishes no scrambling at all. We can find an example of a 1-design by constructing an ensemble that acts on its input with a random Pauli matrix:

$$\mathcal{E}(\rho) = \frac{1}{4^n} \sum_{P_x \in \mathcal{P}_n} P_x \rho P_x^\dagger$$

Observables are not scrambled at all, in fact they are exactly where we left them. If we have some observable that can be measured on the first k qubits before the map, then it can be measured by the first k qubits after the map. The only real scrambling effect here is that Bob will not be able to recover the measurement if he does not know what Pauli matrix was applied. If Bob does not know the Pauli, then the density matrix of any subsystem conditioned on his knowledge is maximally mixed. If Bob does know the Pauli then he can invert it to determine whatever information he desires. A 1-design provides "cryptographic" scrambling [31], or a quantum analog of a one time pad, even though the information stays exactly where it started.

The interesting behavior is for a 2 design. Here we already have very strong scrambling behavior. Indeed, most of the unitaries in a 2-design act as good quantum codes [87],

so observables measureable on small systems before the circuit will not be obtainable by small measurements afterward with high probability. There are many known examples of unitary 2-designs and approximate-2-designs. It is well known that random circuit drawn from universal sets of quantum gates form approximate designs [26, 32, 81, 83] and these works also estabish that random two qubit Clifford unitaries also form approximate designs. [83] establishes an ε-Twirl design for $\varepsilon = 1/poly(n)$ in $O(n\log(n))$ many gates and depth $O(\log(n))$, the same parameters we establish here. The benefit of our approach over these is that it is simpler to analyze, and consists of layers of commuting phase gates. As observed in [121], the later property might allow for easier physical implementation. The full Clifford group forms an exact 3-design, which implies it is also an exact 2-design [161]. Currently, experimentalists are working with only a handful of qubits so the full Clifford group is feasible, however sampling from the Clifford group requires $O(n^2/\log(n))$ [1] operations and will soon be out of reach.

There is a very interesting approximate 2-design discovered by Dankert et al. [47], many of the tricks employed by that paper are borrowed for this chapter. This construction consists of performing some specific random Clifford operations at each step. The effect of these operations is to randomize over the set of Pauli operators provided that the operations are repeated sufficiently many times. In order to obtain a ε-Twirl-design, [47] requires circuits of size $O(n\log(1/\varepsilon))$ and depth $O(\log(n)\log(1/\varepsilon))$. We will beat the depth by a $\log(n)$ factor with our construction for *some* values of ε.

The Nakata et al. 2-design [121] has a similar form to [47], although it requires non-Clifford gates. Here the idea is to alternate random uniform unitaries diagonal in the Z basis with unitaries diagonal in the X basis. Sampling this many independent parameters

would be intractable, hence they also provide an efficient equivalent circuit implementation. This circuit can be parallelized so that one "layer" of their scheme has $O(n\log(n))$ many gates and depth $O(\log(n))$. Depending on the particular quantum codes/architecture [27] accurate non-Clifford gates can come with prohibitive overhead. There are many other known schemes which use either measurements or non-Clifford gates [82, 120], and our scheme could be beneficial over these depending on the restrictions the user has for the design.

Currently the best known exact-2-design is due to Cleve et al [41]. They pick a subgroup of the Clifford group that retains its strong mixing properties and provide a sparse implementation for Clifford elements in this subgroup. They provably obtain exact 2 designs using $O(n\log^2(n)\log(\log(n)))$ many gates and depth $O(\log^2(n))$, and conjecture the existence of 2 designs of size $O(n\log(n)\log(\log(n))$ and depth $O(\log(n))$. We are able to beat the first set by a $\log(n)\log(\log(n))$ factor and the second set by $\log(\log(n))$ factor. Of course, we provide only approximate designs with a poor approximation factor, so the schemes may not be comparable.

5.5 Preliminaries

5.5.1 Connection Between Approximate Pauli Mixing and Twirl Designs

Dankert's thesis [46] and others [122] provide the main technical tool for this connection (lemma 5.2.4 in the first reference). We will assume throughout that we are considering an ensemble $\mathcal{E} = \{c_k r_j, C_k R_j\}$. c_k is the probability that the unitary C_k is drawn from the

ensemble. This ensemble consists of first twirling by a random element of the Pauli group, and then acting some ensemble of Clifford unitaries. So, each of the probabilities r_j are the same. For the Twirl definition the random Pauli allows us to replace Λ by some depolarizing map Λ' (Pauli Channel). This allows us to avoid picking up exponential factors when converting from trace norm to $||\cdot||_2$, as is very common when working with Diamond designs.

Lemma 5.10 ([46]). *Let $\mathcal{E}$ be an ensemble as described above and define*

$$\Phi_{\Lambda}^{\mathcal{E}}(\rho) = \sum_k c_k \sum_j r_j R_j^\dagger C_k^\dagger \Lambda(C_k R_j \rho R_j^\dagger C_k^\dagger) C_k R_j$$

Then, $\Phi_{\Lambda}^{\mathcal{E}}(\rho) = \Phi_{\Lambda'}^{\mathcal{E}'}$ where

$$\Phi_{\Lambda'}^{\mathcal{E}'}(\rho) = \sum_x p_x \sum_k c_k \left(C_k^\dagger P_x C_k\right) \rho \left(C_k^\dagger P_x^\dagger C_k\right) \quad \textit{and} \quad \Lambda'(\rho) = \sum_x p_x P_x \rho P_x^\dagger$$

Proof. We can write by definition:

$$\Phi_{\Lambda}^{\mathcal{E}}(\rho) = \sum_k c_k \sum_j r_j R_j^\dagger C_k^\dagger \Lambda(C_k R_j \rho R_j^\dagger C_k^\dagger) C_k R_j \tag{5.28}$$

Lets fix k, and substitute $A_z = \sum \alpha_{z,r} P_r^z$ where P_r^z are Pauli matrices. We obtain:

$$\propto \sum_j R_j^\dagger C_k^\dagger \sum_z \left(\sum_r \alpha_{z,r} P_r^z\right) C_k R_j \rho R_j^\dagger C_k^\dagger \left(\sum_{r'} \alpha_{z,r'}^* P_{r'}^z\right) C_k R_j \tag{5.29}$$

where we have set $P_{r'}^z = (P_{r'}^z)^\dagger$ since the Pauli basis can be chosen to be Hermitian. Now fix z, r, r' we will show that if $r \neq r'$ then the right hand side is 0. We obtain:

$$\alpha_{z,r}\alpha_{z,r'}^* \sum_j R_j^\dagger C_k^\dagger P_r^z C_k R_j \rho R_j^\dagger C_k^\dagger P_{r'}^z C_k R_j \tag{5.30}$$

Let us define $C_k R_j C_k^\dagger \propto R_j'$. We obtain:

$$\propto \sum_j C_k^\dagger R_j' P_r^z R_j' C_k \rho C_k^\dagger R_j' P_{r'}^z R_j' C_k = \sum_j (-1)^{\langle R_j', P_r^z P_{r'}^z\rangle} C_k^\dagger P_r^z C_k \rho C_k^\dagger P_{r'}^z C_k \tag{5.31}$$

Since any non-zero Pauli commutes with exactly half of the other Pauli matrices, the sum

is zero unless $P_r^z = P_{r'}^z$.

Hence, we obtain the expression:

$$\sum_j R_j^\dagger C_k^\dagger \sum_{z,r} |\alpha_{z,r}|^2 P_r^z C_k R_j \rho R_j^\dagger C_k^\dagger P_{r'}^z C_k R_j \tag{5.32}$$

Once we conjugate the R_j's through, we obtain the desired expression. □

5.5.2 Proof of Lemma 5.6

We have derived the following form for our ensemble:

$$\Phi_\Lambda^{\mathcal{E}}(\rho) = \sum_x p_x \sum_k c_k \left(C_k^\dagger P_x C_k\right) \rho \left(C_k^\dagger P_x C_k\right) \tag{5.33}$$

and for the Haar random unitary case we can derive:

$$\Phi_\Lambda^{\mathcal{H}}(\rho) = (1-p_0) \sum_k \frac{1}{4^n - 1} P_k \rho P_k^\dagger + p_0 \rho \tag{5.34}$$

It is easy to see the above expression is true with the observation that the uniform ensemble over $\mathcal{C}\ell_n$ provides a 2-design [161]. Let $\mathcal{E}'$ be this ensemble. By this fact, $\Phi_\Lambda^{\mathcal{H}}(\rho) = \Phi_\Lambda^{\mathcal{E}'}$. If we repeat the analysis from Lemma 5.10, we obtain Equation (5.34) (clearly the full Clifford group is Pauli invariant). We can then write Equation (5.33) as:

$$\Phi_\Lambda^{\mathcal{E}}(\rho) = p_0 \rho + \sum_{x \neq 0} p_x \sum_y \mathbb{P}\left[P_x \xrightarrow{\mathcal{E}} P_y\right] P_y \rho P_y \tag{5.35}$$

Now we can leverage Lemma 2.11 which implies $||\Phi_\Lambda^{\mathcal{E}} - \Phi_\Lambda^{\mathcal{H}}||_\diamond$ is at most the statistical distance between the distributions in Equation (5.33) and Equation (5.34)

$$\begin{aligned} ||\Phi_\Lambda^{\mathcal{E}} - \Phi_\Lambda^{\mathcal{H}}||_\diamond &\leq \sum_{y \neq 0} \left| \sum_{x \neq 0} p_x \mathbb{P}\left[P_x \to P_y\right] - \frac{1-p_0}{4^n - 1} \right| = \sum_{y \neq 0} \left| \sum_{x \neq 0} p_x \left(\mathbb{P}\left[P_x \to P_y\right] - \frac{1}{4^n - 1} \right) \right| \\ &\leq \sum_{x,y \neq 0} p_x \left| \mathbb{P}\left[P_x \to P_y\right] - \frac{1}{4^n - 1} \right| \leq \sum_{x \neq 0} p_x \varepsilon \leq \varepsilon \end{aligned} \tag{5.36}$$

5.5.3 Statistical Facts

We will need a modified form of the Chernoff bound, for read-k families of functions. These are binary random variables $\{Q_j\}$ that depend only on some other independent random variables $\{P_i\}$ in such a way that a single P_i does not effect too many Q_j.

Theorem 5.11 (Read-k Chernoff [74])**.** *Let $\{P_i\}_{i=1}^m$ be independent Bernoulli random variables, and let $\{Q_i\}_{i=0}^r$ be binary random variables where each Q_j is a function of some of the variables in $\{P_i\}$. Suppose that each Q_j has average value q_j, and let $q = \frac{q_1+\ldots+q_r}{r}$. If each P_i influences at most k variables Q_j then we obtain the following Chernoff-like inequalities:*

$$\mathbb{P}\left[Q_1 + \ldots + Q_r \geq (q+\varepsilon)r\right] \leq 2^{-D(q+\varepsilon||q)\cdot r/k} \tag{5.37}$$

$$\mathbb{P}\left[Q_1 + \ldots + Q_r \leq (q-\varepsilon)r\right] \leq 2^{-D(q-\varepsilon||q)\cdot r/k} \tag{5.38}$$

Chernoff bounds are useful primarily when we are examining a large number of random variables, we also need the following facts corresponding to a *small* number of random variables:

Lemma 5.12. *Let $Q = P_1 + \ldots + P_q$ where P_i are i.i.d., $\mathbb{E}[P_i] = p$, and addition is over $\mathbb{F}_2$. Then,*

$$\mathbb{E}[Q] = \frac{1-(1-2p)^q}{2} \quad \textit{and} \quad \frac{1-|1-2p|}{2} \leq \mathbb{E}[Q] \leq \frac{1+|1-2p|}{2} \tag{5.39}$$

Proof. It is easy to see that:

$$\mathbb{E}[Q] = \sum_{j=1:2:q} \binom{q}{j} (p)^j (1-p)^{q-j} \tag{5.40}$$

Define

$$Tot = \sum_{j=0}^{q} \binom{q}{j} (p)^j (1-p)^{q-j} \qquad Even = \sum_{j=0:2:q} \binom{q}{j} (p)^j (1-p)^{q-j}$$
$$Odd = \sum_{j=1:2:q} \binom{q}{j} (p)^j (1-p)^{q-j} \tag{5.41}$$

By the binomial theorem,

$$Even + Odd = Tot = 1 \quad \text{and} \quad Even - Odd = (1-2p)^q \Rightarrow Odd = \frac{1-(1-2p)^q}{2} \tag{5.42}$$

The inequalities follow immediately. □

We will not restate Chernoff here, however for easy reference we will give a statement with our parameters of interest substituted in:

Theorem 5.13. *Let $Q \sim Bin(k, 2/3)$ be a binomially distributed random variable with success probability of each trial $2/3$ and let $\theta \in (0, 1/2]$. Then, $\mathbb{P}[Q < \theta k] \leq 2^{-D(\theta||2/3)k}$ and, in particular, $\mathbb{P}[Q < \theta k] \leq \frac{1}{3}$*

Proof. The first inequality is clearly just a Chernoff bound. For the second inequality, we use Chernoff if k is large, and numerically check when k is small. Suppose $k \geq 23$. We calculate:

$$2^{-D(\theta||2/3)k} \leq 2^{-D(1/2||2/3)k} \leq 2^{-0.08k} \leq 2^{-0.08 \cdot 23} < \frac{1}{3} \tag{5.43}$$

For $1 \leq k \leq 22$, we have numerically verified the lemma. □

The following is elementary, it proves that a conditioned random variable is "close" to the original random variable if the event is likely.

Lemma 5.14. *Let Ω be some finite outcome space, and let $A = \{A_i\}$ be some partition of the outcome space. Let P_A be some measure on the subsets A_i with $P_A[A_i] = p_{A_i}$, and let X*

be the corresponding random variable (i.e. $\mathbb{P}[X = A_i] = p_{A_i}$*). Given some other measure* Q_A*, we can define the "coarse-grained" statistical distance:*

$$\Delta_A(P_A, Q_A) = \sum_i |p_{A_i} - q_{A_i}| \tag{5.44}$$

Let B *be some event with* $\mathbb{P}[B] > 0$*, and suppose we construct a conditional measure* $P'_A[A_i] = \mathbb{P}[X = A_i|B]$*. Then,*

$$\Delta_A(P_A, P'_A) \leq 2\mathbb{P}[\neg B] \tag{5.45}$$

Proof.

$$\Delta_A(P'_A, P_A) = \sum_i \left| \mathbb{P}[X = A_i|B] - \mathbb{P}[X = A_i] \right| \tag{5.46}$$

$$= \sum_i \left| \mathbb{P}[X = A_i|B] - \left(\mathbb{P}[X = A_i \cap B] + \mathbb{P}[X = A_i \cap \neg B] \right) \right| \tag{5.47}$$

$$\leq \sum_i \left| \frac{\mathbb{P}[X = A_i \cap B]}{\mathbb{P}[B]} - \mathbb{P}[X = A_i \cap B] \right| + \mathbb{P}[X = A_i \cap \neg B] \tag{5.48}$$

We can write:

$$\sum_i \mathbb{P}[X = A_i \cap \neg B] = \mathbb{P}\left[\bigcup_i (X = A_i \cap \neg B) \right] = \mathbb{P}\left[\left(\bigcup_i X = A_i \right) \cap \neg B \right] \tag{5.49}$$

The first equality follows because the events $\{A_i\}$ are disjoint, and the second from distributivity of intersection over union. The first event contains all outcomes in the outcome space, so the above is equal to $\mathbb{P}[\neg B]$. We can write:

$$\Delta_A(P_A, P'_A) \leq \mathbb{P}[\neg B] + \sum_i \mathbb{P}[X = A_i \cap B] \left| \frac{1}{\mathbb{P}[B]} - 1 \right| \tag{5.50}$$

By the same argument on the second term,

$$\Delta_A(P_A, P'_A) \leq \mathbb{P}[\neg B] + (1 - \mathbb{P}[B]) = 2\mathbb{P}[\neg B] \tag{5.51}$$

□

5.5.4 Expander Graph Facts

Recall in Chapter 3 we studied expander graphs as an example of a kind of "unembeddale" topology. Here we are interested in using their expanding properties in our construction. Expander graphs have a property called *vertex expansion* (which is essentailly equivalent to the EML definition) which implies that small subsets $S \subseteq V$ "expand well" into their neighbor sets. First, we will need to define Ramanujan graphs, which are simply d-regular expanander graphs with optimal asymptotic behavior [21, 124].

Definition 5.15 (Ramanujan Graph)**.** *Let G be a d regular graph, and denote the eigenvalues of the adjacency matrix as $d = \lambda_0 > \lambda_1 \geq \ldots \geq \lambda_{n-1}$. G is a d-regular Ramanujan graph if $\max_{|\lambda_i|<d} |\lambda_i|$ is at most $2\sqrt{d-1}$.*

Note that the $\lambda_0 > \lambda_1$ condition enforces the connectivity of the graph. The definition is stated in terms of *spectral* expansion, which is equivalent to *combinatorial* expansion. There are many results which point at such a connection, we will make use of the simplest one:

Theorem 5.16 ([157])**.** *Let G be a d-regular Ramanujan graph on n vertices, with $d \geq 21$. Then, for all sets $|S| \leq (1/d^2)n$*

$$\frac{|N_G(S)|}{|S|} \geq \frac{d}{5} \tag{5.52}$$

Proof. By [157], for all sets S:

$$\frac{|N_G(S)|}{|S|} \geq \frac{d^2}{\lambda^2 + (d^2 - \lambda^2)\frac{|S|}{n}} - 1 \tag{5.53}$$

In the above we have subtracted 1 to account for the difference in definitions of neighbor set. The neighbor set we have defined does not include the set itself, while in Tanner's definition elements of the set S could be included in the neighbor set.

For $|S| \leq (1/d^2)n$,

$$\geq \frac{d^2 - \lambda^2 - \frac{1}{d^2}(d^2 - \lambda^2)}{\lambda^2 + \frac{1}{d^2}(d^2 - \lambda^2)} \tag{5.54}$$

by hypothesis $0 \leq \lambda \leq 2\sqrt{d-1}$, so

$$\geq \frac{d^2 - 4(d-1) - \frac{1}{d^2}d^2}{4(d-1) + \frac{1}{d^2}d^2} = \frac{d^2}{4(d-1)}\left[\frac{1 - \frac{4(d-1)}{d^2} - \frac{1}{d^2}}{1 + \frac{1}{4(d-1)}}\right] \geq \frac{d}{4}\left[\frac{1 - \frac{4(d-1)}{d^2} - \frac{1}{d^2}}{1 + \frac{1}{4(d-1)}}\right] \tag{5.55}$$

It holds that

$$\frac{1 - \frac{4(d-1)}{d^2} - \frac{1}{d^2}}{1 + \frac{1}{4(d-1)}} \geq \frac{4}{5} \text{ when } d \geq 21$$

□

There are even explicit constructions for graphs that satisfy this property [114, 119]. The first such result was demonstrated by Lubotzky, Sarnack and Phillips [114], so called LPS graphs:

Theorem 5.17 ([114]). *For infinitely many d, there exist explicit d-regular infinite families $\{G_n\}_{n=1}^{\infty}$ of Ramanujan graphs. The girth of these graphs asymptotically is larger than $(4/3)\log_{d-1}(n)$.*

Throughout the chapter, we will have some underlying d-regular Ramanujan graph G on n vertices. We will denote the girth of G as g or as $\alpha \log(n)$, and we will assume all sets of size at most βn have expansion at least $\frac{|N_G(S)|}{|S|} \geq d/5$.

Lemma 5.18. *Let S be a subset of points in a d-regular graph G with girth g. If $|S| < \frac{g}{2d}$, then S has at least $(d-2)|S|$ many unique neighbors.*

Proof. Let us define the sets $B = \{v \in V \setminus S : N_G(v) \cap S \geq 2\}$ and $S' = S \cup B$. It holds that $|B| \leq |S|$: if $|S| < |B|$ then consider the subgraph induced by $S \cup B$. It must be a forest

since it is smaller than the girth, but it must contain at least $2|B| > |B| + |S|$ edges. Since a forest on n vertices has at most $n-1$ edges we reach a contradiction.

It follows that $|S' \cup N_G(S')| \leq 2d|S|$, hence $G[S' \cup N_G(S')]$ is also a forest. There are at least $(d-1)|S'|$ total edges in $G[S' \cup N_G(S')]$, and at most $|S'|-1$ edges in $G[S']$. It follows that there are at least $(d-2)|S'|$ "outgoing" edges from S' (edges between S' and $N_G(S')$). Consider such an edge and assume it is connected to a point in S. Then, by definition it must be a unique neighbor of S, since otherwise it would be included in B.

There are at most $(d-2)|B|$ outgoing edges attached to a vertex in B, which leaves at least $(d-2)|S'| - (d-2)|B| = (d-2)(|S| + |B|) - (d-2)|B| = (d-2)|S|$ many outgoing edges attached to a vertex in S. □

We obtain as a simple corollary that most points in S have a large number of unique neighbors:

Corollary 5.19. *Let S be a subset of points in a d-regular graph G with girth g, and let $\eta \in (0,1)$. If $|S| < \frac{g}{2d}$, then at least $\left(1 - \frac{2}{d(1-\eta)}\right)|S|$ many points have at least ηd unique neighbors.*

Proof. Let B be the set of vertices with less than ηd neighbors, and suppose that $|B| = \zeta|S|$. It holds that:

$$\begin{gathered}(d-2)|S| \leq \ \#\text{ Unique neighbors} \leq (1-\zeta)d + \eta d\zeta|S| \\ \Rightarrow \frac{d-2}{d} \leq 1 - \zeta + \eta\zeta \Rightarrow \zeta(1-\eta) \leq \frac{2}{d} \Rightarrow \zeta \leq \frac{2}{d(1-\eta)}\end{gathered}$$

□

Lastly, we will need the following lemma. It states that one conjugation of a d regular

graph can reduce the weight of a Pauli operator by at most a factor of $2d$.

Lemma 5.20. *Let* $U_G X_{\mathbf{e}_1} Z_{\mathbf{e}_2} U_G^\dagger = X_{\mathbf{q}_1} Z_{\mathbf{q}_2}$. *For a d-regular graph* G. *Define* $w_1 = |\mathbf{e}_1|$, $w_2 = |\mathbf{e}_2|$ *and* $w_3 = |supp(\mathbf{e}_1) \cup supp(\mathbf{e}_2)|$. *Then* $w_4 = |supp(\mathbf{q}_1) \cup supp(\mathbf{q}_2)| \geq \frac{w_3}{2d}$.

Proof. We can calculate $U_G X_{\mathbf{e}_1} Z_{\mathbf{e}_2} U_G^\dagger = X_{\mathbf{e}_1} Z_{\mathbf{e}_2 + A\mathbf{e}_1}$ by Equation (5.26) where A is the adjacency matrix of the graph G. If $\frac{w_1}{w_3} \geq \frac{1}{2d}$, then $w_4 \geq w_1 \geq \frac{w_3}{2d}$. Otherwise, $w_1 < \frac{w_3}{2d}$ so $w_2 \geq w_3 \left(1 - \frac{1}{2d}\right)$ and $|A\mathbf{e}_1| < \frac{w_3}{2}$. Hence, $w_4 \geq |\mathbf{e}_2 + A\mathbf{e}_1| \geq |\mathbf{e}_2| - |A\mathbf{e}_1| \geq w_3 \left(1 - \frac{1}{2d}\right) - \frac{w_3}{2} \geq \frac{w_3}{3}$. □

5.6 Analysis

5.6.1 Pauli Weight Amplifiers

Here we analyze phase 2 of our construction to show that it corresponds to a "weak" scrambler according to Definition 5.4. In each step, we will Clifford Twirl around each qubit, followed by conjugation by U_G (see Figure 5.5). The first part has the effect of randomizing the Pauli operator on it's support: Any non-identity Pauli matrix conjugates to X, Y or Z with equal probability. Then, the unitary U_G expands the support of the Pauli operator according to conjugation relations from Equation (2.28) using the adjacency matrix of the graph G. We imagine one step of this procedure as one step of a Markov chain with state space $X_{\mathbf{e}_1} Z_{\mathbf{e}_2}$ for $\mathbf{e}_1$, $\mathbf{e}_2 \in \mathbb{F}_2^n$. Each step is independent because our single Clifford qubit twirls are independent in each step. The Markov chain is obvious from our discussion, we define it formally for clarity of exposition:

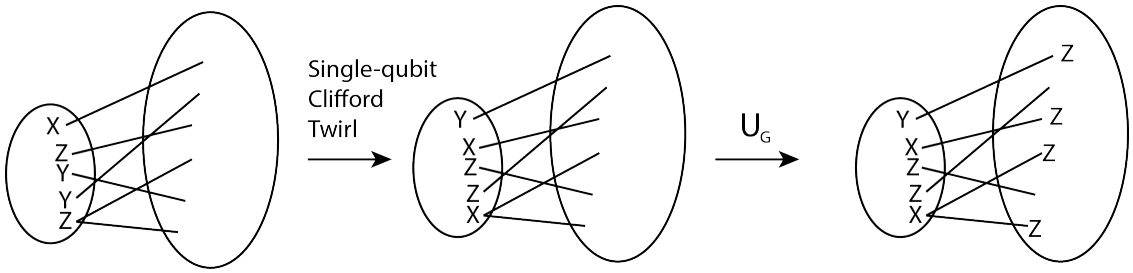


Figure 5.5: In each step, the support of the Pauli operator is sent to some other full weight Pauli string (uniformly), then conjugation by the graph state circuit increases the Pauli weight into the neighbor set.

Definition 5.21. *Let G be a graph with adjacency matrix A. We define $\mathcal{M}^G$ as a Markov chain with state space $\{X_{\mathbf{e}_1}Z_{\mathbf{e}_2} : \mathbf{e}_1, \mathbf{e}_2 \in \mathbb{F}_2^n\}$. If $(X_{\mathbf{e}_1^1}Z_{\mathbf{e}_2^1}) \to (X_{\mathbf{e}_1^2}Z_{\mathbf{e}_2^2}) \to \ldots \to (X_{\mathbf{e}_1^t}Z_{\mathbf{e}_2^t})$ is a sequence of steps in the Markov chain, then transitions $(X_{\mathbf{e}_1^i}Z_{\mathbf{e}_2^i}) \to (X_{\mathbf{e}_1^{i+1}}Z_{\mathbf{e}_2^{i+1}})$ are defined as follows.*

1. *Randomly generate $(\mathbf{q}_1, \mathbf{q}_2)$ where $\mathbf{q}_i \in \mathbb{F}_2^n$ in the following way: If $\mathbf{e}_1^i(j) \neq 0$ or $\mathbf{e}_2^i(j) \neq 0$ (or both), then*

$$(\mathbf{q}_1(j), \mathbf{q}_2(j)) = \begin{cases} (1,0) & \text{with probability } \frac{1}{3} \\ (0,1) & \text{with probability } \frac{1}{3} \\ (1,1) & \text{with probability } \frac{1}{3} \end{cases} \tag{5.56}$$

2. *Calculate $(\mathbf{e}_1^{i+1}, \mathbf{e}_2^{i+1}) = (\mathbf{q}_1, \mathbf{q}_2 + A\mathbf{q}_1)$*

There are a number of internal parameters involved in the analysis of the construction. the end goal of this section is to show that if the inequalities in equations 5.57-5.63 are satisfied for large enough l the outlined scheme forms a weak scrambler.

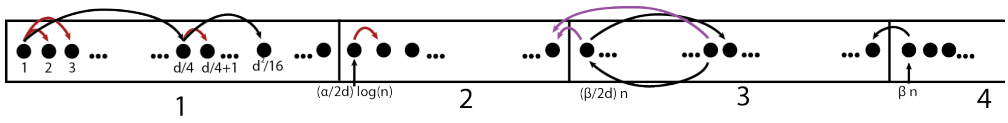


Figure 5.6: Here we depict four different regimes with labeled bad (red) and very bad (purple) edges.

$$f(\theta, d, \eta) := D\left(\theta \middle|\middle| \frac{2}{3}\right)\left(1 - \frac{2}{d(1-\eta)}\right)\frac{d}{2} \tag{5.57}$$

$$\theta\eta d\left(1 - \frac{2}{d(1-\eta)}\right) \geq 1 \tag{5.58}$$

$$h(\lambda) + \log\left(\frac{1}{3}\right)\lambda < 0 \tag{5.59}$$

$$1 - f(\theta, d, \eta)\mu < 0 \tag{5.60}$$

$$\log\left[\frac{d}{5}\left(\frac{1}{3} - \zeta\right)\right](1 - \mu - \lambda) + \log\left(\frac{1}{2d}\right)\mu > 0 \tag{5.61}$$

$$0 < \eta, \mu, \lambda < 1 \qquad 0 < \theta \leq 1/2 \tag{5.62}$$

$$0 < \zeta < 1/3 \qquad d \geq 21 \tag{5.63}$$

We need to describe another random process corresponding to the transitions of this Markov chain. It corresponds to examining the individual transitions for expansion. First, however, let us mark the transitions $X_{\mathbf{e}_1}Z_{\mathbf{e}_2} \to X_{\mathbf{f}_1}Z_{\mathbf{f}_2}$ according to the following rules:

Definition 5.22. *In the above chain we mark some edges as "good", "ok", "bad" and "very bad" in the following way. Suppose the edge is of the form $X_{\mathbf{e}_1}Z_{\mathbf{e}_2} \to X_{\mathbf{f}_1}Z_{\mathbf{f}_2}$. We will denote $size_e := |supp(\mathbf{e}_1) \cup supp(\mathbf{e}_2)|$ and $size_f := |supp(\mathbf{f}_1) \cup supp(\mathbf{f}_2)|$.*

- *"Good" edge if* $\left[size_e < \frac{\beta n}{2d} \text{ and } size_f > \frac{d}{5}\left(\frac{1}{3} - \zeta\right) size_e\right]$ *or* $\left[size_e \geq \beta n/(2d) \text{ and } size_f \geq \beta n/(2d)\right]$
- *"Ok" edge if* $1 \leq size_e < \frac{\beta n}{2d}$ *and* $size_e \leq size_f \leq \frac{d}{5}\left(\frac{1}{3} - \zeta\right) size_e$
- *"Bad" edge if* $size_e < \frac{\beta n}{2d}$ *and* $size_f < size_e$.

- *"Very Bad" edge if* $\frac{\beta n}{2d} \leq size_e \leq \beta n$ *and* $size_f < \frac{\beta n}{2d}$.

Now we can define a process corresponding to the 4 types of edges we might see as we traverse the Markov chain:

Definition 5.23. *Given* t *steps in the Markov walk* $\mathcal{M}^G$*, we define a random sequence* $\mathbf{F}^G = (F_1, F_2, ..., F_{t-1})$ *associated with the* $t-1$ *transitions of* $\mathcal{M}$*. We set:*

$$F_i = \begin{cases} +1 \text{ if transition is good} \\ 0 \text{ if transition is ok} \\ -1 \text{ if transition is bad} \\ -2 \text{ if transition is very bad} \end{cases} \tag{5.64}$$

Now we are in a position to start proving the main lemmas.

Lemma 5.24. *Let* $X_{\mathbf{e}_1^t} Z_{\mathbf{e}_2^t}$ *be any support,* η*,* θ *and* ζ *be constants satisfying equations 5.57-5.63. Then, for* n *large enough the probability that* $X_{\mathbf{e}_1^t} Z_{\mathbf{e}_2^t} \to X_{\mathbf{e}_1^{t+1}} Z_{\mathbf{e}_2^{t+1}}$ *is a bad edge is at most* $2^{-f(\theta,d,\eta)}$ *and the probability that an edge is ok is at most* $1/3$*.*

Proof. We break up the set of strings into 4 regimes as depicted in Figure 5.6. Each Pauli operator falls into one of the regimes based on it's Pauli weight. In regime 1, the tree-like regime, we can use several of the technical lemmas to arrive at the conclusion. Otherwise, the Read-k Chrenoff bound provides an upper bound.

- **Regime 1**: It is easy to see that if $size_e < d/2$, then there are no bad edges. In order for the weight to strictly decrease, at least one of the vertices must have a X or a Y operator. If this is the case, then we can conclude the weight actually increased by examining the neighbors of that point. Hence, we can assume the weight is at least $d/2$. Now let us apply Corollary 5.19, which says that there are at least $\left(1 - \frac{2}{d(1-\eta)}\right)|S|$ many points, each with ηd neighbors. It a fraction θ of these are "active" (the random

Clifford rolls a X or Y on that point) then we will get expansion by a factor of at least:

$$\theta\eta\left(1-\frac{2}{d(1-\eta)}\right) \tag{5.65}$$

By Equation (5.58), this should be at least 1, so this edge is good or ok. Hence, we can upper bound the probability of a bad edge as

$$2^{-D(\theta||2/3)\left(1-\frac{1}{d(1-\eta)}\right)|S|} \tag{5.66}$$

By assumption, $|S| \geq d/2$, so the probability is at most

$$2^{-D(\theta||2/3)\left(1-\frac{2}{d(1-\eta)}\right)d/2} =: 2^{-f(\theta,d,\eta)} \tag{5.67}$$

To upper bound the probability of an ok edge, we know from Corollary 5.19, that there are at least $\left(1-\frac{4}{d}\right)|S|$ many points, each with $d/2$ neighbors. As long as half of these points "roll" a X or Y, we should have expansion by a factor of $d/4$. By Lemma 5.18, the probability this does not happen is at most $1/3$.

- **Regime 2**: Here the weight is larger than our girth cutoff $\alpha\log(n)/(2d)$, but may not be as large as $\Omega(n)$. Suppose the support is some set S. By hypothesis, the neighbor set of S is at least $(d/5)|S|$. Along the lines of Theorem 5.11, let us denote $\{P_1, P_2, \ldots P_{|S|}\}$ as random variables where each P_i is 1 if the Clifford twirl results in a X or Y operator at points in S. Similarly define $\{Q_1, Q_2, \ldots Q_{|N(S)|}\}$ to be random variables where each Q_i is 1 if as a result of conjugation by U_G there is a Z operator on that vertex and 0 otherwise. Suppose $\mathbb{E}[Q_i] = q_i$, and q is the average of q_i. By Lemma 5.12, $1/3 \leq q \leq 2/3$. It is clear we can apply Theorem 5.11 in its stated form.

$$\mathbb{P}\left[Q_1 + \ldots + Q_{|N(S)|} \leq \frac{d}{5}\left(\frac{1}{3} - \zeta\right)|S|\right] \leq \mathbb{P}\left[Q_1 + \ldots + Q_{|N(S)|} \leq |N(S)|\left(\frac{1}{3} - \zeta\right)\right] \tag{5.68}$$
$$\leq \mathbb{P}\left[Q_1 + \ldots + Q_{|N(S)|} \leq |N(S)|\left(q - \zeta\right)\right] \leq 2^{-D(q-\zeta||q)|N(S)|/d} \leq 2^{-\zeta^2|N(S)|/d}$$

Since for this regime, $|N(S)| \propto \log(n)$, clearly for large enough n and $\zeta > 0$ this upper bound is smaller than $2^{-f(\theta,d,\eta)}$.

- **Regimes 3 and 4**: By definition, in these regimes there are no bad edges or ok edges (only very bad edges) so that probability of a bad or ok edge is zero.

□

The main technical lemma is as follows:

Lemma 5.25. *Consider a walk on $\mathcal{M}^G$ of length $t = l\log(n)$, and let ζ, μ, λ be constants satisfying equations 5.57-5.63. For large enough l and n there will be at least $(1-\lambda-\mu)l\log(n)$ many good steps, no very bad steps and less than $\mu l\log(n)$ many bad steps with probability $1 - O(1/poly(n))$.*

Proof. Consider all possible outcomes of the stochastic process $\mathbf{F}^G$. These are all sequences of length $l\log(n) - 1$ where each element in the series is $+1$, -1, 0, or -2.

- The probability of any outcome that contains an element equal to -2 is exponentially small with n. This is easy to see from Theorem 5.11. By Lemma 5.20, we can see that any support $X_{\mathbf{e}_1}Z_{\mathbf{e}_2}$ can get at most a factor of $2d$ smaller in each step. Thus, all very bad edges start in regime 3, where we still have our expansion guarantee. As before

let $\{Q_i\}_{i=1}^{|N(S)|}$ be indicator random variables for Z operators in the neighbor set, and let $q = \frac{\sum_i \mathbb{E}[Q_i]}{|N_G(S)|}$. Then,

$$\mathbb{P}\left[Q_1 + \ldots + Q_{|N(S)|} \leq |S|\right] \leq \mathbb{P}\left[Q_1 + \ldots + Q_{|N(S)|} \leq \frac{d}{5}\left(\frac{1}{3} - \zeta\right)|S|\right] \tag{5.69}$$
$$\leq \mathbb{P}\left[Q_1 + \ldots + Q_{|N(S)|} \leq |N(S)|\left(\frac{1}{3} - \zeta\right)\right] \leq \mathbb{P}\left[Q_1 + \ldots + Q_{|N(S)|} \leq |N(S)|\,(q - \zeta)\right]$$
$$\leq 2^{-D(q-\zeta||q)|N(S)|/d} \leq 2^{-(\zeta)^2|N(S)|/d} = 2^{-\Omega(n)}$$

Hence, we can assume that *no* very bad edges occur.

- Consider the random variables $\mathbf{F}^G = (F_1, F_2, ..., F_{l\log(n)-1})$, and some particular outcome $(F_1, F_2, ..., F_{l\log(n)-1}) = (f_1, f_2, ..., f_{l\log(n)-1})$ where at least $\mu l \log(n)$ many $f_i = -1$. We can write using the chain rule:

$$\mathbb{P}[\mathbf{F} = \mathbf{f}] = \tag{5.70}$$
$$\mathbb{P}[F_1 = f_1]\mathbb{P}[F_2 = f_2|F_1 = f_1]...\mathbb{P}[F_{l\log(n)-1} = f_{l\log(n)-1}|F_{l\log(n)-2} = f_{l\log(n)-2}, ..., F_1 = f_1]$$

We can "calculate" $\mathbb{P}[F_i = f_i|F_{i-1} = f_{i-1}, ..., F_1 = f_1]$ by conditioning on all possible paths (on the supports) consistent with $(F_1 = f_1, ..., F_{i-1} = f_{i-1})$. Since our upper bound from Lemma 5.24 is uniform over *all* supports, $\mathbb{P}[F_i = -1|F_{i-1} = f_{i-1}, ..., F_1 = f_1] \leq 2^{-f(\theta,d,\eta)}$ and trivially $\mathbb{P}[F_i = +1|F_{i-1} = f_{i-1}, ..., F_1 = f_1] \leq 1$. There are at most $2^{l\log(n)}$ configurations of the positions of bad edges, so

$$\mathbb{P}\left[\text{at least } \mu\, l\log(n) \text{ bad steps}\right] \leq \log(n)2^{l\log(n)}\left(2^{-f(\theta,d,\eta)}\right)^{\mu l\log(n)} \tag{5.71}$$
$$= 2^{\wedge}\left[l\log(n)\left(1 + \log\left(2^{-f(\theta,d,\eta)}\right)\mu\right)\right]$$

As long as the term in parenthesis above is negative (Equation (5.60)) for large enough l the above is polynomially small in n.

- Now we turn to providing an upper bound on the probability that there are at least $\lambda l \log(n)$ many ok steps. Again by the union bound, we calculate:

$$\begin{aligned}\mathbb{P}\left[\text{at least } \lambda\, l\log(n) \text{ ok steps}\right] &\leq l\log(n)\binom{l\log(n)}{\lambda l\log(n)}\left(\frac{1}{3}\right)^{\lambda l\log(n)} \\ &\leq 2^{\wedge}\left[l\log(n)\left(h(\lambda)-\log(1/3)\lambda\right)\right]\end{aligned} \tag{5.72}$$

  The term in parenthesis is negative by Equation (5.59), hence for large enough l the above is polynomially small.

□

Theorem 5.26. *Let $(X_{\mathbf{e}_1^1}Z_{\mathbf{e}_2^1}) \to (X_{\mathbf{e}_1^2}Z_{\mathbf{e}_2^2}) \to \ldots \to (X_{\mathbf{e}_1^{t+1}}Z_{\mathbf{e}_2^{t+1}})$ be some Markov walk on $\mathcal{M}^G$ of length $l\log(n)$. With probability at least $1 - O(1/poly(n))$ for large enough d and l, the weight $|supp(\mathbf{e}_1^{t+1} \cup supp(\mathbf{e}_2^{t+1})|$ is at least $\frac{\beta n}{2d}$.*

Proof. Let us use the previous lemma. We know that the walk has at most $\mu l\log(n)$ bad steps and $\lambda l\log(n)$ ok steps with high probability. In each good step, the weight must have gotten larger by a factor of $d/5(1/3-\zeta)$, or must have been larger than $\beta n/(2d)$. Hence the weight must be:

$$|supp(\mathbf{e}_1^{t+1}) \cup supp(\mathbf{e}_2^{t+1})| \geq \min\left\{\left(\frac{d}{6}\left(\frac{1}{3}-\zeta\right)\right)^{(1-\mu-\lambda)l\log(n)}\left(\frac{1}{2d}\right)^{\mu l\log(n)}, \frac{\beta n}{2d}\right\} \tag{5.73}$$

The first term can be written as:

$$2^{\wedge}\left[l\log(n)\left(\log\left(\frac{d}{5}\left(\frac{1}{3}-\zeta\right)\right)(1-\mu-\lambda)+\mu\log\left(\frac{1}{2d}\right)\right)\right] \tag{5.74}$$

Assuming Equation (5.61) the term in parenthesis is positive. Hence, at some point, for large enough l, the weight must have gotten larger than $\beta n/(2d)$. Since we can assume there are no very bad edges, the weight must be at least $\beta n/(2d)$ at the end of the walk. □

5.6.2 ER Analysis

Suppose we have some Pauli operator $X_{\mathbf{e}_1} Z_{\mathbf{e}_2}$ with $|\mathbf{e}_2| \geq \beta' n$.

Theorem 5.27. *Let Let ER_1, ER_2, and ER_3 be Erdos-Renyi graphs with probability $\frac{w \ln(n)}{n}$ of edge formation, each sampled independently. Let U_{ER_1}, U_{ER_2} and U_{ER_3} be the corresponding graph state unitaries (defined in Equation (5.26)).*

Denote $\mathcal{Q}_{\mathbf{e}_1,\mathbf{e}_2}(\mathbf{q}_1, \mathbf{q}_2)$ be the measure corresponding to the conjugated Pauli matrix:

$$\mathcal{Q}_{\mathbf{e}_1,\mathbf{e}_2}(\mathbf{q}_1, \mathbf{q}_2) = \mathbb{P}\left[C X_{\mathbf{e}_1} Z_{\mathbf{e}_2} C^\dagger \propto X_{\mathbf{q}_1} Z_{\mathbf{q}_2}\right]$$

For large enough $w = O_n(1)$:

$$\Delta(\mathcal{Q}_{\mathbf{e}_1,\mathbf{e}_2}, \mathcal{U}) = O\left(\frac{1}{poly(n)}\right) \tag{5.75}$$

We will require another helper lemma:

Lemma 5.28. *Let $\mathbf{e}$ be a binary vector on n bits with weight at least $\beta' n$. Let B be a binary symmetric matrix where each off diagonal entry is 1 with probability $p = \frac{w \ln(n)}{n}$ and 0 otherwise. Further, let each diagonal element be 1 with probability $1/2$ and zero otherwise. For large enough w, it holds that:*

$$\forall \mathbf{q} \quad \left(\frac{1}{2}\right)^n \left(1 - \frac{4}{n^{2\beta' w - 1}}\right) \leq \mathbb{P}\left(\mathbf{q} = B\mathbf{e}\right) \leq \left(\frac{1}{2}\right)^n \left(1 + \frac{4}{n^{2\beta' w - 1}}\right) \tag{5.76}$$

Proof. We define several sets of bits. Let the set A_1 denote the support of $\mathbf{e}$, let A_2 denote the support of the vector $\mathbf{q}$ minus A_1, and let A_3 denote the rest of the bits. Let the sizes of these sets be a_1, a_2 and a_3 respectively. The probability calculated exactly is:

$$\mathbb{P}(\mathbf{q} = B\mathbf{e}) = \left(\frac{1}{2}\right)^{a_1} \left(\frac{1 - (1-2p)^{a_1}}{2}\right)^{a_2} \left(\frac{1 + (1-2p)^{a_1}}{2}\right)^{a_3} \tag{5.77}$$

By assumption $\beta' n \leq a_1 \leq n$ so,

$$(1-2p)^n \leq (1-2p)^{a_1} \leq (1-2p)^{\beta' n} \tag{5.78}$$

Calculating limits we can see that for large enough n:

$$\frac{1}{2n^{2w}} \leq (1-2p)^{a_1} \leq \frac{2}{n^{2\beta' w}} \tag{5.79}$$

The upper bound is calculated as:

$$\begin{aligned}\mathbb{P}(\mathbf{q} = B\mathbf{e}) \leq \left(\frac{1}{2}\right)^{a_1+a_2+a_3} \left(1+\frac{2}{n^{2\beta' w}}\right)^{a_3} &= \left(\frac{1}{2}\right)^{a_1+a_2+a_3} \left(\sum_{j=0}^{a_3} \left(\frac{2}{n^{2\beta' w}}\right)^j \binom{a_3}{j}\right) \\ &\leq \left(\frac{1}{2}\right)^{a_1+a_2+a_3} \left(1+\frac{4a_3}{n^{2\beta' w}}\right)\end{aligned} \tag{5.80}$$

where in the last inequality we assumed n, w large enough.

The lower bound is calculated as:

$$\mathbb{P}(\mathbf{q} = B\mathbf{e}) \geq \left(\frac{1}{2}\right)^{a_1+a_2+a_3} \left(1-\frac{2}{n^{2\beta' w}}\right)^{a_2} \geq \left(\frac{1}{2}\right)^{a_1+a_2+a_3} \left(1-\frac{4a_2}{n^{2\beta' w}}\right) \tag{5.81}$$

□

Proof of Theorem 5.27. Let U_G be a unitary described in Equation (5.26), and suppose G has adjacency matrix A. Then by Equation (2.28)

$$U_G H^{\otimes n} X_{\mathbf{e}_1} Z_{\mathbf{e}_2} H^{\otimes n} U_G^\dagger = X_{\mathbf{e}_2} Z_{\mathbf{e}_1 + A\mathbf{e}_2} \tag{5.82}$$

If we apply this to our ensemble of random Erdos-Renyi graphs, we can derive:

$$\begin{bmatrix}\mathbf{q}_1 \\ \mathbf{q}_2\end{bmatrix} = \begin{bmatrix}A_2 & A_2A_1 + \mathbb{I} \\ A_3A_2 + \mathbb{I} & A_1 + A_3(A_2A_1 + \mathbb{I})\end{bmatrix} \begin{bmatrix}\mathbf{e}_1 \\ \mathbf{e}_2\end{bmatrix} \tag{5.83}$$

where $X_{\mathbf{e}_1} Z_{\mathbf{e}_2}$ conjugates to $X_{\mathbf{q}_1} Z_{\mathbf{q}_2}$ under our ensemble, and each A_i corresponds to the adjacency matrix of ER_i.

The outcome space here is the set of randomly generated matrices A_i. Define the random variables $(\mathbf{Q}_1, \mathbf{Q}_2)$ equal to the randomly generated LHS. The set of events $\{(\mathbf{Q}_1, \mathbf{Q}_2) =$

$(\mathbf{q}_1, \mathbf{q}_2)\}$ form a partition of the outcome space. Define the event:

$$B = \left\{ |\mathbf{Q}_1| \geq \beta' n \cap |\mathbf{e}_1 + A_1 \mathbf{e}_2| \geq \beta' n \right\} \tag{5.84}$$

and let $(\widetilde{\mathbf{Q}}_1, \widetilde{\mathbf{Q}}_2)$ be the distribution of $(\mathbf{Q}_1, \mathbf{Q}_2)$ conditioned on this event. We can calculate:

$$\mathbb{P}[\neg B] = \mathbb{P}\left\{ |\mathbf{Q}_1| < \beta' n \cup |\mathbf{e}_1 + A_1 \mathbf{e}_2| < \beta' n \right\} \leq \mathbb{P}[|\mathbf{Q}_1| < \beta' n] + \mathbb{P}[|\mathbf{e}_1 + A_1 \mathbf{e}_2| < \beta' n] \tag{5.85}$$

$$= \mathbb{P}\left[|\mathbf{Q}_1| < \beta' n \cap |\mathbf{e}_1 + A_1 \mathbf{e}_2| < \beta' n \right] + \mathbb{P}\left[|\mathbf{Q}_1| < \beta' n \cap |\mathbf{e}_1 + A_1 \mathbf{e}_2| \geq \beta' n \right] +$$

$$\mathbb{P}[|\mathbf{e}_1 + A_1 \mathbf{e}_2| < \beta' n] \leq 2\mathbb{P}[|\mathbf{e}_1 + A_1 \mathbf{e}_2| < \beta' n] + \mathbb{P}\left[|\mathbf{Q}_1| < \beta' n \cap |\mathbf{e}_1 + A_1 \mathbf{e}_2| \geq \beta' n \right]$$

Lemma 5.28 applied to matrices A_1 and A_2 implies both of these terms are exponentially small with n. Hence, if the measure of $(\mathbf{Q}_1, \mathbf{Q}_2)$ is $\mathcal{Q}_{\mathbf{e}_1, \mathbf{e}_2}$ and the measure of $(\widetilde{\mathbf{Q}}_1, \widetilde{\mathbf{Q}}_2)$ is $\widetilde{\mathcal{Q}}_{\mathbf{e}_1, \mathbf{e}_2}$ then Lemma 5.14 implies $\Delta(\mathcal{Q}_{\mathbf{e}_1, \mathbf{e}_2}, \widetilde{\mathcal{Q}}_{\mathbf{e}_1, \mathbf{e}_2})$ is exponentially small.

Now we must analyze the distribution of $(\widetilde{\mathbf{Q}}_1, \widetilde{\mathbf{Q}}_2)$ to obtain a bound on the statistical distance from $\mathcal{U}$. We will first condition on the value of $\mathbf{e}_1 + A_1 \mathbf{e}_2$. Let $\mathbf{q}_1$ be some vector with weight at least $\beta' n$:

$$\mathbb{P}\left[\widetilde{\mathbf{Q}_1} = \mathbf{q}_1 \right] = \mathbb{P}\left[\mathbf{Q}_1 = \mathbf{q}_1 \mid \left(|\mathbf{Q}_1| \geq \beta' n \right) \cap \left(|\mathbf{e}_1 + A_1 \mathbf{e}_2| \geq \beta' n \right) \right] \tag{5.86}$$

Since $\mathbf{Q}_1 = \mathbf{q}_1 \Rightarrow |\mathbf{Q}_1| \geq \beta' n$, we can re-write this as:

$$= \frac{\mathbb{P}\left[\left(\mathbf{Q}_1 = \mathbf{q}_1 \right) \cap \left(|\mathbf{e}_1 + A_1 \mathbf{e}_2| \geq \beta' n \right) \right]}{\mathbb{P}\left[\left(|\mathbf{Q}_1| \geq \beta' n \right) \cap \left(|\mathbf{e}_1 + A_1 \mathbf{e}_2| \geq \beta' n \right) \right]} =$$

$$\frac{\mathbb{P}\left[\mathbf{Q}_1 = \mathbf{q}_1 \;\middle|\; |\mathbf{e}_1 + A_1 \mathbf{e}_2| \geq \beta' n \right] \mathbb{P}\left[|\mathbf{e}_1 + A_1 \mathbf{e}_2| \geq \beta' n \right]}{\mathbb{P}\left[\left(|\mathbf{Q}_1| \geq \beta' n \right) \cap \left(|\mathbf{e}_1 + A_1 \mathbf{e}_2| \geq \beta' n \right) \right]}$$

By Lemma 5.28, the numerator is within a factor $\left(1 \pm \frac{1}{poly(n)}\right)$ of $1/2^n$ and the denominator is exponentially close to 1 by Equation (5.85). Hence,

$$\frac{1}{2^n}\left(1 - \frac{1}{poly(n)}\right) \leq \mathbb{P}[\widetilde{\mathbf{Q}}_1 = \mathbf{q}_1] \leq \frac{1}{2^n}\left(1 + \frac{1}{poly(n)}\right) \tag{5.87}$$

We can rewrite the matrix equations as:

$$\mathbf{q}_1 = A_2(\mathbf{e}_1 + A_1\mathbf{e}_2) + \mathbf{e}_2 \tag{5.88}$$
$$\mathbf{q}_2 = A_3\mathbf{q}_1 + \mathbf{e}_1 + A_1\mathbf{e}_2 \tag{5.89}$$

So far our conditioning has been completely independent of A_3. Therefore, we can once again apply Lemma 5.28 to A_3 to obtain:

$$\mathbb{P}\left[\widetilde{\mathbf{Q}_2} = \mathbf{q}_2 \mid \widetilde{\mathbf{Q}_1} = \mathbf{q}_1\right] = \frac{1}{2^n}\left(1 \pm \frac{1}{poly(n)}\right) \tag{5.90}$$

Now we are equipped to calculate the statistical distance. Let us denote the measure corresponding to $(\widetilde{\mathbf{Q}_1}, \widetilde{\mathbf{Q}_2})$ as $\mathbf{Q}'_{\mathbf{e}_1,\mathbf{e}_2}$. We calculate:

$$\Delta(\mathcal{Q}_{\mathbf{e}_1,\mathbf{e}_2}, \mathcal{U}) \leq \Delta(\mathcal{Q}_{\mathbf{e}_1,\mathbf{e}_2}, \mathcal{Q}'_{\mathbf{e}_1,\mathbf{e}_2}) + \Delta(\mathcal{Q}'_{\mathbf{e}_1,\mathbf{e}_2}, \mathcal{U}) \tag{5.91}$$

We have already demonstrated the first term is exponentially small. For the second, denote the random Pauli operator from $\mathcal{U}$ as $X_{\mathbf{P}_1}Z_{\mathbf{P}_2}$.

$$\Delta(\mathcal{Q}'_{\mathbf{e}_1,\mathbf{e}_2}, \mathcal{U}) = \mathbb{P}\left[|\mathbf{P}_1| < \beta' n\right] + \sum_{\substack{|\mathbf{q}_1| \geq \beta' n \\ \mathbf{q}_2}} \left| \frac{1}{4^n}\left(1 \pm \frac{1}{poly(n)}\right) - \frac{1}{4^n - 1} \right| \leq \frac{1}{poly(n)} \tag{5.92}$$

□

5.6.3 Explicit Derivation of Main Results

Let us fix a particular Pauli operator P_x. Let d be the regularity of the Ramanujan graph, and set $\beta = 1/(2d^3)$. The first step is to find constants which satisfy equations 5.57 through 5.63. Any standard solver can be used to look for solutions to these equations. In MATLAB,

we ran an sqp solver with the goal of minimizing d subject to the inequalities. We obtain:

$$[d, \eta, \lambda, \mu, \theta, \zeta] = [33, 0.56, 0.61, 0.06, 0.06, 1/100]$$

Using these constants, for large enough d, l, Theorem 5.26 implies that we can expect the conjugated Pauli operator after phase 2 to have Pauli weight at least $(1/d^2)(1/2d)n$ with all but inverse polynomial probability. To prove that we can expect $|\mathbf{e}_2| \geq (\beta/6)n$ after phase 2.5, a simple Chernoff bound is sufficient. The final Clifford Twirl (phase 2.5) maps each operator in the support to X, Y or Z with probability 1/3 independently. We can expect weight at least $\beta/6n$ with all but exponentially small probability (conditioned on the last step). Let B be the event "The conjugated Pauli after phase 2.5 has $|\mathbf{e}_2| \geq (\beta/6)n$". We have established that

$$\mathbb{P}[\neg B] = O\left(\frac{1}{poly(n)}\right) \tag{5.93}$$

Now we will leverage Lemma 5.14 and Theorem 5.27 to obtain Theorem 5.9. Let

$$\mathcal{Q}[P_y] = \mathbb{P}[P_x \xrightarrow{\mathcal{E}} P_y] \tag{5.94}$$

be the propagator for the entire ensemble. Let $(\mathbf{Q}_1, \mathbf{Q}_2)$ be the corresponding random variables. Just as in Theorem 5.27, we construct $(\widetilde{\mathbf{Q}}_1, \widetilde{\mathbf{Q}}_2)$ which is $(\mathbf{Q}_1, \mathbf{Q}_2)$ conditioned on the event B. Let $\mathcal{Q}'$ be the corresponding measure.

By Lemma 5.14 and Equation (5.93), $\Delta(\mathcal{Q}, \mathcal{Q}') = O(1/poly(n))$.

Let $\mathcal{A}$ be the set of Pauli operators satisfying event B after phase 2.5. We can calculate $\mathcal{Q}'[P_y]$ by conditioning on the Pauli operator after phase 2.5.

$$\mathcal{Q}'[P_y] = \frac{\sum_{P_z \in \mathcal{A}} \mathbb{P}[P_x \xrightarrow{\mathcal{E}_{2.5} \circ \mathcal{E}_2 \circ \mathcal{E}_1} P_z] \mathbb{P}[P_z \xrightarrow{\mathcal{E}_3} P_y]}{\sum_{P_z \in \mathcal{A}} \mathbb{P}[P_x \xrightarrow{\mathcal{E}_{2.5} \circ \mathcal{E}_2 \circ \mathcal{E}_1} P_z]} \tag{5.95}$$

Define the measure $\mathcal{R}_{P_z}[P_y] = \mathbb{P}[P_z \xrightarrow{\mathcal{E}_3} P_y]$. Theorem 5.27 implies a uniform upper bound on

the statistical distance $\Delta(\mathcal{U}, \mathcal{R}_{P_z}) \leq \varepsilon = O(1/poly(n))$ for large enough w. The measure $\mathcal{Q}'$ is a linear combination of the measures $\mathcal{R}_{P_z}$, so

$$\Delta(\mathcal{U}, \mathcal{Q}') \leq \varepsilon \frac{\sum_{P_z \in \mathcal{A}} \mathbb{P}[P_z \xrightarrow{\mathcal{E}_3} P_y]}{\sum_{P_z \in \mathcal{A}} \mathbb{P}[P_z \xrightarrow{\mathcal{E}_3} P_y]} = \varepsilon \tag{5.96}$$

With the observation $\Delta(\mathcal{U}, \mathcal{Q}) \leq \Delta(\mathcal{U}, \mathcal{Q}') + \Delta(\mathcal{Q}, \mathcal{Q}')$ we have demonstrated the assumptions of Lemma 5.6, and hence we derive Theorem 5.9.

5.7 Conclusions and Future Directions

Here we have presented a novel scheme providing a "weak" approximate unitary 2-design with better circuit depth or gate count than many other known constructions. We imagine that it is useful primarily in noise estimation/randomized benchmarking contexts since our construction seems to provide poor scrambling properties (i.e. see the discussion in Section 5.2). This result fits in well with other schemes [33, 84], which show that randomized benchmarking can be achieved with "weak" sampling of the Clifford group.

The careful reader will note that the required block size for our construction to be effective is very large. The problem here is in the proof of Lemma 5.24, Regime 2. Here we require $\log(n)$ be on the order of d^2/ζ^2, which implies a very large $\log(n)$. This can be improved on by using a more aggressive expansion promise (i.e. [95]), however this would reduce the largest set size on which we have an expansion promise (recall in the maximum set size for our expansion promise is $|S| \leq n/d^2$ in Theorem 5.16). The issue with this is that we would have to increase the parameter w to compensate, and we need $\log(n)$ to be large with respect to w for the ER analysis. Note that the minimal requirement for that lemma is that the *girth* be large with respect to d^2/ζ^2 (a constant). It is possible that different constructions

of expander graphs could yield good expansion properties and large girth. The results can be modified to work if the graph is only *quasi-Ramanujan* so it is possible that a number of combinatorial constructions [14, 21] can be used in place of LPS graphs. Nonetheless, we have given a set of ideas here which can potentially be improved to yield a very efficient randomized benchmarking scheme.

The most interesting part of the scheme is phase 2. As discussed, this phase has the effect of increasing the Pauli weight of it's input with high probabillity. It is possible that this phase alone could be some type of approximate design. Since expander graphs are known to mix quickly, it makes good sense that the repeated conjugation by these circuits might yield a distribution over Pauli operators close to uniform. It this were the case, it would yield an approximate-ε-Twirl design with time independent, deterministic qubit to qubit couplings. This could be a very interesting construction in the context of experimental randomized benchmarking. [81] contains a similar conjecture. There they speculate that random two qubit gates along the edges of an expander could yield a design in logarithmic depth.

Bibliography

[1] Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Physical Review A*, 70(5):052328, 2004.

[2] Dorit Aharonov and Lior Eldar. On the complexity of commuting local hamiltonians, and tight conditions for topological order in such systems. In *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*, pages 334–343. IEEE, 2011.

[3] Dorit Aharonov and Lior Eldar. The commuting local Hamiltonian problem on locally expanding graphs is approximable in NP. *Quantum Information Processing*, 14(1):83–101, 2015.

[4] Noga Alon, László Babai, and Alon Itai. A fast and simple randomized parallel algorithm for the maximal independent set problem. *Journal of algorithms*, 7(4):567–583, 1986.

[5] Noga Alon and Fan RK Chung. Explicit construction of linear sized tolerant networks. *Annals of Discrete Mathematics*, 38:15–19, 1988.

[6] Salah A Aly. A class of quantum ldpc codes derived from latin squares and combinatorial design. Technical report, Tech. rep., Department of Computer Science, Texas A&M University, 2007.

[7] Andris Ambainis and Joseph Emerson. Quantum t-designs: t-wise independence in the quantum world. In *Computational Complexity, 2007. CCC'07. Twenty-Second Annual IEEE Conference on*, pages 129–140. IEEE, 2007.

[8] Theodore W Anderson, Ingram Olkin, and Les G Underhill. Generation of random orthogonal matrices. *SIAM Journal on Scientific and Statistical Computing*, 8(4):625–629, 1987.

[9] Erdal Arikan. Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Transactions on Information Theory*, 55(7):3051–3073, 2009.

[10] Alexei Ashikhmin and Simon Litsyn. Upper bounds on the size of quantum codes. *IEEE Transactions on Information Theory*, 45(4):1206–1215, May 1999.

[11] Alán Aspuru-Guzik, Anthony D Dutoi, Peter J Love, and Martin Head-Gordon. Simulated quantum computation of molecular energies. *Science*, 309(5741):1704–1707, 2005.

[12] Ohad Barak, David Burshtein, and Meir Feder. Bounds on achievable rates of ldpc codes used over the binary erasure channel. *IEEE transactions on information theory*, 50(10):2483–2492, 2004.

[13] Avraham Ben-Aroya and Amnon Ta-Shma. Constructing small-bias sets from algebraic-geometric codes. In *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*, pages 191–197. IEEE, 2009.

[14] Avraham Ben-Aroya and Amnon Ta-Shma. A combinatorial construction of almost-ramanujan graphs using the zig-zag product. *SIAM Journal on Computing*, 40(2):267–290, 2011.

[15] Yael Ben-Haim and Simon Litsyn. Upper bounds on the rate of ldpc codes as a function of minimum distance. *IEEE Transactions on Information Theory*, 52(5):2092–2100, 2006.

[16] Charles H Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. *Theor. Comput. Sci.*, 560(P1):7–11, 2014.

[17] Charles H. Bennett, David P. DiVincenzo, and John A. Smolin. Capacities of quantum erasure channels. *Phys. Rev. Lett.*, 78:3217–3220, Apr 1997.

[18] Charles H Bennett, David P DiVincenzo, John A Smolin, and William K Wootters. Mixed-state entanglement and quantum error correction. *Physical Review A*, 54(5):3824, 1996.

[19] Elwyn Berlekamp, Robert McEliece, and Henk Van Tilborg. On the inherent intractability of certain coding problems (corresp.). *IEEE Transactions on Information Theory*, 24(3):384–386, 1978.

[20] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195, 2017.

[21] Yonatan Bilu and Nathan Linial. Lifts, discrepancy and nearly optimal spectral gap. *Combinatorica*, 26(5):495–519, 2006.

[22] Rico Blaser and Piotr Fryzlewicz. Random rotation ensembles. *The Journal of Machine Learning Research*, 17(1):126–151, 2016.

[23] Héctor Bombín. An introduction to topological quantum codes. In Daniel A. Lidar and Todd A. Brun, editors, *Quantum Error Correction*. Cambridge University Press, New York, 2013.

[24] Hector Bombin and Miguel Angel Martin-Delgado. Topological quantum distillation. *Physical review letters*, 97(18):180501, 2006.

[25] Fernando G.S.L. Brandao and Aram W. Harrow. Product-state approximations to quantum ground states. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pages 871–880, New York, NY, USA, 2013. ACM.

[26] Fernando GSL Brandao, Aram W Harrow, and Michal Horodecki. Local random quantum circuits are approximate polynomial-designs. *arXiv preprint arXiv:1208.0692*, 2012.

[27] Sergey Bravyi and Jeongwan Haah. Magic-state distillation with low overhead. *Physical Review A*, 86(5):052329, 2012.

[28] Sergey Bravyi and Matthew B. Hastings. Homological product codes. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, STOC '14, pages 273–282, New York, NY, USA, 2014. ACM.

[29] Sergey Bravyi and Alexei Kitaev. Universal quantum computation with ideal clifford gates and noisy ancillas. *Physical Review A*, 71(2):022316, 2005.

[30] 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.

[31] Anne Broadbent and Stacey Jeffery. Quantum homomorphic encryption for circuits of low t-gate complexity. In *Annual Cryptology Conference*, pages 609–629. Springer, 2015.

[32] Winton Brown and Omar Fawzi. Scrambling speed of random quantum circuits. *arXiv preprint arXiv:1210.6644*, 2012.

[33] Winton G Brown and Bryan Eastin. Randomized benchmarking with restricted gate sets. *arXiv preprint arXiv:1801.04042*, 2018.

[34] David Burshtein, Michael Krivelevich, Simon Litsyn, and Gadi Miller. Upper bounds on the rate of ldpc codes. *IEEE Transactions on Information Theory*, 48(9):2437–2449, 2002.

[35] A Robert Calderbank, Eric M Rains, PM Shor, and Neil JA Sloane. Quantum error correction via codes over gf (4). *IEEE Transactions on Information Theory*, 44(4):1369–1387, 1998.

[36] Arthur R. Calderbank and Peter W. Shor. Good quantum error-correcting codes exist. *Phys. Rev. A*, 54(2):1098–1105, Aug 1996.

[37] Thomas Camara, Harold Ollivier, and J-P Tillich. Constructions and performance of classes of quantum ldpc codes. *arXiv preprint quant-ph/0502086*, 2005.

[38] Herman Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *The Annals of Mathematical Statistics*, pages 493–507, 1952.

[39] David Lai Gwai Cheung. *Structures and properties of liquid crystals and related molecules from computer simulation.* PhD thesis, Durham University, 2002.

[40] Isaac L Chuang and Michael A Nielsen. Prescription for experimental determination of the dynamics of a quantum black box. *Journal of Modern Optics*, 44(11-12):2455–2467, 1997.

[41] Richard Cleve, Debbie Leung, Li Liu, and Chunhao Wang. Near-linear constructions of exact unitary 2-designs. *arXiv preprint arXiv:1501.04592*, 2015.

[42] Benoît Collins and Piotr Śniady. Integration with respect to the haar measure on unitary, orthogonal and symplectic group. *Communications in Mathematical Physics*, 264(3):773–795, 2006.

[43] A. Couvreur, N. Delfosse, and G. Zémor. A construction of quantum ldpc codes from cayley graphs. *IEEE Transactions on Information Theory*, 59(9):6087–6098, Sept 2013.

[44] A. Cross, G. Smith, J. A. Smolin, and B. Zeng. Codeword stabilized quantum codes. *IEEE Transactions on Information Theory*, 55(1):433–438, Jan 2009.

[45] Andrew W Cross, Easwar Magesan, Lev S Bishop, John A Smolin, and Jay M Gambetta. Scalable randomised benchmarking of non-clifford gates. *npj Quantum Information*, 2:16012, 2016.

[46] Christoph Dankert. Efficient simulation of random quantum states and operators. *arXiv preprint quant-ph/0512217*, 2005.

[47] Christoph Dankert, Richard Cleve, Joseph Emerson, and Etera Livine. Exact and approximate unitary 2-designs and their application to fidelity estimation. *Physical Review A*, 80(1):012304, 2009.

[48] Nicolas Delfosse and Naomi H Nickerson. Almost-linear time decoding algorithm for topological codes. *arXiv preprint arXiv:1709.06218*, 2017.

[49] Nicolas Delfosse and Gilles Zémor. Quantum erasure-correcting codes and percolation on regular tilings of the hyperbolic plane. In *Information Theory Workshop (ITW), 2010 IEEE*, pages 1–5. IEEE, 2010.

[50] Nicolas Delfosse and Gilles Zémor. Upper bounds on the rate of low density stabilizer codes for the quantum erasure channel. *Quantum Info. Comput.*, 13(9-10):793–826, September 2013.

[51] Nicolas Delfosse and Gilles Zémor. Linear-time maximum likelihood decoding of surface codes over the quantum erasure channel. *arXiv preprint arXiv:1703.01517*, 2017.

[52] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Journal of Mathematical Physics*, 43(9):4452–4505, 2002.

[53] David Deutsch and Richard Jozsa. Rapid solution of problems by quantum computation. In *Proc. R. Soc. Lond. A*, volume 439, pages 553–558. The Royal Society, 1992.

[54] Igor Devetak and Peter W Shor. The capacity of a quantum channel for simultaneous transmission of classical and quantum information. *Communications in Mathematical Physics*, 256(2):287–303, 2005.

[55] Irit Dinur. The pcp theorem by gap amplification. *Journal of the ACM (JACM)*, 54(3):12, 2007.

[56] David P DiVincenzo and Peter W Shor. Fault-tolerant error correction with efficient quantum codes. *Physical review letters*, 77(15):3260, 1996.

[57] Dominic Dotterrer, Tali Kaufman, and Uli Wagner. On expansion and topological overlap. *Geometriae Dedicata*, pages 1–11, 2016.

[58] Frédéric Dupuis. The decoupling approach to quantum information theory. *arXiv preprint arXiv:1004.1641*, 2010.

[59] Lior Eldar and Aram W Harrow. Local hamiltonians whose ground states are hard to approximate. In *Foundations of Computer Science (FOCS), 2017 IEEE 58th Annual Symposium on*, pages 427–438. IEEE, 2017.

[60] Lior Eldar, Maris Ozols, and Kevin F Thompson. The need for structure in quantum ldpc codes. *arXiv preprint arXiv:1610.07478*, 2016.

[61] Peter Elias. Coding for two noisy channels. In *Information Theory, Third London Symposium*, volume 67. London, England, 1955.

[62] Joseph Emerson, Robert Alicki, and Karol Życzkowski. Scalable noise estimation with random unitary operators. *Journal of Optics B: Quantum and Semiclassical Optics*, 7(10):S347, 2005.

[63] Kevin Eng, Thaddeus D Ladd, Aaron Smith, Matthew G Borselli, Andrey A Kiselev, Bryan H Fong, Kevin S Holabird, Thomas M Hazard, Biqin Huang, Peter W Deelman, et al. Isotopically enhanced triple-quantum-dot qubit. *Science Advances*, 1(4):e1500214, 2015.

[64] Paul Erds and Alfréd Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5:17–61, 1960.

[65] Michael Evans and Timothy Swartz. *Approximating integrals via Monte Carlo and deterministic methods*, volume 20. OUP Oxford, 2000.

[66] Shai Evra and Tali Kaufman. Bounded degree cosystolic expanders of every dimension. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2016, pages 36–48, New York, NY, USA, 2016. ACM.

[67] David Fattal, Toby S Cubitt, Yoshihisa Yamamoto, Sergey Bravyi, and Isaac L Chuang. Entanglement in the stabilizer formalism. *arXiv preprint quant-ph/0406168*, 2004.

[68] Richard P Feynman. Quantum mechanical computers. *Foundations of physics*, 16(6):507–531, 1986.

[69] Austin G Fowler, Matteo Mariantoni, John M Martinis, and Andrew N Cleland. Surface codes: Towards practical large-scale quantum computation. *Physical Review A*, 86(3):032324, 2012.

[70] 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.

[71] Alexey Frolov. An upper bound on the minimum distance of ldpc codes over gf (q). In *Information Theory (ISIT), 2015 IEEE International Symposium on*, pages 2885–2888. IEEE, 2015.

[72] Robert Gallager. Low-density parity-check codes. *IRE Transactions on information theory*, 8(1):21–28, 1962.

[73] Javier Garcia-Frias and Kejing Liu. Design of near-optimum quantum error-correcting codes based on generator and parity-check matrices of ldgm codes. In *Information Sciences and Systems, 2008. CISS 2008. 42nd Annual Conference on*, pages 562–567. IEEE, March 2008.

[74] Dmitry Gavinsky, Shachar Lovett, Michael Saks, and Srikanth Srinivasan. A tail bound for read-k families of functions. *Random Structures & Algorithms*, 47(1):99–108, 2015.

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

[76] Daniel Gottesman. The heisenberg representation of quantum computers. *arXiv preprint quant-ph/9807006*, 1998.

[77] Daniel Gottesman. Fault-tolerant quantum computation with constant overhead. *Quantum Information and Computation*, 14(15&16):1338–1371, Nov 2014.

[78] M. Grassl, A. Klappenecker, and M. Rotteler. Graphs, quadratic forms, and quantum codes. In *Information Theory, 2002. Proceedings. 2002 IEEE International Symposium on*, pages 45–, 2002.

[79] Antoine Grospellier, Anthony Leverrier, and Omar Fawzi. Efficient decoding of random errors for quantum expander codes. In *Journées Informatique Quantique 2017*, 2017.

[80] Daniel Harlow. The ryu–takayanagi formula from quantum error correction. *Communications in Mathematical Physics*, 354(3):865–912, 2017.

[81] Aram Harrow and Saeed Mehraban. Approximate t-designs by random quantum circuits with nearly optimal depth. *arXiv preprint*, 2018.

[82] Aram W Harrow and Richard A Low. Efficient quantum tensor product expanders and k-designs. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 548–561. Springer, 2009.

[83] Aram W Harrow and Richard A Low. Random quantum circuits are approximate 2-designs. *Communications in Mathematical Physics*, 291(1):257–302, 2009.

[84] AK Hashagen, ST Flammia, D Gross, and JJ Wallman. Real randomized benchmarking. *arXiv preprint arXiv:1801.06121*, 2018.

[85] Matthew B. Hastings. Decoding in hyperbolic spaces: Quantum ldpc codes with linear rate and efficient error correction. *Quantum Info. Comput.*, 14(13-14):1187–1202, October 2014.

[86] Matthew B. Hastings. Quantum codes from high-dimensional manifolds. In *Innovations in Theory of Computer Science (ITCS) 2017: 25:1-25:26*, August 2016.

[87] Patrick Hayden, Michał Horodecki, Andreas Winter, and Jon Yard. A decoupling approach to the quantum capacity. *Open Systems & Information Dynamics*, 15(01):7–19, 2008.

[88] Patrick Hayden and John Preskill. Black holes as mirrors: quantum information in random subsystems. *Journal of High Energy Physics*, 2007(09):120, 2007.

[89] M. Hein, W. Dür, J. Eisert, R. Raussendorf, M. Van den Nest, and H. . Briegel. Entanglement in Graph States and its Applications. *eprint arXiv:quant-ph/0602096*, February 2006.

[90] M. Hein, J. Eisert, and H. J. Briegel. Multiparty entanglement in graph states. *Phys. Rev. A*, 69:062311, Jun 2004.

[91] W. Helwig. *Multipartite Entanglement: Transformations, Quantum Secret Sharing, Quantum Error Correction*. PhD thesis, University of Toronto, 2014.

[92] Erik Hostens, Jeroen Dehaene, and Bart De Moor. Stabilizer states and clifford operations for systems of arbitrary dimensions and modular arithmetic. *Physical Review A*, 71(4):042315, 2005.

[93] Pavan Hosur, Xiao-Liang Qi, Daniel A Roberts, and Beni Yoshida. Chaos in quantum channels. *Journal of High Energy Physics*, 2016(2):4, 2016.

[94] Pavithran Iyer and David Poulin. Hardness of decoding quantum stabilizer codes. *IEEE Transactions on Information Theory*, 61(9):5209–5223, 2015.

[95] N Kahale. On the second eigenvalue and linear expansion. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 10:49–62, 1992.

[96] Nabil Kahale. Eigenvalues and expansion of regular graphs. *Journal of the ACM (JACM)*, 42(5):1091–1106, 1995.

[97] Volker Kaibel. On the expansion of graphs of 0/1-polytopes. In *The Sharpest Cut: The Impact of Manfred Padberg and His Work*, pages 199–216. SIAM, 2004.

[98] A Makhdoumi Kakhaki, H Karkeh Abadi, P Pad, H Saeedi, F Marvasti, and K Alishahi. Capacity achieving linear codes with random binary sparse generating matrices. *arXiv preprint arXiv:1102.4099*, 2011.

[99] Ivan Kassal, Stephen P Jordan, Peter J Love, Masoud Mohseni, and Alán Aspuru-Guzik. Polynomial-time quantum algorithm for the simulation of chemical dynamics. *Proceedings of the National Academy of Sciences*, 105(48):18681–18686, 2008.

[100] Harry Kesten. The critical probability of bond percolation on the square lattice equals 1/2. *Communications in mathematical physics*, 74(1):41–59, 1980.

[101] A. Yu. Kitaev, A. H. Shen, and M. N. Vyalyi. *Classical and Quantum Computation (Graduate Studies in Mathematics)*. Amer Mathematical Society, 5 2002.

[102] Alexei Yu Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, January 2003.

[103] Emanuel Knill and Raymond Laflamme. A theory of quantum error-correcting codes. *preprint*, 1995.

[104] Emanuel Knill, D Leibfried, R Reichle, J Britton, RB Blakestad, JD Jost, C Langer, R Ozeri, Signe Seidelin, and David J Wineland. Randomized benchmarking of quantum gates. *Physical Review A*, 77(1):012307, 2008.

[105] V. F. Kolchin. *Random Graphs (Encyclopedia of Mathematics and its Applications)*. Cambridge University Press, 1998.

[106] Alexey A. Kovalev, Ilya Dumer, and Leonid P. Pryadko. Design of additive quantum codes via the code-word-stabilized framework. *Phys. Rev. A*, 84:062319, Dec 2011.

[107] 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.

[108] Ilia Krasikov and Simon Litsyn. Estimates for the range of binomiality in codes' spectra. *IEEE Transactions on Information Theory*, 43(3):987–991, May 1997.

[109] Andrew J Landahl, Jonas T Anderson, and Patrick R Rice. Fault-tolerant quantum computing with color codes. *arXiv preprint arXiv:1108.5738*, 2011.

[110] Nima Lashkari, Douglas Stanford, Matthew Hastings, Tobias Osborne, and Patrick Hayden. Towards the fast scrambling conjecture. *Journal of High Energy Physics*, 2013(4):22, 2013.

[111] Felix Leditzky, Debbie Leung, and Graeme Smith. Quantum and private capacities of low-noise channels. *arXiv preprint arXiv:1705.04335*, 2017.

[112] Richard A Low. Pseudo-randomness and learning in quantum computation. *arXiv preprint arXiv:1006.5227*, 2010.

[113] Alexander Lubotzky. Ramanujan complexes and high dimensional expanders. *Japanese Journal of Mathematics*, 9(2):137–169, 2014.

[114] Alexander Lubotzky, Ralph Phillips, and Peter Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.

[115] Michael G Luby, Michael Mitzenmacher, Mohammad Amin Shokrollahi, and Daniel A Spielman. Efficient erasure correcting codes. *IEEE Transactions on Information Theory*, 47(2):569–584, 2001.

[116] D. J. C. MacKay, G. Mitchison, and P. L. McFadden. Sparse Graph Codes for Quantum Error-Correction. *eprint arXiv:quant-ph/0304161*, April 2003.

[117] Florence Jessie MacWilliams and Neil James Alexander Sloane. *The theory of error correcting codes*. North-Holland mathematical library. North-Holland Pub. Co. New York, Amsterdam, New York, 1977.

[118] Easwar Magesan, Jay M Gambetta, and Joseph Emerson. Characterizing quantum gates via randomized benchmarking. *Physical Review A*, 85(4):042311, 2012.

[119] Grigorii Aleksandrovich Margulis. Explicit group-theoretical constructions of combinatorial schemes and their application to the design of expanders and concentrators. *Problemy peredachi informatsii*, 24(1):51–60, 1988.

[120] Rawad Mezher, Joe Ghalbouni, Joseph Dgheim, and Damian Markham. Efficient quantum pseudorandomness with simple graph states. *Physical Review A*, 97(2):022333, 2018.

[121] Yoshifumi Nakata, Christoph Hirche, Ciara Morgan, and Andreas Winter. Unitary 2-designs from random x-and z-diagonal unitaries. *Journal of Mathematical Physics*, 58(5):052203, 2017.

[122] Michael A Nielsen. A simple formula for the average gate fidelity of a quantum dynamical operation. *Physics Letters A*, 303(4):249–252, 2002.

[123] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 10 anv edition, 1 2011.

[124] Alon Nilli. On the second eigenvalue of a graph. *Discrete Mathematics*, 91(2):207–210, 1991.

[125] Joe O'Gorman and Earl T Campbell. Quantum computation with realistic magic-state factories. *Physical Review A*, 95(3):032338, 2017.

[126] Don N Page. Average entropy of a subsystem. *Physical review letters*, 71(9):1291, 1993.

[127] G Massimo Palma, Kalle-Antti Suominen, and Artur K Ekert. Quantum computers and dissipation. In *Proc. R. Soc. Lond. A*, volume 452, pages 567–584. The Royal Society, 1996.

[128] Lluis Pamies-Juarez, Cyril Guyot, and Robert Mateescu. Spider codes: Practical erasure codes for distributed storage systems. In *Information Theory (ISIT), 2016 IEEE International Symposium on*, pages 1207–1211. IEEE, 2016.

[129] Ori Parzanchevski. Mixing in high-dimensional expanders. 2013.

[130] Ori Parzanchevski, Ron Rosenthal, and Ran J. Tessler. Isoperimetric inequalities in simplicial complexes. *Combinatorica*, 36(2):195–227, 2016.

[131] Fernando Pastawski, Beni Yoshida, Daniel Harlow, and John Preskill. Holographic quantum error-correcting codes: Toy models for the bulk/boundary correspondence. *Journal of High Energy Physics*, 2015(6):149, 2015.

[132] Henry D Pfister, Igal Sason, and Rudiger Urbanke. Capacity-achieving ensembles for the binary erasure channel with bounded complexity. *IEEE Transactions on Information Theory*, 51(7):2352–2379, 2005.

[133] Hossein Pishro-Nik and Faramarz Fekri. On decoding of low-density parity-check codes over the binary erasure channel. *IEEE Transactions on Information Theory*, 50(3):439–454, 2004.

[134] John Preskill. Fault-tolerant quantum computation. In *Introduction to quantum computation and information*, pages 213–269. World Scientific, 1998.

[135] Eric M Rains. Quantum weight enumerators. *IEEE Transactions on Information Theory*, 44(4):1388–1394, 1998.

[136] Omer Reingold. Undirected connectivity in log-space. *Journal of the ACM (JACM)*, 55(4):17, 2008.

[137] Omer Reingold, Salil Vadhan, and Avi Wigderson. Entropy waves, the zig-zag graph product, and new constant-degree expanders. *Annals of mathematics*, pages 157–187, 2002.

[138] Thomas J Richardson, Mohammad Amin Shokrollahi, and Rüdiger L Urbanke. Design of capacity-approaching irregular low-density parity-check codes. *IEEE transactions on information theory*, 47(2):619–637, 2001.

[139] Massimiliano F Sacchi. Optimal discrimination of quantum operations. *Physical Review A*, 71(6):062340, 2005.

[140] Pradeep Sarvepalli and Andreas Klappenecker. Degenerate quantum codes and the quantum hamming bound. *Physical Review A*, 81(3):032318, 2010.

[141] Igal Sason. On universal properties of capacity-approaching ldpc code ensembles. *IEEE Transactions on Information Theory*, 55(7):2956–2990, 2009.

[142] Igal Sason and Rüdiger Urbanke. How sparse can parity-check matrices of binary linear codes be, as a function of their gap to capacity? In *PROCEEDINGS OF THE ANNUAL ALLERTON CONFERENCE ON COMMUNICATION CONTROL AND COMPUTING*, volume 40, pages 1140–1149. The University; 1998, 2002.

[143] Igal Sason and Rudiger Urbanke. Parity-check density versus performance of binary linear block codes over memoryless symmetric channels. *IEEE Transactions on Information Theory*, 49(7):1611–1635, 2003.

[144] Philipp Schindler, Daniel Nigg, Thomas Monz, Julio T Barreiro, Esteban Martinez, Shannon X Wang, Stephan Quint, Matthias F Brandl, Volckmar Nebendahl, Christian F Roos, et al. A quantum information processor with trapped ions. *New Journal of Physics*, 15(12):123012, 2013.

[145] D. Schlingemann and R. F. Werner. Quantum error-correcting codes associated with graphs. *Phys. Rev. A*, 65:012308, Dec 2001.

[146] Claude Elwood Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55, 2001.

[147] M Amin Shokrollahi. New sequences of linear time erasure codes approaching the channel capacity. In *International Symposium on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes*, pages 65–76. Springer, 1999.

[148] Peter Shor and Raymond Laflamme. Quantum macwilliams identities. *arXiv preprint quant-ph/9610040*, 1996.

[149] Peter W Shor. Scheme for reducing decoherence in quantum computer memory. *Physical review A*, 52(4):R2493, 1995.

[150] Peter W Shor. Fault-tolerant quantum computation. In *Foundations of Computer Science, 1996. Proceedings., 37th Annual Symposium on*, pages 56–65. IEEE, 1996.

[151] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2):303–332, 1999.

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

[153] Andrew M. Steane. Error correcting codes in quantum theory. *Phys. Rev. Lett.*, 77(5):793–797, Jul 1996.

[154] Andrew M Steane. Efficient fault-tolerant quantum computing. *Nature*, 399(6732):124, 1999.

[155] Ashley M Stephens. Fault-tolerant thresholds for quantum error correction with the surface code. *Physical Review A*, 89(2):022321, 2014.

[156] Oleg Szehr, Frédéric Dupuis, Marco Tomamichel, and Renato Renner. Decoupling with unitary approximate two-designs. *New Journal of Physics*, 15(5):053022, 2013.

[157] R Michael Tanner. Explicit concentrators from generalized n-gons. *SIAM Journal on Algebraic Discrete Methods*, 5(3):287–293, 1984.

[158] 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.

[159] William G Unruh. Maintaining coherence in quantum computers. *Physical Review A*, 51(2):992, 1995.

[160] Joel J Wallman and Steven T Flammia. Randomized benchmarking with confidence. *New Journal of Physics*, 16(10):103032, 2014.

[161] Zak Webb. The clifford group forms a unitary 3-design. *arXiv preprint arXiv:1510.02769*, 2015.

[162] Stephen Wiesner. Conjugate coding. *ACM Sigact News*, 15(1):78–88, 1983.

[163] William K Wootters and Wojciech H Zurek. A single quantum cannot be cloned. *Nature*, 299(5886):802–803, 1982.

[164] Chengxian Zhang, Xu-Chen Yang, and Xin Wang. Leakage and sweet spots in triple-quantum-dot spin qubits: a molecular orbital study. *arXiv preprint arXiv:1711.06418*, 2017.