On Expansion, High-Dimensional Expanders, and Applications in Coding Theory

Ijay Narang [*] Advisor: Pedro Paredes[†]

October 17, 2024

Abstract

Communication protocols and data storage necessitate the use of error-correcting codes, which inject redundancy into a bit-string to make it resilient against corruption. Desirable properties of such codes include distance (ensuring recoverability of the original message) and rate (capturing transmission efficiency). It is well known that expander graphs—sparse yet highly connected graphs—can be used to construct codes that achieve strong trade-offs between these parameters. Furthermore, the higher dimensional analog of expanders, high dimensional expanders, have also found useful applications in coding theory

Thus, this thesis studies this intersection of expanders, high dimensional expanders, and coding theory. Thematically, this paper can be viewed in two parts: the first is a survey of classical and recent ideas in this area, and the second presents our contributions, which are new combinatorial constructions of expanding square and cubical complexes.

The survey component of this paper begins by introducing foundational concepts from both areas. We begin by exploring simple trade-offs between distance and rate via the Singleton bound and Gilbert-Varshamov bound. We then discuss key constructions in coding theory such such as Hadamard codes, Reed-Solomon codes, Reed-Muller codes, and tensor codes—and discuss known trade-offs among parameters like distance, rate, and locality. Following this, we explore the spectral and combinatorial properties of expander graphs and how they enable the construction of codes with good parameters. This naturally leads to a discussion of high-dimensional expanders (HDXs), including square and cubical complexes, and how they have been used to build codes that have constant rate, distance, and locality (in particular, the codes of [DEL+22]).

The second component of the paper presents our original contributions: new combinatorial constructions of expanding square and cubical complexes. For our construction of square complexes, we are able to characterize the expansion of classical and parallel random walks. For cubical complexes, we propose a general construction in arbitrarily high dimensions and examine its structural features, potential expansion properties, and the technical challenges it raises.

[*]Princeton University. in5787@princeton.edu.
[†]Princeton University. pparedes@cs.princeton.edu.

Contents

1 Introduction

1.1 Structure

This thesis focuses on the interplay between expanders, high dimensional expanders, and coding theory. Excluding this introduction, it is comprised of five sections and meant to be fully self-contained. At a high-level, the paper is structured as follows:

- **Section 2: Preliminaries** focuses on introducing the reader to basic definitions and terminology in coding theory and spectral graph theory. To motivate our interest in spectral graph theory, we demonstrate its useful-ness in characterizing line graphs. For coding theory, we show the Hadamard code, and conduct a brief analysis of it.
- **Section 3: Classical Results in Coding Theory** provides a (non-exhaustive) overview of classical results in coding theory. In particular, we cover some known trade-offs of code parameters, and discuss important constructions of codes (Reed-Solomon codes, Reed-Muller codes, and Tensor codes). Lastly, we motivate the importance of locality in codes and analyze the simple example of Hadamard codes.
- **Section 4: Expander Graphs and Applications to Coding Theory** is our first glance of the interplay between spectral graph theory and coding theory. In particular, we consider expander graphs (graphs that are both sparse and connected) and study their spectral properties. We then show how to construct codes with good parameters from them, and lastly remark on the locality of such codes.
- **Section 5: High Dimensional Expanders** This section demonstrates how to generalize expanders to hypergraphs via defining High Dimensional Expanders (HDXs). Though there are multiple definitions of expanders, we focus on classical HDXs, square complexes, and cubical complexes. Our motivation for studying square complexes comes from [DEL$^+$22], which used a square complex to construct the first known codes with good locality and code parameters. We present an overview of their work. We also discuss cubical complexes (higher dimensional generalizations of square complexes) which were used by [DLV24] to construct almost good locally testable quantum LTCs.
- **Section 6: Combinatorial Constructions of Expanding Complexes** This section primarily focuses on our contributions. We begin with a short discussion of [Gol21], which presents a combinatorial construction of classical HDXs. Our work primarily focuses on combinatorial constructions of Square and Cubical Complexes. To that end, we first present a combinatorial construction for square complexes and fully characterize its spectrum for the classical random walk and parallel random walk. Secondly, we provide a construction for arbitrarily high dimensional cubical complexes. We suggest some reasons why this cubical complex construction should have good expansion, and identify some of its properties/technical roadblocks.

1.2 Motivation

In some sense, the above outline is structured to track the historical development of the interplay between coding theory and expansion. This means that much of this thesis (Sections 2-5) act as

a survey of developments in coding theory. Section 6 is thematically different in that it is a 'mini paper'. In Section 6.1, we first discuss a known result ([Gol21]) that inspired our methodology and then our two main contributions. The first, termed the *Generalized Corner Product*, takes as input two graphs, and outputs a square complex that has good 1-skeleton expansion and constant parallel expansion. In demonstrating its instantiation, we discuss its behavior when the input graphs are both Ramanujan expanders. Our second contribution is a modification of [Gol21] to function for cubical complexes. While we have not fully characterized the behavior of this construction, it is not unreasonable to believe that it will exhibit constant expansion for all parallel random walks. We discuss some of its characteristics and our (incomplete) work towards characterizing its spectrum.

Though Section 6 serves as a discussion of our work, Sections 2 through 5 act as a survey on Expanders, Coding Theory, and High Dimensional Expanders. We now motivate why this intersection is interesting and its core motivations/applications.

Coding theory studies the question of data storage and communication across noisy channels. For example, suppose Alice wants to send the message 1 to Bob across a noisy channel. As bits can be flipped and deleted, Bob may receive an entirely different message, rendering communication impossible. A natural scheme is to use what is called a *repetition code* where Alice instead sends the code 11111 and Bob simply decodes it by taking the majority bit. So, even if 2 bits are flipped (Bob maybe receives 10101), Bob is able to recover the original message by noticing that more bits are 1.

The key idea underlying the *repetition code* is adding redundancy, as this makes our message more susceptible to noise. However, this has an obvious issue – we are being very wasteful with space. If we want to transmit a very, very large message, then repeating each bit 5 times could cause space and runtime issues. Surely, there must be a trade-off between these notions of "redundancy" and "efficiency." Indeed, this question is one of the cornerstones of coding theory, which informally, is the study of designing codes (families of bit-strings) that have good parameters. These parameters include distance (which intuitively measures how the susceptibility of a code to noise) and rate (which intuitively measures how the efficiency of a code).

Let's return to a slightly more advanced version of the communication game between Alice and Bob. This time, Alice wants to transmit any bit-string of length 2 to Bob over a noisy channel. Being an efficient individual, Alice does not want to waste bits, and thus wants to design a code that has good rate. One solution is for Alice to append a (third) bit to her message which is equal to the sum of the bits of her previous message. So, if Alice's original message was 11, then she would send 110. Such a code is also able to withstand error, as Bob can check if the first two bits sum to the third in order to ensure message accuracy. This special "third bit" is a parity check, and is critical towards building error correcting codes. Note that the code used by Alice and Bob mandates that all code-words are of the form $x_1x_2(x_1 \oplus x_2)$, which is equivalent to saying that all 3 bits $x_1x_2x_3$ sum to 0. Thus, parity checks can be viewed as a constraint on the bits of a code-word. Some constraints are simple (like the code used by Alice) whereas others are a bit more complex (such as mandating that ordered subset of bits follow a specific pattern). Regardless, most codes can be viewed as a set of parity checks which lends to useful alternative views of codes. Codes can be designed in a variety of manners: based on combinatorial structure, polynomials, or tensor products. In Section 3, we see some examples of well-known codes and analyze their rate and distance.

One view of error correcting code is combinatorial. Consider a code of length n with m parity checks. We now create a bipartite graph $G = (L \cup R, E)$ where the left partition (L) has n vertices

and the right partition has m vertices. Because all code-words are n bits long, we are able to place the i^{th} bit of each code-word on the i^{th} vertex of the left partition. Furthermore, we connect vertices on the left side to the j^{th} vertex on the right side only if they are involved in the j^{th} parity check.

Under this view, the placement and amount of edges must be selected strategically. If there are too few edges, then our code will be very unconstrained, so it will have bad distance (as code-words could be close). If there are too many edges, then there will not be many feasible code-words (and therefore the rate will be bad). In essence, we must construct the bipartite graph G so that it is simultaneously sparse and well-connected – two properties at odds with each other. Luckily, such graphs exist, and are called expanders. Utilizing a bipartite expander as the parity check graph yields *expander codes* with good parameters. Through using a more sophisticated expander-based parity check, we attain Tanner codes. The study of expander graphs, expander codes and Tanner codes is the subject of Section 4

Let's now utilize an expander-based code towards the communication problem between Alice and Bob. Such a scheme would definitely work, but would have issues in more resource-intensive systems. Frequently, communication occurs over some large distributed system in which we need to transmit messages (bit-strings) of very large order. A common practice in such systems is that whenever a server believes some message to be incorrect or corrupted, it requests the message to be re-sent. Note that this means a server must (1) read the whole bit-string, (2) determine if it is incorrect, and if necessary (3) request the message again and repeat the process. Clearly this is a lengthy process (and potential bottleneck) and necessitates some clever optimization.

Ideally, we could solve the above issue by somehow detecting if a bit-string is correct or incorrect without having to read every single bit. Codes which have this property are called local testable codes (LTCs). For any LTC, we can determine a bit-string's validity (whether it is in the code or not) by sampling some small number c of bits. We can then "test" these c bits and determine with high probability if a code-word is valid or not. Unfortunately, designing a code with good locality is difficult, and many codes with good distance and rate are not good LTCs.

Having seen the utility of expansion on graphs, it seems that we may want to turn back to these objects to construct codes with good locality, rate, and distance. Unfortunately, an arbitrary expander is not locally testable. However, it turns out that the higher-dimensional generalization of expanders (High Dimensional expanders, or HDXs) can be utilized to create codes with good rate, distance, and locality. Indeed, Dinur et al ([DEL$^+$22]) utilized a special case of high dimensional expanders called square complexes to construct these codes. This code, and HDXs are the central topic of Section 5. Intuitively, a classical HDX can be thought of as a generalization of expander graphs to hypergraphs satisfying downward closure (i.e. if a set belongs to an HDX, then so do all its subsets). The notion of expansion on HDXs view 'links', which can be thought of as a higher dimensional analog of 'neighborhoods' in graphs. This definition enables the local-to-global paradigm of High Dimensional Expanders (HDXs). That is, if an HDX has good local expansion (expanding links), then the HDX will have global expansion as a whole. We give special attention to the cases of square and cubical complexes, as they can directly be applied to instantiate the codes of [DEL$^+$22] and [DLV24].

As previously mentioned, Section 6 focuses on our contributions, which we articulate upon in the next subsection.

1.3 Our Contributions

Most known constructions of High Dimensional Expanders, Square Complexes, and Cubical complexes are algebraic in nature. Our main contributions are on **combinatorial constructions** of square and cubical complexes. Such constructions (although more limited) can provide us with a better understanding of these objects, and also allow us to exercise more control on their structure.

To (informally) summarize our contributions, we describe the basics of square and cubical complexes. A square complex is a generalization of a graph with cells of up to 2 dimensions: 0-dimensional cells are vertices, 1-dimensional cells are edges, and 2-dimensional cells are 4-cycles or *squares*, hence the name (we will formalize this in the next section.) Cubical complexes are generalizations of these to even higher dimensions, where an i-dimensional cell corresponds to a i-dimensional hypercube. In order to define the notion of expansion we will study, we first need to define the concept of parallel adjacency of edges. Two edges are parallel if they don't share any endpoints and there is a square that contains both of them. Now, the parallel adjacent edges to a given edge are given by the set of edges parallel to the given one. Naturally, this parallel adjacency defines a new graph, where vertices are the edges of the complex and the edges are given by parallel adjacency. We can now define two types of expansion of a square complex: the global expansion, which is given by the spectral expansion of the underlying graph (ignoring the squares) and the parallel expansion, which is given by the spectral expansion of the graph defined by the parallel adjacencies. We remark that this particular definition of expansion is the one used in [DEL$^+$22], but we there are other related ones in the literature.

Our main contribution is an explicit construction of square complexes with non trivial global expansion and constant parallel expansion. Specifically, we prove:

Theorem 1.1 (Informal, see Theorem 6.3 for full result)**.** *For any $d > 0$, there exists an explicit infinite family of square complexes with vertex regularity $(D^{|})$ and edge regularity $(D^{\square})$ both $\mathcal{O}(d^3)$, global expansion $\Theta((D^{|})^{3/4})$ and a constant parallel expansion of $\frac{1}{2}$.*

We obtain this by defining a product between two graphs called the *corner product*, and showing that if the two graphs are good expanders then the square complex resulting from their product has both good global and parallel expansion. With the appropriate choice of parameters we obtain the result above. See Theorem 6.3 for the full result.

Our second contribution is on a generalization to cubical complexes. That is, we provide a combinatorial construction of cubical complexes from graphs that we suspect could have constant expansion for all parallel random walks. In Section 6.2.2, we discuss this construction and make partial progress towards fully characterizing its expansion. We identify that the construction's main roadblock arises from a natural asymmetry in its construction. This second construction is a cubical complex variant of the HDX construction of [Gol21].

2 Preliminaries

2.1 Primer on Spectral Graph Theory

Graphs are natural models of relationships between objects, and thus have a plethora of applications. Associated with graphs are graph matrices that enable the application of spectral methods (linear-algebraic tools) towards further understanding graph properties. Arguably, the two most studied graph matrices are the adjacency matrix and Laplacian, which are defined below.

Definition 2.1 (Adjacency Matrix and Laplacian Matrix)**.** The *adjacency matrix* $A \in \mathbb{R}^{n\times n}$ of the graph G is defined as:

$$A_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

Additionally, define $D \in \mathbb{R}^{n\times n}$ be the *degree matrix*, a diagonal matrix defined by $D_{ii} = \deg(i) = \sum_{j=1}^{n} A_{ij}$. The *Laplacian matrix* $L \in \mathbb{R}^{n\times n}$ of the graph is given by $L = D - A$.

To a reader unfamiliar with the applications of these matrices, we demonstrate how they characterize combinatorial properties such as walks and connectivity in a graph. In particular:

- The adjacency matrix A encodes walk counts: for any integer $k \geqslant 1$, the entry $(A^k)_{ij}$ gives the number of walks of length k from vertex i to vertex j. The trace $\mathrm{Tr}(A^k) = \sum_{i=1}^{n}(A^k)_{ii}$ thus counts the total number of closed walks of length k in the graph.
- The Laplacian matrix $L = D - A$ captures connectivity through its quadratic form. For any vector $\mathbf{x} \in \mathbb{R}^n$, we have $\mathbf{x}^\top L\mathbf{x} = \sum_{(i,j)\in E}(x_i - x_j)^2$. In particular, if $\mathbf{x} = \mathbf{1}_S$ is the indicator vector of a subset $S \subseteq V$, then $\mathbf{1}_S^\top L\mathbf{1}_S = |\{(i,j) \in E : i \in S,\, j \notin S\}|$, the size of the cut between S and its complement.

Central to spectral graph theory is analyzing the spectrum of graph matrices; importantly, recall the following fundamental fact from linear algebra:

Theorem 2.2 (Spectral Theorem)**.** *Let $M \in \mathbb{R}^{n\times n}$ be a real symmetric matrix. Then all eigenvalues of M are real, and there exists an orthonormal basis of $\mathbb{R}^n$ consisting of eigenvectors of M. Equivalently, M is diagonalizable by an orthogonal matrix: there exists $Q \in \mathbb{R}^{n\times n}$ with $Q^\top Q = I$ such that*

$$M = Q\Lambda Q^\top,$$

where Λ is a diagonal matrix whose entries are the eigenvalues of M.

Thus, both A and its Laplacian matrix $L = D - A$, have all eigenvalues real. Furthermore, it is easy to see that the Laplacian matrix is positive semi-definite, as for any vector $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{x}^\top L\mathbf{x} = \sum_{(i,j)\in E}(x_i - x_j)^2 \geqslant 0$. An interesting application of these tools is the case of Line Graphs, which we show in the next section.

2.2 Spectra of Line Graphs

Every graph G, has a line graph $l(G)$ where $V(l(G)) = E(G)$ and $E(l(G)) = \{(u, v) : u \in E(G), v \in E(G), |u \cap v| = 1\}$. That is, u and v are adjacent in $l(G)$ if and only if they are adjacent edges in G. The spectra of a graph G and its line graph $l(G)$ are similar, as characterized by the following lemma from [Spi19]. For completeness, we include a self-contained proof of it.

Lemma 2.3 ([Spi19])**.** *Let G be a d-regular graph with n vertices, and let H be its line graph. Then the spectrum of the Laplacian of H is the same as the spectrum of the Laplacian of G, except that it has $\frac{dn}{2} - n$ extra eigenvalues of $2d$.*

Proof. Define U to be the signed edge-vertex adjacency matrix of a graph G. That is, U is a $E(G) \times V(G)$ matrix where each entry takes on value:

$$U((a, b), c) = \begin{cases} 1 & \text{if } a = c \\ -1 & \text{if } b = c \\ 0 & \text{otherwise} \end{cases}.$$

Note that the Laplacian L_G can be expressed as $U_G^\intercal U_G$. Denote the graph $|U|$ to be the $E(G) \times V(G)$ matrix where each element $|U|_{ij} = |U_{ij}|$. The matrix $|U|^\intercal |U|$ is equal to the Laplacian, except that its off-diagonal entries are 1 instead of -1. So, we have that $|U|^T |U| = D_G + M_G = dI + M_G$

Now, consider $|U||U|^T$. This is a matrix with $\frac{nd}{2}$ rows and $\frac{nd}{2}$ columns, indexed by edges of G. Observe that the entry at the intersection of row (u, v) and column (w, z) takes on value 2 if they are the same edge, 1 if they share a vertex, and 0 otherwise. That is $|U||U|^T = 2I_{\frac{nd}{2}} + M_H$. Thus, $|U||U|^T$ and $|U|^T|U|$ have the same eigenvalues (besides for the $\frac{nd}{2} - n$ extra eigenvalues of for $|U|^T|U|$).

Suppose that λ_i is an eigenvalue of $L_G = dI - M_G$. Then, $2d - \lambda_i$ is an eigenvalue of $D_G + M_G = |U|^\intercal |U|$ and thus an eigenvalue of $|U|^\intercal |U| = 2I_{\frac{nd}{2}} + M_H$. It follows that $2d - \lambda_i - 2$ is an eigenvalue of M_H. Because H is $(2d - 1)$-regular, we attain that λ_i is an eigenvalue of $D_H - M_H = L_H$.

By identical reasoning, the extra $\frac{dn}{2} - n$ zero eigenvalues of $2I_{\frac{nd}{2}} + M_H$ map to $2d$ in L_H, yielding the desired result. □

Corollary 2.4. *Let G be a d-regular graph with n vertices, and denote the un-normalized and normalized adjacency matrices of G to be M_G and $\tilde{M}_G$ respectively. Let H be the line graph of G and similarly, denote the un-normalized and normalized adjacency matrices of H to be M_H and $\tilde{M}_H$. If λ is the second largest eigenvalue of $\tilde{M}_G$, then the second largest eigenvalue of $\tilde{M}_H$ is $\frac{d+d\lambda-2}{2d-2}$.*

Proof. H is $2(d - 1)$ regular, so its (un-normalized) Laplacian can be expressed as $2(d - 1)I - M_H$. Now, denote γ to be an eigenvalue of M_H. It must follow that $d - \gamma$ is an eigenvalue of L_G and (by Lemma 2.3) an eigenvalue of L_H. Therefore, $2(d - 1) - (d - \gamma) = d + \gamma - 2$ is an eigenvalue of M_H. So, we have that α being an eigenvalue of $\tilde{M}_G$ implies that $\frac{d+d\alpha-2}{2d-2}$ is an eigenvalue of $\tilde{M}_H$.

Note that (by Lemma 2.3), we have that all eigenvalues of L_H are either eigenvalues of L_G or have value $2d$. All the eigenvalues of $2d$ of L_H correspond to eigenvalues of $-2/(2d - 1)$ of $\tilde{M}_H$. It follows that the second largest eigenvalue of $\tilde{M}_H$ is $\frac{d+d\lambda-2}{2d-2}$. □

2.3 Basic Properties of Coding Theory

An error correcting code is formally defined as $\mathcal{C} = \{c : c \in \Sigma^n\}$ where Σ is a finite alphabet (typically a finite field) and n is the length of the code. Each c corresponds to an element in $\mathcal{C}$ called a code-word. We refer to the length of our alphabet as $|\Sigma| = q$, and for the purposes of this paper, restrict our attention to the binary case ($q = 2$). For each code $\mathcal{C}$, we associate an encoding map G which takes in a message and outputs the corresponding code-word. Note that G maps each message m in the set of messages $\mathcal{M}$ to a code-word c in $\mathcal{C}$.

Each code is associated with parameters that measure its quality and efficiency. The first parameter is its rate $R(\mathcal{C})$, which is a fraction that measures its redundancy. $R(\mathcal{C}) = \frac{k}{n}$, where $k = \frac{\log|\mathcal{C}|}{\log|\Sigma|}$ is the dimension of $\mathcal{C}$. The second parameter is the minimum distance. We refer to the Hamming distance between two code-words c_1, c_2 as $\Delta(c_1, c_2)$, the number of bits that c_1 and c_2 differ. Thus, we can define the distance of a code $\mathcal{C}$, as $\Delta(\mathcal{C}) = \min_{c_1 \neq c_2 \in \mathcal{C}} \Delta(c_1, c_2)$. The relative distance of a code is the normalized distance $\delta(\mathcal{C}) = \frac{\Delta(\mathcal{C})}{n}$. We also define the hamming weight of a code-word $wt(c)$ as the number of non-zero bits in the code-word and the weight of a code, $wt(\mathcal{C})$, as the minimum hamming weight across all code-words in $\mathcal{C}$.

An important class of codes are linear codes. A code $\mathcal{C}$ is linear if any linear combination of the code words $c_1, c_2, ..., c_k$, (where $k = \dim(\mathcal{C})$) is also a code-word. Let $[n, k, d]_q$ denote a q-ary linear code with block length n, dimension k, and minimum distance d. For a linear code $\mathcal{C}$ of dimension k, a matrix $G \in \mathbb{F}_q^{n \times k}$, where $\mathbb{F}_q$ is the finite field of length q, is a generator matrix for $\mathcal{C}$ if its k columns span $\mathcal{C}$. So, a linear code $\mathcal{C}$ can be viewed as the image of its generator matrix. We can alternatively define a code by its parity check matrix H, which is the null space of G. So, we can express $\mathcal{C} = \{c \in \mathbb{F}_q^n \mid Hc = 0\}$. In the special case of binary, linear codes, we have the following well-known Lemma:

Lemma 2.5. *If $\mathcal{C}$ is a linear code, then its minimum distance $\Delta(\mathcal{C})$ is equal to the minimum weight $wt(\mathcal{C})$ over all non-zero code-words.*

Proof. Let c denote the minimum non-zero weight code-word in $\mathcal{C}$. By definition $0 \in \mathcal{C}$, and so $\Delta(\mathcal{C}) \leqslant \Delta(0, c) = wt(c) = wt(\mathcal{C})$. All that remains is to show that $wt(\mathcal{C}) \leqslant \Delta(\mathcal{C})$. We can assume, for the sake of contradiction, that it is not. It must then follow that there exists two non-zero code-words c_1 and c_2 such that $\Delta(c_1, c_2) < wt(\mathcal{C})$. Note that $\Delta(c_1, c_2) = wt(c_1 - c_2) \implies wt(c_1 - c_2) < wt(\mathcal{C})$, which is a contradiction. □

2.4 Warm-up: Hadamard Codes

As a warm-up, we start with perhaps the simplest of error correcting codes, which can be easily understood by its encoding map.

Definition 2.6 (Hadamard Code)**.** Let $y^{(k)}$ denote the binary representation of the integer $k \in \{0, 1, 2, \cdots 2^m - 1\}$ on m bits. The Hadamard encoding is a function $f : \mathbb{F}_2^m \to \mathbb{F}_2^{2^m}$ where the k^{th} bit of $f(x)$, (which we denote $f(x)_k$) is given by the inner product $\langle x, y_k \rangle$ over $\mathbb{F}_2$.

The Hadamard code has excellent distance, which we prove in the following Lemma.

Lemma 2.7. *Denote the Hadamard code given by $f : \mathbb{F}_2^m \to \mathbb{F}_2^{2^m}$ to be $\mathcal{C}_m^{Had}$. It then follows that $\delta(\mathcal{C}_m^{Had}) = \frac{1}{2}$.*

Proof. Note that by construction, $\mathcal{C}_m^{\text{Had}}$ is linear, so by Lemma 2.1, it suffices to show $wt(\mathcal{C}_m^{\text{Had}}) = 2^{m-1}$. Expressing the relative weight of a codeword probabilistically yields:

$$\mathbf{Pr}_{k\in\{0,1,2,\cdots,2^m-1\}}\left[(f(x))_k = 1\right] = \mathbf{Pr}_{k\in\{0,1,2,\cdots,2^m-1\}}\left[\langle x, y^{(k)}\rangle = 1\right] = \frac{1}{2} \quad \forall x.$$

The first equality follows from the definition of a Hadamard Code. The second equality follows from the fact that $\mathbb{E}[\langle x, y^{(k)}\rangle = 1] = \mathbb{E}[\sum_i x_i y_1^{(k)} \mod 2] = \frac{1}{2}$ and the only values that $\langle x, y^{(k)}\rangle$ over $\mathbb{F}_2$ can take are 1 and 0. As every code-word has relative weight $\frac{1}{2}$, we have our desired result. □

Clearly, the rate of the Hadamard code is $\frac{m}{2^m}$. It is also worth noting that the Hadamard code can also be defined over $\mathbb{F}_q$, by encoding a message in $\mathbb{F}_q^k$ with its dot product with every vector in $\mathbb{F}_q^k$.

3 Classical Results in Coding Theory

3.1 The Singleton Bound and Reed-Solomon Codes

Note that the rate and distance are opposing parameters. Indeed, there are many results that characterize the rate-distance tradeoff such as Gilbert-Varshamov bound, Plotkin bound etc. We present the Singleton Bound, and its tightness exhibited by the Reed-Solomon code (one of the most fundamental polynomial codes) in this section. We begin with the following general observation:

Lemma 3.1. *Let C be a code of block length n and distance d over an alphabet of size q. Then $|C| \leqslant q^{n-d+1}$.*

Proof. Assume (for the sake of contradiction) that $|C| > q^{n-d+1}$. By the pigeonhole principle, there exists $c_1, c_2 \in C$, with $c_1 \neq c_2$, that have the first $n-d+1$ bits in common. But, then $\Delta(c_1, c_2) \leqslant d-1 < d$, a contradiction. □

This gives an alphabet-independent asymptotic upper bound on the rate as a function of relative distance.

Corollary 3.2 (Singleton Bound)**.** *Given any code $\mathcal{C}$, its rate R and relative distance δ satisfy*

$$R \leqslant 1 - \delta + o(1).$$

Reed-Solomon codes are quite popular in practice, and are the data-safety mechanism underlying CDs and DvDs. A more modern application is in data centers. These codes are defined (typically over large fields) as follows.

Definition 3.3 (Reed-Solomon codes)**.** For integers $1 \leqslant k < n$, a field $\mathbb{F}$ of size $|\mathbb{F}| \geqslant n$, and a set $S = \{\alpha_1, \ldots, \alpha_n\} \subseteq \mathbb{F}$, the Reed-Solomon code is given by

$$\mathrm{RS}_{\mathbb{F},S}[n,k] = \{(p(\alpha_1), p(\alpha_2), \ldots, p(\alpha_n)) \in \mathbb{F}^n \mid p \in \mathbb{F}[X] \text{ is a polynomial of degree } \leqslant k-1\}$$

In other words, the Reed-Solomon code is the evaluations of all polynomials of degree $\leqslant k-1$ on the set S. This is easy to see by observing that the generator matrix for the Reed-Solomon code is the $n \times k$ Vandermonde matrix

$$G = \begin{pmatrix} 1 & \alpha_1 & \alpha_1^2 & \cdots & \alpha_1^{k-1} \\ 1 & \alpha_2 & \alpha_2^2 & \cdots & \alpha_2^{k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha_n & \alpha_n^2 & \cdots & \alpha_n^{k-1} \end{pmatrix}.$$

The matrix G is a generator matrix for $\mathrm{RS}_{\mathbb{F},S}[n,k]$, so we immediately obtain that Reed-Solomon codes are linear codes over $\mathbb{F}$. We now consider the distance and rate parameters of the Reed-Solomon code.

Theorem 3.4. *The Reed-Solomon code $RS_{\mathbb{F},S}[n,k]$ has distance $n-k+1$.*

Proof. It suffices to show that every (non-zero) codeword has weight at least $n-k+1$. Note that we can express a codeword as the polynomial $p(X) = m_0 + m_1 X + \cdots + m_{k-1}X^{k-1}$ evaluated on S. Because a polynomial of degree less than $k-1$ on a finite field has at most $k-1$ roots, it follows that each code-word has has at most $k-1$ zero elements and thus weight at least $n-k+1$. Because the rate of the code is k/n, Corollary 3.2 implies that $d \leqslant n-k+1$, and thus equal to $n-k+1$. □

3.2 The Gilbert-Varshamov Bound

Another important question pertaining to the rate–distance trade-off is: given a minimum distance, how large can the rate be? In other words, we want to understand the best achievable trade-off between rate and distance. This is characterized by the Gilbert-Varshamov bound. Much of our discussion of the GV bound and the results presented in this sections are based on that of [Gur10].

Theorem 3.5 (Discrete Gilbert-Varshamov Bound)**.** *Let Σ be an alphabet of size q. Then for any integers n and $d \leqslant n$, there exists a code $\mathcal{C} \subseteq \Sigma^n$ with minimum distance at least d and*

$$|\mathcal{C}| \geqslant \frac{q^n}{\sum_{j=0}^{d-1} \binom{n}{j}(q-1)^j}.$$

Proof. Start with $\mathcal{C} = \emptyset$ and greedily add words from Σ^n that are at distance at least d from all current members of $\mathcal{C}$. Each new word eliminates at most $\sum_{j=0}^{d-1} \binom{n}{j}(q-1)^j$ other candidates (the number of words within distance less than d). Since there are q^n total words, we have $|\mathcal{C}| \geqslant \frac{q^n}{\sum_{j=0}^{d-1} \binom{n}{j}(q-1)^j}$ as desired. □

The more commonly seen version of the Gilbert-Varshamov bound is its asymptotic form, which is obtained by applying Stirling's approximation and taking the limit as $n \to \infty$.

Corollary 3.6 (Asymptotic Gilbert-Varshamov Bound)**.** *For every $\delta \in [0, 1 - 1/q]$, there exists a family of q-ary codes with relative distance at least δ and rate satisfying*

$$R \geqslant 1 - h_q(\delta) - o(1),$$

where $h_q(\delta)$ is the q-ary entropy function given by

$$h_q(\delta) = \delta \log_q(q-1) - \delta \log_q \delta - (1-\delta)\log_q(1-\delta).$$

The Gilbert-Varshamov bound is not tight for all q, in particular, it is known that for large q, there exist codes that are strictly beat the GV bound. Interestingly, a random linear code can meet the GV bound.

Lemma 3.7. *Fix any $\delta \in [0, 1 - 1/q)$ and $0 < \epsilon < 1 - h_q(\delta)$. Let $k = \lceil (1 - h_q(\delta) - \epsilon)n \rceil$ and sample a random matrix $G \in \mathbb{F}_q^{n \times k}$ uniformly (q prime). With high probability, the code $\mathcal{C} = im(G)$ has rate at least $1 - h_q(\delta) - \varepsilon$ and relative distance at least δ.*

Proof. Note that at most q^{i-1} vectors lie in the span of $i-1$ vectors, so the probability that the i-th column lies in the span of the previous $i-1$ is at most q^{i-1}/q^n. Union bounding over k columns yields that the probability that G is not full rank is at most $\sum_{i=1}^{k} \frac{q^{i-1}}{q^n} \leqslant \frac{k}{q^{n-k}} \leqslant e^{-\Omega(n)}$. Note that G being full rank w.h.p. implies that the rate is at least $(1 - h_q(\delta) - \varepsilon)$ w.h.p. as desired. Now observe that for any $x \neq 0 \in \mathbb{F}_q^k$, Gx is uniformly random vector from $\mathbb{F}_q^n$. So, the probability that $\mathrm{wt}(Gx) \leqslant \delta n$ is at most $\frac{\sum_{j=0}^{\delta n} \binom{n}{j} q^j}{q^n} \leqslant q^{(h_q(\delta)-1)n}$

Union bounding over all $x \neq 0 \in \mathbb{F}_q^k$ yields that the probability that there exists a codeword of weight $\leqslant \delta n$ is at most

$$q^k \cdot q^{(h_q(\delta)-1)n} = q^{(1-h_q(\delta)-\varepsilon)n} \cdot q^{(h_q(\delta)-1)n} = q^{-\varepsilon n} \leqslant e^{-\Omega(n)}.$$

□

We remark that there are no known explicit (poly-time constructible) binary codes that meet the GV bound. This difficulty is because verifying Hamming distance of a code is known to be NP-hard (so simply sampling a random linear code fails). To our knowledge, the binary code which comes closest to GV bound parameters is Ta-Shma's code. It is conjectured that no binary code can asymptotically beat the GV bound (GV is tight for $q = 2$), but this remains unproven.

3.3 Reed-Muller Codes

Reed-Muller Codes are a natural generalization of Reed-Solomon Codes. In Reed Solomon Codes, we defined codes over univariate polynomials. Reed-Muller codes are defined over multivariate polynomials.

Definition 3.8 (Reed-Muller code)**.** Let q be a prime power, and let $r, n \in \mathbb{N}$ with $0 \leqslant r < q$. The Reed–Muller code of degree r in n variables over $\mathbb{F}_q$ is defined as

$$\mathrm{RM}_q(r,n) = \left\{ \langle f(\alpha) \rangle_{\alpha \in \mathbb{F}_q^n} \in \mathbb{F}_q^{q^n} \;\middle|\; f : \mathbb{F}_q^n \to \mathbb{F}_q \text{ is a polynomial of total degree } \leqslant r \right\}$$

where $\langle f(\alpha) \rangle_{\alpha \in \mathbb{F}_q^n}$ denotes the vector of evaluations of f over all points $\alpha \in \mathbb{F}_q^n$ in lexicographic order.

The block length of the $\mathrm{RM}_q[r,n]$ code is q^n, and the size of the code is the number of polynomials in $\mathbb{F}_q[X_1, \ldots, X_n]$ of degree at most r, which is exactly the number of integer tuples $(i_1, \ldots, i_n)$ satisfying $0 \leqslant i_j \leqslant q-1, \quad \sum_{j=1}^{m} i_j \leqslant r$. We now proceed to analyzing the distance of the Reed-Muller code. Recall that the crux of the proof for distance was the fact that univariate polynomial of degree d have at most d roots. We can extend this fact to multivariate polynomials via the Schwartz-Zippel Lemma, which is stated below.

Lemma 3.9 (Schwartz–Zippel)**.** *Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be a nonzero multivariate polynomial of total degree d, and let $S \subseteq \mathbb{F}$ be a finite subset. Then,*

$$\mathbf{Pr}_{\alpha_1, \ldots, \alpha_n \in S} [f(\alpha_1, \ldots, \alpha_n) = 0] \leqslant \frac{d}{|S|},$$

where each α_i is chosen independently and uniformly at random from S.

Proof. The proof is by induction on the number of variables. Observe that the base case is simply the univariate case. Because a nonzero univariate polynomial of degree at most d has at most d distinct roots in $\mathbb{F}$, when $\alpha \in S$ is chosen uniformly at random, we have $\mathbf{Pr}[f(\alpha) = 0] \leqslant \frac{d}{|S|}$.

Now, we proceed to the inductive step. Let $p(x)$ be a polynomial of degree d, and divide it by x_n^k, that is, the monomial with the highest degree in x_n. It follows that

$$f(x_1, \ldots, x_n) = x_n^k \cdot q(x_1, \ldots, x_{n-1}) + r(x_1, \ldots, x_n),$$

Observe that it is equivalent to sample $(x_1, \cdots, x_{n-1})$ first, and then x_n. So, by the inductive hypothesis, we have that $\mathbf{Pr}[q(x_1, \ldots, x_{n-1}) = 0] \leqslant \frac{d-k}{|S|}$. Additionally, it follows that if $q \neq 0$, then $f(x_1, \ldots, x_n)$ becomes a nonzero univariate polynomial in x_n of degree k, so it follows that $\mathbf{Pr}_{x_n \sim S}[f = 0 \mid q \neq 0] \leqslant \frac{k}{|S|}$. Therefore, we can finish the proof as follows:

$$\mathbf{Pr}[f = 0] = \mathbf{Pr}[f = 0 \mid q = 0] \cdot \mathbf{Pr}[q = 0] + \mathbf{Pr}[f = 0 \mid q \neq 0] \cdot \mathbf{Pr}[q \neq 0]$$

$$\leqslant \mathbf{Pr}[q = 0] + \mathbf{Pr}[f = 0 \mid q \neq 0]$$
$$\leqslant \frac{d-k}{|S|} + \frac{k}{|S|} = \frac{d}{|S|}.$$

□

We can now apply Schwartz-Zippel Lemma to compute the distance of Reed-Muller codes. Note that this proof almost exactly follows that of Theorem 3.4.

Theorem 3.10. *Let $RM_q(r, n)$ be the Reed–Muller code of degree r in n variables over $\mathbb{F}_q$, with $0 \leqslant r < q$. Then the distance of $RM_q(r, n)$ is at least $q^n - rq^{n-1}$.*

Proof. It suffices to show that each code-word has weight at least $q^n - rq^{n-1}$. By definition, each codeword corresponds to a polynomial $f : \mathbb{F}_q^n \to \mathbb{F}_q$ of total degree at most r, evaluated at all points $\alpha \in \mathbb{F}_q^n$.

By the Schwartz–Zippel Lemma, a nonzero polynomial of total degree at most r vanishes on at most a $\frac{r}{q}$ fraction of $\mathbb{F}_q^n$. Thus, the number of zero evaluations is at most $\frac{r}{q} \cdot q^n = rq^{n-1}$.

□

3.4 Tensor Codes

Thus far, all codes that we have seen are "base codes," that is, we can explicitly describe them in terms of their encoding map (generator matrix). A powerful technique in coding theory is building a stronger, new code upon a base code. The first code of this kind that we will consider is the tensor code, which combines two codes C_1 and C_2 to obtain a new code $C_1 \otimes C_2$.

Definition 3.11 (Tensor Product of Codes)**.** Let $C_1 \subseteq \Sigma^{n_1}$ and $C_2 \subseteq \Sigma^{n_2}$ be linear codes. The *tensor product* of C_1 and C_2, denoted $C_1 \otimes C_2$, is defined as the code

$$C_1 \otimes C_2 = \left\{ A \in \Sigma^{n_1 \times n_2} \;\middle|\; \begin{array}{l} \text{each column of } A \text{ lies in } C_1, \\ \text{each row of } A \text{ lies in } C_2 \end{array} \right\}.$$

We can also view tensor codes from another perspective. That is, the tensor code can be obtained by starting with a message matrix X, encoding all the rows, and then encoding all the columns. It is completely equivalent to encode the columns first, and then the rows.

Lemma 3.12. *Let $C_1 \subseteq \Sigma^{n_1}$ and $C_2 \subseteq \Sigma^{n_2}$ be linear codes over a finite field Σ, with generator matrices $G_1 \in \Sigma^{n_1 \times k_1}$ and $G_2 \in \Sigma^{n_2 \times k_2}$. Then,*

$$C_1 \otimes C_2 = \left\{ Z \in \Sigma^{n_1 \times n_2} \;\middle|\; Z = G_1 X G_2^\top \text{ for some } X \in \Sigma^{k_1 \times k_2} \right\}.$$

Proof. We first show that any matrix $Z \in \Sigma^{n_1 \times n_2}$ of the form $G_1 X G_2^\top$, for some $X \in \Sigma^{k_1 \times k_2}$, lies in $C_1 \otimes C_2$. Observe that each column of Z is a linear combination of the columns of G_1, so it lies in C_1. Similarly, since each row of Z is a linear combination of the rows of $G_2^\top$, it lies in C_2. Thus, $Z \in C_1 \otimes C_2$.

Conversely, suppose $Z \in C_1 \otimes C_2$. Since the columns lie in the image of G_1, there exists a matrix $Y \in \Sigma^{k_1 \times n_2}$ such that $Z = G_1 Y$. Furthermore, because each row of Z lies in the image of $G_2^\top$, we can express $Y = X G_2^\top$ for some $X \in \Sigma^{k_1 \times k_2}$. Putting these together yields the desired result. □

From the above, it is not too difficult to deduce that the generator matrix for $C_1 \otimes C_2$ is $G_1 \otimes G_2$, which exactly yields that the rate of the code is $\frac{k_1 k_2}{n_1 n_2}$. We now compute the distance.

Theorem 3.13. *The tensor product code $C_1 \otimes C_2$ is a $[n_1 n_2, k_1 k_2, d_1 d_2]_\Sigma$ linear code.*

Proof. The linearity of the code, dimension, and block-lengths follow from the above characterization of the generator matrix. Now, consider the codewords $c_1 \in C_1$ with weight d_1 and $c_2 \in C_2$ with weight d_2. Because $c_1 c_2^\top$ has weight $d_1 d_2$, the distance is upper bounded by $d_1 d_2$.

Now, we show that the distance is lower bounded by $d_1 d_2$. Consider some non-zero code-word C and choose some non-zero element $C_i j$. Because the i^{th} row belongs to C_2, there are d_2 non-zero elements in this row. For each of these elements, consider the column they belong to. These columns must have d_1 non-zero elements (because they are members of C_1). Thus, C has weight at least $d_1 d_2$.

□

3.5 Locality in Codes

In practice, error correcting codes are useful because of their resistance to corruption. So, if we are given w which is close to a codeword $c \in \mathcal{C}$ in Hamming distance, we would like to be able to correct and decode w to recover the original message. Interesting cases are when we can achieve this in sublinear time (so we do not even read the entire bit-string w). This motivates the following definitions.

Definition 3.14 (Locally Decodable Code)**.** A code $\mathcal{C} : \Sigma^k \to \Sigma^n$ is a (q, δ)-LDC if, for every $i \in [k]$, there exists a randomized decoder A_i that satisfies the following:

- (query-efficient) A_i queries at most q coordinates of w.
- (accurate decoding) For every message $x \in \Sigma^k$ and $w \in \Sigma^n$ such that $\Delta(C(x), w) \leqslant \delta n$,

$$\mathbf{Pr}[A_i(w) = x_i] > \frac{1}{2}$$

Definition 3.15 (Locally Correctable Code)**.** A code $\mathcal{C} : \Sigma^k \to \Sigma^n$ is a (q, δ)-LCC if, for every $i \in [n]$, there exists a randomized corrector A_i that satisfies the following:

- (query-efficient) A_i queries at most q coordinates of w.
- (accurate correction) For every codeword $c \in \mathcal{C}$ and every $w \in \Sigma^n$ such that $\Delta(c, w) \leqslant \delta n$,

$$\mathbf{Pr}[A_i(w) = c_i] > \frac{1}{2}$$

There is a very close relationship between LCCs and LDCs. In particular, if a linear code $\mathcal{C} : \Sigma^k \to \Sigma^n$ is LCC, then it is LDC. This can be seen by rewriting the generator matrix of the LCC in systematic form, and then applying the fact that message is exactly the first k bits of its corresponding code-word.

The last local property we discuss is local testability, which is of much interest due to its applications in PCPs. There are many (mostly equivalent) mathematical formulations of locally testable codes. The most basic is the following:

Definition 3.16 (Locally Testable Code)**.** A code $\mathcal{C}$ is called κ- locally testable with q queries if there exists an algorithm T, which queries at most q bits of r, and has the following behavior:

- (completeness) If $r \in \mathcal{C}$, then $\Pr[T \text{ accepts}] = 1$,
- (soundness) If $r \notin \mathcal{C}$, then $\Pr[T \text{ rejects}] \geqslant \kappa \cdot \delta(r, \mathcal{C})$.

where $\delta(r, \mathcal{C})$ refers to the minimum distance between r and a codeword $c \in \mathcal{C}$.

Parsing the above definition, we note that completeness ensures that if r is a valid codeword, then T will always accept it, whereas soundness guarantees that if r is far from being a member of $\mathcal{C}$, then T will reject it with high probability. Good locally testable codes include Reed-Solomon codes, Reed-Muller codes, and Hadamard codes.

We now proceed by showing the proofs for locality of Hadamard codes, which were formally defined in Section 2.4. These codes have excellent locality, and are in fact a 2-query LDC, 2-query LCC, and 3-query LTC.

Lemma 3.17. *For every $\delta \in (0, 1/4)$, the Hadamard code is a $(2, \delta)$-locally correctable code. The Hadamard code is also a $(2, \delta)$-locally decodable LDC.*

Proof. We are given a corrupted string r that is at most hamming distance δn away from the correct code-word c. Suppose we want to recover the i^{th} bit, we achieve this as follows:

1. Uniformly sample an integer $k \in \{0, 1, \cdots, 2^m - 1\}$
2. Query the positions k and $k + i$ in w.
3. Output the bit given by $r_k + r_{k+i} \in \mathbb{F}_2$.

Observe that the queried positions k and $k + i$ are uniformly random and independent over $\{0, 1, \cdots, 2^m - 1\}$, so each lands in the corrupted set with probability at most δ. So, by applying union bound, the probability that either of the queried positions is corrupted is at most 2δ. Thus, with probability at least $1 - 2\delta$, both queries are correct, and we successfully recovered the i^{th} bit.

Note that the original bits from the message are members of the corresponding code-word for Hadamard codes. So, the code is locally decodable following the exact same proof.

□

We now provide the tester T for the Hadamard Code and show that the Hadamard Code is locally testable.

1. Independently and uniformly sample 2 integers k_1, k_2 from $\{0, 1, 2, \cdots, 2^m - 1\}$. Denote $k_3 = k_1 + k_2 \mod 2^m$.
2. Query the bits r_{k_1}, r_{k_2}, and r_{k_3}. Accept if $r_{k_1} + r_{k_2} = r_{k_3}$, reject otherwise

Lemma 3.18. *Let $r \in \mathbb{F}_2^{2^m}$ be a bit-string and $c = f(x) \in \mathcal{C}$ to be the closest code-word to r and $\delta = \delta(r, c) = \delta(r, \mathcal{C})$. If $0 < \delta \leqslant 5/16$, then the Hadamard code is $(3\delta - 6\delta^2)$- locally testable with 3 queries.*

Proof. We first show completeness. If $r \in \mathcal{C}$, then there exists x such that $r_k = \langle x, y^{(k)} \rangle \ \forall k$. Then, we have that $r_{k_1} + r_{k_2} = \langle x, y^{(k_1)} \rangle + \langle x, y^{(k_2)} \rangle = \langle x, y^{(k_1)} + y^{(k_2)} \rangle = \langle x, y^{(k_3)} \rangle = r_{k_3}$. Now, we show soundness:

Denote $S = \{k : r_k \neq \langle x, y^{(k)} \rangle\}$, by definition, $|S| = \delta \cdot 2^m$. Additionally, note that if k_1 are k_3 are uniformly and independently sampled from residues $\mod 2^m$, then $k_3 = k_1 + k_2$ is uniformly sampled from residues $\mod 2^m$. Thus, the probability that $\Pr[k_1 \in S] = \delta$, $\Pr]k_2 \in S] = \delta$, and $\Pr[k_3 \in S] = \delta$. Note that T will reject r if exactly one of k_1, k_2, k_3 is in S. We thus compute as follows:

$$\begin{aligned}\Pr[\text{reject } r] &\geqslant \Pr[k_1 \in S \vee k_2 \in S \vee k_3 \in S] - \sum_{i \neq j} \Pr[k_i \in S \wedge k_j \in S] \\ &\geqslant \sum_i \Pr[k_i \in S] - 2 \sum_{i \neq j} \Pr[k_i \in S \wedge k_j \in S] \geqslant 3\delta - 6\delta^2\end{aligned}$$

□

I would like to thank Devan Shah for his emotional support and his work on Theorem 7.1.

4 Expander Graphs and Applications to Coding Theory

4.1 Expander Graphs as Pseudo-Random Objects

An expander graph is a graph that is "random-like," namely, it is an object with an explicit description that shares the properties of the average graph (from a known model of random graphs such as Erdos-Reyni random graphs or the configuration model). There are multiple definitions of expanders: edge expansion, vertex expansion, unique neighbor expanders, spectral expanders etc. As we will see throughout this section, these various notions of expansion share a deep relationship.

Definition 4.1 (Spectral Expander)**.** Let $G = (V, E)$ be a d-regular undirected graph with n vertice-sand denote A_G to be its *adjacency matrix* and we write its (real) eigenvalues $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_n$. We say that G is a λ-*spectral expander* if $\lambda(G) := \max\{|\lambda_2|, |\lambda_n|\} \leqslant \lambda$.

Note that it is easy to see that $\lambda_1 = d$ and $\lambda_i < d$ for all $i \neq 1$ if G is connected. The above definition of expansion is known as *two-sided expansion*, as it controls both the second largest and smallest eigenvalues. It is also common to work with a weaker notion of *one-sided expansion*, defined solely in terms of λ_2.

We will often work with a normalized version of the above, so we will write $\widetilde{A_G} = \frac{1}{d}A_G$ to denote the *normalized adjacency matrix* of G, whose eigenvalues are the same as the ones of A_G but scaled by $\frac{1}{d}$. This normalized adjacency matrix is exactly the Markov operator for the random walk on G. Thus, for G a $\lambda-$spectral expander, λ characterizes the rate of convergence of a random walk in G to the uniform distribution (its stationary distribution, assuming G is connected and not bipartite). We will occasionally use the language of random walk operators when referring to the normalized adjacency matrix of graphs, whenever it is convenient to think about $\widetilde{A_G}$ as a Markov operator.

A "good expander" is a graph for which $\lambda(G)$ is bounded away from d. The well-known Alon-Boppana [Alo86, Nil91, Fri93] bound asserts that $2\sqrt{d-1}$ is essentially a lower bound to expansion, and so an optimal spectral expander, also known as a *Ramanujan graph*, is a graph G for which $\lambda(G) \leqslant 2\sqrt{d-1}$. Perhaps rather surprisingly, Friedman's Theorem [Fri08, Bor20] shows that (almost) Ramanujan graphs are very common: for any $d \leqslant 3$, if G is a uniformly random d-regular graph, then $\lambda(G) \leqslant 2\sqrt{d-1} + o(1)$ with high probability. Many explicit constructions are also known (see [MOP22] for a comprehensive list of recent constructions). Some of these are algebraic, such as the results of Lubotzky-Phillips-Sarnak [LPS88], Margulis [Mar88] and Morgenstern [Mor94], who showed that when $d-1$ is a prime power there exist explicit constructions of d-regular Ramanujan graphs. Some are combinatorial, such as the derandomization of [Bor20] by Mohanty-O'Donnell-Paredes [MOP22] and Alon [Alo21].

We now discuss the notion of combinatorial expansion. To do this, we first define the notion of conductance and volume for a graph G.

Definition 4.2 (Conductance)**.** Let $G = (V, E)$ be an undirected graph. For any subset $S \subseteq V$ with $0 < |S| \leqslant |V|/2$, we define the edge boundary of S to be $E(S, V \setminus S) = \{(u, v) \in E : u \in S, v \in V \setminus S\}$ and the volume of S to be $\mathrm{vol}(S) = \sum_{v \in S} \deg(v)$. The *conductance* (or edge expansion) of S is defined as:

$$\phi(S) = \frac{|E(S, V \setminus S)|}{\min(\mathrm{vol}(S), \mathrm{vol}(V \setminus S))}.$$

Intuitively, the conductance is a measure of the isoperimetry of a graph, which directly measures the expansion of a graph. Note that this also corresponds to fast mixing time.

Definition 4.3. A graph G is a *ϕ-expander* if for all $S \subseteq V$ with $0 < |S| \leqslant n/2$, we have $\phi(S) \geqslant \phi$.

Returning to the analogy of expanders being random-like, we show that this conductance based definition of expanders is satisfied by most Erdos-Reyni random graphs above the connectivity threshold of $p = \frac{\log n}{n}$.

Lemma 4.4. *Let $G \sim G(n,p)$ be an Erdos-Reyni Graph with $p = C\frac{\log n}{n}$ for a sufficiently large constant $C > 0$. Then with high probability, G is an $1/8$-expander.*

Proof. Consider a fixed subset $S \subseteq V$, $|S| \leqslant n/2$. Note that $\mathbb{E}\,|E(S, V \setminus S)| = p|S| \cdot (n - |S|)$. Furthermore, we have that $\mathbb{E}[\mathrm{vol}(S)] = |S|(n-1)p$. Because both these expectations were obtained by summing across Bernoulli random variables, we apply the Chernoff Bound to get:

$$\mathbf{Pr}\left[E(S, V \setminus S) \leqslant \frac{1}{2}p|S| \cdot (n - |S|)\right] \leqslant \exp\left(-\Omega(|S|np)\right)$$

$$\mathbf{Pr}\left[\mathrm{vol}(S) \geqslant 2|S|(n-1)p\right] \leqslant \exp\left(-\Omega(|S|np)\right)$$

Union bounding yields that with probability at least $1 - 2\exp(-\Omega(snp))$, we have $\phi(S) = \frac{|E(S,V\setminus S)|}{\mathrm{vol}(S)} \geqslant \frac{(1/4)snp}{2snp} = \frac{1}{8}$. Applying a Union bound over all sets S implies that the probability that there exists a set with conductance worse than $1/8$ is upper bounded

$$\sum_{i=1}^{n/2} \binom{n}{i} \cdot \exp(-\Omega(inp)) = \sum_{i=1}^{n/2} \exp\left(i \log n - \Omega(inp)\right).$$

which tends to 0 for C, n sufficiently large. So, with high probability, all sets S of size at most $n/2$ satisfy $\phi(S) \geqslant \frac{1}{8}$, yielding the desired result. □

Recall that we will primarily deal with regular graphs, so the volume of a given set S is exactly $d|S|$. To simplify matters (and allow a slightly more parametrized definition of combinatorial expansion), we will use the following definition of an edge expander.

Definition 4.5. A d-regular graph G is a *(K, ϵ)-edge expander* if for all subsets S of size at most K, $E(S, V \setminus S) \geqslant \epsilon|S|d$.

One reason to utilize this regular graph specific definition of edge-expansion, is that it allows us to elegantly describe the connection between spectral and combinatorial expansion via the Expander Mixing Lemma, which is stated below.

Lemma 4.6 (Expander Mixing Lemma)**.** *Let $G = (V, E)$ be a d-regular graph on n vertices with adjacency matrix $A \in \mathbb{R}^{n\times n}$. Suppose all eigenvalues of A satisfy $\lambda_1 = d \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_n$, and define $\lambda := \max\{|\lambda_2|, |\lambda_n|\}$. Then for any subsets $S, T \subseteq V$, we have*

$$\left|e(S,T) - \frac{d}{n}|S||T|\right| \leqslant \lambda\sqrt{|S||T|},$$

where $e(S,T)$ denotes the number of edges between S and T.

Proof. Let $\mathbf{1}_S, \mathbf{1}_T \in \mathbb{R}^n$ denote the indicator vectors of the sets S and T. Then we can express the number of edges between S and T as

$$e(S,T) = \mathbf{1}_S^\top A \mathbf{1}_T.$$

Define the all-ones vector $\mathbf{u} = \frac{1}{\sqrt{n}}\mathbf{1}$, which is the eigenvector of A corresponding to the eigenvalue d. Let $\{v_1, v_2, \ldots, v_n\}$ be an orthonormal eigenbasis of $\mathbb{R}^n$ with $v_1 = \mathbf{u}$, and $Av_i = \lambda_i v_i$ for each i. We now write $\mathbf{1}_S$ and $\mathbf{1}_T$ in this basis to attain

$$e(S,T) = \left(\sum_{i=1}^n \alpha_i v_i\right)^\top A \left(\sum_{j=1}^n \beta_j v_j\right) = \sum_{i=1}^n \alpha_i \beta_i \lambda_i,$$

Now, observe that $\alpha_1 = \langle \mathbf{1}_S, \mathbf{u}\rangle = \left\langle \mathbf{1}_S, \frac{1}{\sqrt{n}}\mathbf{1}\right\rangle = \frac{|S|}{\sqrt{n}}$, and $\beta_1 = \frac{|T|}{\sqrt{n}}$ which implies that

$$e(S,T) = \frac{d}{n}|S||T| + \sum_{i=2}^n \alpha_i \beta_i \lambda_i.$$

We now bound the remainder term using the triangle inequality and the fact that $|\lambda_i| \leqslant \lambda$ for all $i \geqslant 2$:

$$\left| e(S,T) - \frac{d}{n}|S||T| \right| = \left| \sum_{i=2}^n \alpha_i \beta_i \lambda_i \right| \leqslant \lambda \sum_{i=2}^n |\alpha_i \beta_i|.$$

Applying Cauchy–Schwarz,

$$\sum_{i=2}^n |\alpha_i \beta_i| \leqslant \left(\sum_{i=2}^n \alpha_i^2\right)^{1/2} \left(\sum_{i=2}^n \beta_i^2\right)^{1/2}.$$

Note that $\sum_{i=1}^n \alpha_i^2 = \|\mathbf{1}_S\|^2 = |S|$, and $\alpha_1^2 = \left(\frac{|S|}{\sqrt{n}}\right)^2 = \frac{|S|^2}{n}$, so

$$\sum_{i=2}^n \alpha_i^2 = |S| - \frac{|S|^2}{n}, \quad \sum_{i=2}^n \beta_i^2 = |T| - \frac{|T|^2}{n}.$$

Therefore,

$$\left| e(S,T) - \frac{d}{n}|S||T| \right| \leqslant \lambda \sqrt{\left(|S| - \frac{|S|^2}{n}\right)\left(|T| - \frac{|T|^2}{n}\right)} \leqslant \lambda \sqrt{|S||T|},$$

□

Another fundamental result connecting combinatorial and spectral expansion is Cheeger's inequality, which is stated below. We omit its proof.

Lemma 4.7 (Cheeger's Inequality)**.** *Let $G = (V, E)$ be a connected, undirected graph with normalized Laplacian $\mathcal{L}$. Denote ν_2 to be the second smallest eigenvalue of $\mathcal{L}$, and $\phi(G)$ denote the Cheeger constant of G, defined as $\phi(G) := \min_{\emptyset \neq S \subsetneq V} \phi(S)$ where $\phi(S)$ denotes the conductance of S as defined in Definition 4.2. Then,*

$$\frac{\nu_2}{2} \leqslant \phi(G) \leqslant \sqrt{2\nu_2}.$$

We also comment on another combinatorial notion of expansion, called vertex expansion. It is defined very similarly to edge expansion.

Definition 4.8 (Vertex Expander)**.** Let $G = (V, E)$ be a d-regular undirected graph. We say that G is an (α, ϵ)-*vertex expander* if for every subset $S \subseteq V$ with $|S| \leqslant \alpha|V|$, the neighborhood of S satisfies $|N(S)| \geqslant (1-\epsilon)d \cdot |S|$

It is fairly easy to see that a random graph (with parameter p above the connectivity threshold of $C\frac{\log n}{n}$) will have this property, and the proof almost exactly follows that of Lemma 4.4. This notion of vertex-expansion is sometimes quite convenient, as we will see in our analysis of Tanner codes.

Lastly, we note that all definitions of expansion we have provided can be easily generalized to bipartite-expanders. As an example, a bipartite vertex-expander can be defined as follows:

Definition 4.9 (Bipartite Vertex Expander)**.** Let $G = (L \cup R, E)$ be a (d, D)- regular bipartite graph. We say that G is an (α, ϵ)-bipartite vertex expander if for every subset $S \subseteq L$ with $|S| \leqslant \alpha|L|$, its neighborhood $N(S) \subseteq R$ satisfies $|N(S)| \geqslant (1-\varepsilon)d|S|$.

4.2 Expander Codes

Let $G = (L \cup R, E)$ be a bipartite graph, where $|L| = n$, $|R| = m$, and the edge set $E \subseteq L \times R$. We define the *biadjacency matrix* $H \in \mathbb{F}_2^{n \times m}$ associated with G as:

$$H_{i,j} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases} \quad \text{for } i \in L,\ j \in R.$$

Definition 4.10. Let G be a bipartite graph, and H its biadjacency matrix. Using H, we define the $\mathcal{C}(G) := \ker H^\top = \{x \in \mathbb{F}_2^n : H^\top x = 0\}$, that is the kernel of $H^\top$ over $\mathbb{F}_2$. In the case that G is a bipartite expander graph, we say that $\mathcal{C}(G)$ is an *expander code*.

Informally, we can interpret each node $i \in L$ as a coordinate of a codeword x, and each node $j \in R$ as a parity-check constraint. The code consists of all vectors $x \in \mathbb{F}_2^n$ such that the XOR (mod-2 sum) of the coordinates in x corresponding to neighbors of each $j \in R$ is zero.

Clearly, the code is linear. Because H has dimension at most m, and $\mathcal{C}$ is its kernel, it follows that $\mathcal{C}$ has dimension at least $n - m$, and thus rate $1 - m/n$.

Moving to the analysis of expander code's distance, it turns out that with some expanding properties, we can reach a distance $\Omega(n)$, making expander codes good codes. In order to prove this, we first need to understand the combinatorial properties of bipartite expander graphs.

Lemma 4.11. *For a bipartite graph $G = (L \cup R, E)$ and a given subset $S \subset L$, define its unique neighborhood, $U(S)$ to be:*

$$U(S) = \{v \in R | N(v) \cap S = 1\}.$$

If a (d, D)-regular bipartite graph G is an $(\alpha, D(1-\epsilon))$-expander, then for all $S \subseteq L$ such that $|S| \leqslant \alpha n$

$$U(S) \geqslant D(1-2\epsilon)|S|$$

Proof. Note that since G is D-left-regular, then the number of edges coming out of S is $D|S|$. Because the vertices in $U(S)$ receive only one edge from S, the number of edges between S to $N(S) \setminus U(S)$ is equal to $D|S| - |U(S)|$. Every vertex in $N(S) \setminus U(S)$ has at least two edges coming from S, so $E(S, N(S) \setminus U(S)) \geqslant 2(|N(S)| - |U(S)|) \implies D|S| \geqslant 2|N(S)| - |U(S)|$. Because G is an expander, we know that $|N(S)| \geqslant D(1-\epsilon)|S|$. Thus, $U(S) \geqslant 2D(1-\epsilon)|S| - D|S| \geqslant D(1-2\epsilon)|S|$ as desired. □

Lemma 4.12. *Given an expander code $\mathcal{C}(G)$, where G is a bipartite $(\alpha, D(1-\epsilon))$- expander, $\Delta(\mathcal{C}) \geqslant 2\alpha(1-\epsilon)n$.*

Proof. Assume (for the sake of contradiction) that there exists a codeword $c \in \mathcal{C}$ with Hamming weight $< 2\alpha(1-\epsilon)n$, and denote T to be its support. Let S be some subset of T such that $|S| = \alpha n$. By Lemma 4.16, we have that $|U(S)| \geqslant (D - 2\epsilon)|S|$.

Note that the number of edges stemming from $T \setminus S$ is $D|T \setminus S|$, which means $D|T \setminus S| = D(|T| - |S|) < D(2\alpha(1-\epsilon)n - |S|) = D(1-2\epsilon)|S|$. Thus, $D|T \setminus S| < |U(S)|$.

This implies that there exists at least one unique neighbor of S that is not connected to any vertex in $T \setminus S$. Since $S \subseteq T$, this neighbor is connected to exactly one vertex in the support of c. Therefore, the corresponding parity check sums to 1 and fails, contradicting c being a valid codeword.

□

4.3 Tanner Codes

Definition 4.13. Given a (d, D)-regular bipartite graph G, where $|L| = n$ and $|R| = m$, and a linear code $C_0 \subseteq \mathbb{F}_2^D$, we define a *Tanner code* $T(G, C_0)$ of G and C_0 as

$$T(G, C_0) = \left\{ x \in \mathbb{F}_2^n \;\middle|\; \forall v \in R,\ x|_{N(v)} \in C_0 \right\},$$

where $x|_{N(v)}$ denotes the restriction of x to the neighborhood $N(v)$ of v in G.

Call the local code C_0 the parity check code. Note that if C_0 is linear, then $T(G, C_0)$ is also linear. This follows from the fact that for any $x_1, x_2 \in \mathbb{F}_2^n$, we have:

$$\left(x_1|_{N(v)}\right) + \left(x_2|_{N(v)}\right) = (x_1 + x_2)\,|_{N(v)} \quad \text{for all } v \in R.$$

So if $x_1, x_2 \in T(G, C_0)$, then for each $v \in R$, $(x_1 + x_2)\,|_{N(v)} \in C_0$, and hence $x_1 + x_2 \in T(G, C_0)$.

Additionally, Definition 4.13 gives a clean interpretation of Tanner codes as global codes defined by local constraints. Each right vertex $v \in R$ enforces that the bits assigned to its neighborhood $N(v) \subseteq L$ must form a valid codeword in the smaller code C_0. When the underlying graph is a good expander, the local constraints propagate well, forming a good global code. We now compute the relevant parameters (distance and rate) for Tanner codes.

Lemma 4.14. *Let $C_0 \subseteq \mathbb{F}_2^D$ be a linear code, and let G be a (d, D)-regular bipartite graph with $|L| = n$, $|R| = m$. Then,*

$$\dim(T(G, C_0)) \geqslant n - m(D - \dim(C_0)).$$

Proof. For each right vertex $v \in R$, the constraint $x|_{N(v)} \in C_0$ imposes $D - \dim(C_0)$ independent linear conditions on $x \in \mathbb{F}_2^n$. Since there are m such vertices, the total number of independent constraints is at most $m(D - \dim(C_0))$. Therefore, the dimension of the space of solutions is at least $n - m(D - \dim(C_0))$, as desired. □

Recall that the dimension of a linear code is exactly equal to its rate, thus all that remains is to characterize the distance of $T(G, C_0)$.

Lemma 4.15. *Let $C_0 \subseteq \mathbb{F}_2^D$ be a linear code with relative distance $\delta_0 = \frac{\Delta_0}{D}$, and let G be a (d, D)-regular (α, ϵ) bipartite vertex expanding graph with $|L| = n$ and $|R| = m$. If $(1 - \epsilon)\Delta_0 > 1$, then, the resulting Tanner code $T(G, C_0)$ has distance $\Delta(T(G, C_0)) \geqslant \alpha n$*

Proof. Assume (for the sake of contradiction) that there exists a non-zero codeword $c \in T(G, C_0) \setminus \{0\}$ with weight less than αn and $S = \text{supp}(c) \subseteq L$ its support. Denote $T = N(S) \subseteq R$ to be the set of right vertices adjacent to S.

Because $|S| < \alpha|L|$, by bipartite vertex expansion, we have $|T| \geqslant (1 - \varepsilon)d \cdot |S|$. Furthermore, we also must have that each $v \in T$ satisfies $c|_{N(v)} \in C_0 \setminus \{0\} \implies \text{wt}(c|_{N(v)}) \geqslant \Delta_0$. Summing over T yields:

$$\sum_{v \in T} \text{wt}(x|_{N(v)}) \geqslant \Delta_0 \cdot |T| \geqslant \Delta_0 \cdot (1 - \varepsilon)d \cdot |S|.$$

Observe that each bit can appear in at most d local views, so we have that $d|S| > \sum_{v \in T} \text{wt}(x|_{N(v)}) \implies 1 \geqslant \Delta_0(1 - \epsilon)$, a contradiction.

□

4.4 Expander Codes are VLTCs

The Expander and Tanner codes that we have discussed have good rate and distance for almost any expander. However, the same is not true for locality. That is, if we were to choose a random expander graph, then we can 'fool' the natural locality tester for expander graphs.

Lemma 4.16. *For a bipartite graph $G = (L \cup R, E)$ and a given subset $S \subset L$, define its unique neighborhood, $U(S)$ to be:*

$$U(S) = \{v \in R | N(v) \cap S = 1\}.$$

If a (d, D)-regular bipartite graph G is an $(\alpha, D(1 - \epsilon))$-expander, then for all $S \subseteq L$ such that $|S| \leqslant \alpha n$

$$U(S) \geqslant D(1 - 2\epsilon)|S|$$

Definition 4.17 (Natural Tester)**.** Given a (d, D) regular bipartite $(\alpha, D(1 - \epsilon))$ expander G, consider its expander code C_G. A natural local tester would:

- Pick $O(1)$ parity checks uniformly at random,
- Accept if all of the parity checks are satisfied.

We will (informally) show that there exist expander graphs G for which the above natural tester fails. We sketch an argument to show that there is some $x \notin C(G)$ that is close to $C(G)$, for which the tester struggles to detect errors ($\mathbf{Pr}[T(x) \neq 1] = O(1/n)$).

Recall Lemma 4.16, which stats that if G is a $(\alpha, D(1-2\epsilon))$ unique-neighbor expander (every set S of size at most αn has $U(S) \geqslant D(1-2\epsilon)$). Now, consider a random (bipartite) graph G, which we know to be an expander.

The idea is to randomly remove some parity check (right node) from G to form the graph G'. We now choose a code-word $x \in C(G') \setminus C(G)$ that is not a member of $C(G)$. As this word only violates a single parity check of $C(G)$, the probability of a natural tester for G detecting it is low.

In some sense, the arbitrary error code failed because its errors do not propagate enough. If we desire a code that is locally testable, we require any error to propagate and affect multiple parity checks. This means that there do exist expander graphs that are LTC (and we will describe one in Section 5.3). Indeed, one key ingredient in the construction that we describe there (from [DEL+22]) is the use of high-dimensional expansion, which intermixes the parity checks.

While arbitrary expander codes are not locally testable, they do satisfy a relaxation of local testability, called vicinity locally testable codes (VLTCs), that is defined below. We use the definition given by [CY24].

Definition 4.18 (VLTC)**.** A code $C \subseteq \mathbb{F}^N$ is called a $(q, \delta, \kappa, \sigma)$-*VLTC* (vicinity locally testable code) if there exists a randomized procedure (which we denote Tes) that takes as input a binary string of length N. It queries at most q positions and outputs either $\{\circ, \perp\}$. It also satisfies the following:

- (relaxed soundness) For every $c \in C$ and $w \in \mathbb{F}^N$ with $\text{RelDist}(w, c) \leqslant \delta$,

$$\mathbf{Pr}[\text{Tes}(w) = \perp] \geqslant \kappa \cdot \text{RelDist}(w, c) - \sigma.$$

- (completeness) $\text{Tes}(c) = \circ$ with probability 1 for all $c \in C$.

Lemma 4.19 ([CY24])**.** *Let $G = (L, R, E)$ be a d-left-regular (γ, α)-unique-neighbor expander with average right-degree $\bar{c}$. Then, for every $b > 1$, $\text{EC}(G)$ is a*

$$(b\bar{c}, \gamma, \alpha\bar{c}, \tfrac{1}{b})\text{-VLTC}.$$

Proof. Let $R' = \{v \in R \mid \deg(v) \leqslant b\bar{c}\}$. Since the average degree over R is $\bar{c}$, Markov's inequality implies that $|R'| \geqslant (1 - \frac{1}{b})|R|$.

Now, consider the following tester for $\text{EC}(G)$ with access to $w \in \mathbb{F}^L$. It samples $v \in R'$ uniformly at random, queries w on $\Gamma(v)$, and accepts if $\sum_{u \in \Gamma(v)} w_u = 0$; otherwise, it rejects. The query complexity is at most $b\bar{c}$, and any codeword in $\text{EC}(G)$ trivially passes the test.

Now suppose w differs from some $c \in \text{EC}(G)$ on a set $S \subseteq L$, with $|S| \leqslant \gamma|L|$. The expansion property implies that $|\Gamma_u(S)| \geqslant \alpha d|S|$. Since the tester samples only from R', it detects an error with probability at least

$$\frac{|\Gamma_u(S)| - |R \setminus R'|}{|R|} \geqslant \frac{\alpha d|S|}{|R|} - \frac{1}{b} = \alpha\bar{c} \cdot \text{RelDist}(w, c) - \frac{1}{b}.$$

□

5 High Dimensional Expansion

5.1 Classical High Dimensional Expanders

A high dimensional expander (HDX) is a 'hypergraph generalization' of expander graphs. High dimensional expansion is generally defined on a more structured hypergraph, called a simplicial complex, which we define as follows:

Definition 5.1. A *simplicial complex* X on a vertex set V is a collection of subsets of V, known as faces, satisfying downward-closure, i.e. if $\tau \in X$ and $\sigma \subseteq \tau$ then $\sigma \in X$.

We denote by $X(k)$ the set of *k-dimensional faces*, meaning the set of faces with cardinality $k+1$. In particular, this implies that $X(0)$ is the vertex set V. The *dimension* of a simplicial complex X is the maximum dimension of a face.

The first definitions of expansion in simplicial complex were topological, and based on the seminal work of Lubotzky-Samuels-Vishne [LSV05] that defined and constructed *Ramanujan complexes*. More relevant to us is the spectral notion of high dimensional expanders. To fully discuss these objects, we must provide some definitions.

The 1-*skeleton* of a simplicial complex X is the graph defined by $(X(0), X(1))$, which means it corresponds to the underlying graph of X. Given a d-dimensional simplicial complex X, the *link* of a k-face τ for $k \leqslant d-1$ is a $(d-k-1)$-dimensional simplicial complex defined as $X_\tau := \{\sigma \setminus \tau : \sigma \in X, \tau \subseteq \sigma\}$. Links encapsulate the "local information" of a face. For example, if X is a 1-dimensional (i.e. a graph) then the link of a vertex is its neighborhood.

Definition 5.2. For $-1 \leqslant k \leqslant d-2$, the *k-dimensional local expansion* of a simplicial complex X is defined as

$$\nu^{(k)}(X) = \min_{\sigma \in X(k)} \nu_2\left(X_\sigma(0), X_\sigma(1)\right),$$

The *local expansion* of X is the minimum of the k-dimensional local expansions over all $k \geqslant 0$, whereas the *global expansion* is the expansion of the 1-skeleton.

Recall that $\lambda_2(\cdot)$ refers to the second largest eigenvalue of an operator whereas $\nu_2(\cdot)$ refers to its spectral gap. Note that in simplicial complexes spectral expansion is usually defined in terms of normalized one-sided expansion, as is evident in the above definition. This exact definition was introduced in [DK17] motivated by the study of PCPs and agreement tests. A more direct reason for why we care about expanding simplicial complexes is that their natural random walks will have good expansion. In particular, consider the stochastic process defined as follows:

Definition 5.3. Given a d-dimensional simplicial complex X, for $-1 \leqslant k \leqslant d-1$, we define the *up-step random walk operator* $W_k^\uparrow \in \mathbb{R}^{X(k+1)\times X(k)}$ so that for $\sigma \in X(k)$, $\tau \in X(k+1)$,

$$\widetilde{W}_k^\uparrow \in \mathbb{R}^{X(k+1)\times X(k)}$$

so that for $\sigma \in X(k)$, $\tau \in X(k+1)$,

$$\widetilde{W}_k^\uparrow(\tau,\sigma) = \begin{cases} \frac{1}{\deg^\uparrow(\sigma)} & \text{if } \sigma \subset \tau, \\ 0 & \text{otherwise,} \end{cases}$$

where $\deg^\uparrow(\sigma)$ denotes the number of $(k+1)$-simplices in X that contain σ.

The down-step random walk operator on a d-dimensional simplicial complex X is similarly defined.

Definition 5.4. For $0 \leqslant k \leqslant d$, define the *down-step random walk operator* $W_k^{\downarrow} \in \mathbb{R}^{X(k-1)\times X(k)}$ so that for $\sigma \in X(k)$, $\tau \in X(k-1)$,

$$W_k^{\downarrow}(\tau,\sigma) = \begin{cases} \frac{1}{k+1} & \text{if } \tau \subset \sigma \in X(k), \\ 0 & \text{otherwise.} \end{cases}$$

By putting together the up-step and the down-step operators, we attain the *up-down* and *down-up* random walk operators $W_k^{\uparrow\downarrow} = W_{k+1}^{\downarrow} \circ W_k^{\uparrow}$, $\quad W_k^{\downarrow\uparrow} = W_{k-1}^{\uparrow} \circ W_k^{\downarrow}$. Note that the *up-down* operator is exactly equivalent to a lazy random walk (with lazy parameter $1/2$) on a graph, as the up-step maps a vertex to all edges that contain it, and the down-step maps it to either itself of one of its neighbors with equal probability.

The connection between globally expanding simplicial complexes and the up-down / down-up random walk can be made explicit by the following theorem.

Theorem 5.5 ([AL20])**.** *Let X be an H-dimensional simplicial complex, and let $W_k^{\uparrow\downarrow}$ denote the up-down walk operator on X. Then for every $0 \leqslant k \leqslant H-1$,*

$$\nu_2\left(W_k^{\uparrow\downarrow}\right) \geqslant \frac{1}{k+2}\prod_{j=-1}^{k-1}\nu^{(j)}(X).$$

That is, simplicial complex with good local expansion will have expanding up-down random walks. This local-to-global paradigm drives the utility of classical HDXs. Perhaps the most famous result characterizing this phenomenon is Oppenheim's Trickling Down Theorem, which roughly states that good local expansion implies good global expansion.

Theorem 5.6 (Oppenheim's Trickling Down Theorem)**.** *Let X be a d-dimensional simplicial complex for $d \geqslant 2$ with fully connected 1-skeleton. If, for all $v \in X(0)$, the link X_v is a λ-one-sided spectral expander, then the underlying graph of X is a $\frac{\lambda}{1-\lambda}$-one-sided spectral expander.*

Already, HDXs seem to share the same flavor as LTCs, and indeed, the c^3-LTC hypothesis was resolved by utilizing square complexes (a special class of simplicial complexes) that are the focus of the next section.

5.2 Square Complexes

The discussion of high-dimensional expanders so far has focused on simplicial complexes, but there are other interesting high-dimensional objects, in particular square complexes, which is the focus of this section.

Definition 5.7. A *square complex* Z is a 2-dimensional object defined as 3 collections of subsets of a vertex set V:

- The vertices, denoted by $Z(0)$, are exactly the vertex set V.
- The edges, denoted by $Z(1)$, are sets of two vertices.

- The squares, denoted by $Z(2)$, are ordered sets of four vertices, such that if $(v_1, v_2, v_3, v_4) \in Z(2)$, then $\{v_i, v_{i+1}\} \in Z(1)$ (where we use a slightly abuse of notation so that $v_{4+1} = v_1$).

This definition is very similar to the definition of a 2-dimensional simplicial complex. In a 2-dimension simplicial complex, 2-faces are triangles or 3-cycles, whereas in this definition they are squares or 4-cycles. Also, the property enforced in the definition of squares is analogous to downward closure. There are a few additional interesting things to remark about this definition.

Remark 5.8. Under this notation, we have that $(v_1, v_2, v_3, v_4) = (v_4, v_1, v_2, v_3) = (v_3, v_4, v_1, v_2) = (v_2, v_3, v_4, v_1)$, but $(v_1, v_2, v_3, v_4) \neq (v_2, v_1, v_3, v_4)$.

$Z(2)$ is a set of 4-cycles of Z, but not that not all 4-cycles (in the graph sense) are necessarily in $Z(2)$[1].

Remark 5.9. Square complexes are a special case of a high-dimensional object known as *cubical complex*. We can naturally extend the above definition to contain sets $Z(3)$, $Z(4)$, etc as sets containing "cubes", or 4-dimensional hypercubes, under a similar downward closure property. In this sense, square complexes are 2-dimensional examples of cubical complexes.

We note that it is common in the literature (especially the pure mathematics one) to define square complexes algebraically in the language of chain complexes. Since we won't make use of the extra algebraic properties that come with this definition, we opted to use the simpler combinatorial one, which is also implicitly in [DEL+22].

We define the *1-skeleton* of a square complex Z to be the graph $(Z(0), Z(1))$. To characterize the relationships between layers of Z, we define the *vertex-association* and *edge-association* as follows:

Definition 5.10 (Vertex- and Edge-Association)**.** Given a square complex Z and a vertex v, we define the *vertex-association* L_v of v to be the set of squares of which v is a member. Mathematically, $L_v = \{\tau : v \in \tau, \tau \in Z(2)\}$. Similarly, (with some abuse of notation), for an edge e, we define its *edge-association* to be $L_e = \{\tau : e \in \tau, \tau \in Z(2)\}$.

Note that these quantities are analogous to links in simplicial complexes, in the sense that they represent the "local neighborhoods" of vertices and edges.

We can furthermore extend the notion of regularity to square complexes. That is, we say a square complex is *d-vertex regular* if its 1-skeleton forms a d-regular graph, and *k-edge regular* if $\forall e \in Z(1), |L_e| = k$. We denote the vertex-regularity of a square complex Z as $d_Z^{|}$ and the edge regularity as $d_Z^{\square}$.

To define our notion of expansion in square complexes, we first need to define parallelism.

Definition 5.11. Given a square complex Z, we say that an edge e is parallel to e' if both e and e' are members of a common square, but share no vertices. Thus, each square $s = (v_1, v_2, v_3, v_4)$ has exactly two pairs of parallel edges: $(v_1, v_2) \parallel (v_3, v_4)$ and $(v_2, v_3) \parallel (v_1, v_4)$. Additionally, given an edge e, we define its *parallel adjacency* $E^{\parallel}(e)$ to be the set of all edges it is parallel to. That is, $E^{\parallel}(e) = \{e' : e' \parallel e, \{e, e'\} \subset s \in Z(2)\}$.

The above definition induces a notion of parallel-skeleton, which is a graph G whose vertices are the edges $Z(1)$ and edges are given by the sets $E^{\parallel}$.

[1] In the world of simplicial complexes the notion of *pure simplicial complex* plays a similar role in making this distinction.

Definition 5.12. The global expansion of a square complex Z is given by the $\widetilde{\lambda}(\text{1-skeleton}(Z))$. The parallel expansion of a square complex Z is given by the $\widetilde{\lambda}(\text{parallel-skeleton}(Z))$.

We will use the notation of Markov operators to represent the above expansions. Let $M_Z^{\downarrow}$ be the Markov operator of the random walk on the 1-skeleton, i.e. $\lambda(M_Z^{\downarrow})$ is the global expansion of Z. The *parallel random walk* on a square complex is given as the random walk defined on edges, such that given an edge we pick one parallel edge uniformly at random to go to. This is a natural generalization of the random walk on G, as the two vertices $(v_1, v_2) \in E(G)$ are parallel to each-other with respect to edges. So, let $M_Z^{\square}$ to be the Markov operator of the parallel random walk, i.e. $\lambda(M_Z^{\square})$ is the parallel expansion of Z.

The definition of expansion we use is also implicitly used in [DEL+22], and it is equivalent to other definitions seen in the literature. For example, in [LZ22] parallel adjacency is replaced by a form of "corner adjacency", defined between vertices, where two vertices are "corner adjacent" if they are on opposite corners of a square. We prefer to use the definition of parallel adjacency since it gives us an adjacency between edges that is very similar to the definition of a "up" operator from edges to squares that given an edge it picks a random square containing that edge. This "up" operator is not very nice to work with since it is not symmetric, so the parallel operator is a better choice.

Non spectral notions of expansion have also been studied. The work of Jordan and Livné [JL00] defined and constructed *Ramanujan cubical complexes*, in a way analogous to the work of [LSV05]. Additionally, in [DLV24] certain topological expansion definitions of cubical complexes are studied, motivated by the construction of quantum locally testable codes.

5.3 Dinur et al's c^3-LTCs

We now present the c^3-LTCs of Dinur et al, which resolved a longstanding open problem asking if there exist a family of error correcting codes with asymptotically constant rate, constant distance, and constant locality. This result is particularly nice as it is a direct application of the topics we have discussed so far. The main result of [DEL+22] is as follows:

Theorem 5.13. *For every $0 < r < 1$, there exist $\delta, \kappa > 0$ and $q \in \mathbb{N}$ and a polynomial-time construction of an infinite family of error correcting codes $\{C_n\}$ with rate r and distance δ, such that for all n, C_n is κ-locally testable with q queries.*

Namely, every code C_n comes with a randomized local tester that reads at most q bits from a given word w and then accepts or rejects, such that

- *For all $w \in C_n$, $\mathbf{Pr}[accept] = 1$.*
- *For all $w \notin C_n$, $\mathbf{Pr}[reject] \geqslant \kappa \cdot \delta(w, C_n)$.*

The intuition behind their construction is to construct a special type of expanding Square Complex, termed a Left-Right Cayley complex. Then, an error correcting code is defined on square complex by identifying each square with a bit, and constraining the bits based on the underlying structure of the Left-Right Cayley Complex. Importantly, if the square complex has good expansion, then the constraints will 'mix well' which leads to the code being locally testable.

Definition 5.14 (Left-Right Cayley Complex)**.** Let G be a group with two symmetric sets of generators A, B, namely, each is closed under taking inverses. We assume that the identity element of G is neither in A nor in B. Define the *Left-Right Cayley Complex* $X = Cay^2(A, G, B)$ as follows

- The vertices are $X(0) = G$.
- The edges are $X(1) = X^A(1) \sqcup X^B(1)$ where

$$X^A(1) = \{\{g, ag\} \mid \ g \in G, a \in A\}, \qquad X^B(1) = \{\{g, gb\} \mid \ g \in G, b \in B\}.$$

- The squares are $X(2) = A \times G \times B / \sim$ where for every $a \in A, b \in B, g \in G$,

$$(a, g, b) \sim (a^{-1}, ag, b) \sim (a^{-1}, agb, b^{-1}) \sim (a, gb, b^{-1}),$$

  and denote the equivalence class of (a, g, b) by $[a, g, b]$, so

$$[a, g, b] = \{(a, g, b), (a^{-1}, ag, b), (a^{-1}, agb, b^{-1}), (a, gb, b^{-1})\}.$$

Observe that the graph $(X(0), X^A(1))$ is exactly the Cayley graph $Cay(G, A)$ and that $(X(0), X^B(1))$ is the Cayley graph $Cay(G, B)$.

The Left-Right Cayley complexes are equipped with notation inherited from HDX literature. That is, for each $g \in G$, the link of g is $X_g = \{[a, g, b] \mid \ a \in A, b \in B\}$. Additionally, for every edge $e = \{g, ag\}$, the link of e is denoted $X_e = \{[a, g, b] \mid \ b \in B\}$, and if $e = \{g, gb\}$ we let $X_e = \{[a, g, b] \mid \ a \in A\}$.

In order to prevent degenerate squares, it must be that $g \neq agb$, or equivalently, $g^{-1}ag \neq b$ for all $a \in A, g \in G, b \in B$. This is codified as the total no-conjugacy condition (TNC). Much of the algebraic machinery of Dinur et al's work is towards showing their exists good expanding Left-Right Cayley complexes for which TNC holds. For our purposes, we assume this is true, so we can define the relevant error correcting code.

Definition 5.15 (Dinur et al's c^3-LTC)**.** Let G, A, B and $X = Cay^2(G, A, B)$. Recall that for any vertex $g \in X(0)$ (resp. any edge $e \in X(1)$) we denote by $X_g \subset X(2)$ (resp. $X_e \subset X(2)$) the set of squares in X containing the vertex g (resp. the edge e). Let $C_A \subset \mathbb{F}_2^A$ and let $C_B \subset \mathbb{F}_2^B$ be two fixed linear error correcting codes with rates $\rho_A = (C_A), \rho_B = (C_B)$ and distances $\delta_A = (C_A), \delta_B = (C_B)$, respectively. Define the code $C = C[G, A, B, C_A, C_B]$ as follows. For an edge $e = \{g, ag\} \in X^A(1)$, we define a local code

$$C_e = \{f : X_e \to \mathbb{F}_2 \mid \ f([a, g, \cdot]) \in C_B\}.$$

Similarly, for an edge $e = \{g, gb\} \in X^B(1)$, we define a local code

$$C_e = \{f : X_e \to \mathbb{F}_2 \mid \ f([\cdot, g, b]) \in C_A\}.$$

Note that this definition appears to depend on the choice of $g \in e$ but it does not. Finally, we define a global code

$$C = \{f : X(2) \to \mathbb{F}_2 \mid \ \forall e \in X(1), f|_{X_e} \in C_e\}.$$

This code has a very nice interpretation: in particular, the core idea of the ([**?**]) construction was to "raise Tanner codes one dimension up" to induce local testability. The use of the Left-Right Cayley Complex is necessary because it causes the parity checks to be very dependent on each other (by its expansion). Additionally, the use of "squares" causes the code to locally look like a tensor code, which is actually key to the locality tester of the code. In particular, observe that for each $g \in G$, the codes C_e induce the local tensor code

$$C_g = \left\{ f : X_g \to \mathbb{F}_2 \;\middle|\; f([\cdot, g, \cdot]) \in C_A \otimes C_B \right\}.$$

One way to see that C_g is a tensor code is by viewing it as a matrix with all the generators $a \in A$ indexing the rows and all the generators $b \in B$ indexing the columns and applying the definition of C_e for the a- edges and b- edges. This view of C as a lifted tensor code is critical, as it directly yields a natural tester for $\mathcal{C}$. Indeed, given an arbitrary bit-string w, we can test if it is a code-word of $\mathcal{C}$ by first sampling uniformly at random some $g \in G$ and then accepting if $w_g \in \mathcal{C}_g$ and rejecting if not. [DEL+22] prove that this is a good tester by showing that one can utilize this local test as a corrector for the code $\mathcal{C}$.

In particular, when given a string $f : X(2) \to \mathbb{F}_2$ the local corrector will either find the correct codeword $c \in \mathcal{C}$ or give up. Before writing out the algorithm, we define the following quantities:

Definition 5.16 (Local Views and Disagreement)**.** For each vertex g, denote its *local view* $L_g \in C_g$ to be the closest codeword to $f|_{X_g}$ (breaking ties arbitrarily). Given a collection of local views $W = \left\{ L_g \in C_g \;\middle|\; g \in G \right\}$, we define the *disagreement* of the collection to be

$$\Delta(W) = \mathbf{Pr}_{e=\{g,g'\} \in X(1)} [Ł_g|_{X_e} \neq Ł_{g'}|_{X_e}]$$

where e is a uniformly random edge in $X(1)$.

Observe that the disagreement computes the number of edges (parity checks) where the local test and code-word f disagree. The local self-correction algorithm below works because it monotonically decreases $\Delta(W)$, and thus will always lead to a valid codeword. The local-correction algorithm can be described as follows.

1. **Initialization:** For each vertex g, choose choose the codeword $Ł_g^0 \in C_g$ that is closest to the restriction $f|_{X_g}$, that is,
$$L_g^0 = \text{argmin}_{w \in C_g} \delta(w, f|_{X_g}).$$
Set $L_g \leftarrow L_g^0$ for every $g \in G$, and intialize the set of local views to be $W = \{Ł_g\}$.

2. **Iterative update:** As long as there exists some vertex g and a $w \in C_g$ such that updating $Ł_g \leftarrow w$ decreases $\Delta(W)$, perform the update. Repeat this process until no further improvement is possible.

3. **Final step:** If the final disagreement value satisfies $\Delta(W) > 0$, return "far." Otherwise, when $\Delta(W) = 0$, construct $f_0 : X(2) \to \mathbb{F}_2$ by, for each square $s \in X(2)$, selecting a vertex $g \in s$ and setting $f_0(s) = Ł_g(s)$. Output f_0.

We do not present the proof of why this algorithm works, as it is quite involved (see [DEL+22] for the full proof). We also remark that if one were to instantiate their code with arbitrary square complex and base codes C_A, and C_B, they would not necessarily obtain a c^3-LTC, as the choice of the code must be done carefully to maintain the so-called 'agreement testability' of the tensor codes. The choice of the Square Complex is also critical, as the walks must have near-Ramanujan expansion, but also must have well-defined labels (which in the Left-Right Cayley complex are given by the generators). If any one of these criterion are not met, then the resulting code may not be as interesting.

5.4 An Analogous Theory of Cubical Complexes

To motivate the study of cubical complexes, we point the reader to [DLV24] which describes the use of a cubical complex towards construction of "almost-good" quantum LTCs. The cubical complexes constructed in their paper can be thought of as a higher-dimensional analog of the Left-Right Cayley complexes in [DEL+22]. In a more general sense, cubical complexes can be thought of as a higher-dimensional analog od square complexes. To further belabor this point, much of the notation of cubical complexes are inherited from square complexes.

Definition 5.17 (Cubical Complex)**.** A d-dimensional *cubical complex* $Z = (Z(0), Z(1), \cdots, Z(d))$ are defined as collections of subsets of a vertex set V that satisfy the following:

- The vertices, denoted by $Z(0)$, are exactly the vertex set V, and the the i-dimensional layer, denoted by $Z(i)$, are sub-sets of size 2^i vertices.
- Every subset $\sigma \in Z(i)$ for $0 \leqslant i \leqslant d$ satisfies a weakened notion of *downward closure*. That is, if $\sigma \in Z(i)$, then it must be $\{\tau \subseteq \sigma \mid |\tau| = 2^j\} \subseteq Z(j)$ for all $j \leqslant i$.

Much like simplicial complexes, we say that a d-dimensional cubical complex Z is a *pure* cubical complex, if all faces are a subset of some $\sigma \in Z(d)$. In other words, the cubical complex is completely defined by specifying the top layer $Z(d)$. For our purposes, we solely consider pure cubical complexes.

Remark 5.18. One key distinction between the definition of cubical complexes is that the squares were equivalent to 4-cycles in Definition 5.7, and hence the order of the set mattered. In cubical complexes, we assume a fixed ordering defined by $\succ (\cdot, \cdot)$ of V, and that all subsets are ordered according to $\succ$, so a face is exactly defined by its elements.

The notions of vertex and edge-associations from square complexes are extended to cubical complexes as expected.

Definition 5.19 (Cube- Association)**.** Given a d-dimensional cubical complex Z and a face $\sigma \in Z(i)$ where $0 \leqslant i < d$, we define the *cube-association* L_σ of σ to be $L_\sigma = \{\tau : \sigma \in \tau, \tau \in Z(i+1)\}$. We also extend this to 'higher dimensional neighborhoods' by defining the *k-higher cube association* for any $\sigma \in Z(i)$ where $0 \leqslant i \leqslant d - k$ as $L_\sigma(k) = \{\tau : \sigma \in \tau, \tau \in Z(i+k)\}$. The *complete cube-association* for $\sigma \in Z(i)$ is given by $L_\sigma(C) = \cup_{k=1}^{d-i} L\sigma(k)$

Observe that the Definition 5.19 from square complexes is exactly the 1-higher cube association. Much like square complexes, these definitions are analogous to links in simplicial complexes. Parallelism is also extended to cubical complexes in a typical fashion.

Definition 5.20 (Cubical Parallel Adjacency)**.** Given a cubical complex Z, we say that an face $\sigma \in Z(i)$ for $0 \leqslant i < d$ is parallel to σ' ($\sigma \parallel \sigma'$) if $\sigma \cup \sigma' \in Z(i+1)$ and $\sigma \cap \sigma' = \emptyset$. Additionally, given a face $\sigma \in Z(i)$ for $0 \leqslant i < d$, we define its *cubical parallel adjacency* $Z^{\parallel}(\sigma)$ to be $Z^{\parallel}(\sigma) = \{\sigma' : \sigma' \parallel \sigma\}$, that is the set of all faces it is parallel to.

Much like the parallelism on square complexes induced a parallel-skeleton from which we defined its expansion, the cubical complex induces a *skeleton hierarchy*, which is defined as follows.

Definition 5.21 (Skeleton hierarchy)**.** Given any d-dimensional cubical complex, and $0 \leqslant i < d$, we define its $(i, i+1)$-*skeleton* to be the graph $G^{(i,i+1)}$ where $V(G) = Z(i)$, and $E(G) = \{\{\sigma, \sigma'\} \in V(G) \times V(G) \mid \sigma \parallel \sigma'\}$.

The *skeleton hierarchy* $SkH(Z)$ of Z is defined as a collection of graphs $SkH(Z) = \{G^{(i,i+1)}\}_{i=0}^{d-1}$

Recall that the parallel random walk on square complexes is the walk where at each step, we start from a face σ, and uniformly at random move to face σ' that is parallel to it. This is precisely the definition of parallel random walk that we define for cubical complexes: that is, for all $0 \leqslant i < d$, the *i-layer parallel random walk* is the walk, where one step is defined by starting at σ, and uniformly at random outputs $\sigma' \in Z^{\parallel}(\sigma)$. Observe that one step of the *i-layer parallel random walk* is exactly equivalent to one step on the graph $G^{(i,i+1)}$ from the skeleton hierarchy.

This relationship between the parallel random walk and the skeleton hierarchy precisely yield a convenient definition of regularity for cubical complex. That is a cubical complex Z is $(r^{(0,1)}, r^{(1,2)}, \cdots, r^{(d-1,d)})$-regular if $G^{(i,i+1)} \in SkH(Z)$ is a $r^{(i,i+1)}$ -regular graph. Regularity ensures we have control over the top eigenvalue of the spectrum of all $G^{(i,i+1)} \in SkH(Z)$, so we can define expansion on an expanding cubical complex as follows.

Definition 5.22 (Expanding Cubical Complex)**.** A d-dimensional regular cubical complex Z with $(\lambda^{(0,1)}, \lambda^{(1,2)}, \cdots, \lambda^{(d-1,d)})$-expanding if for all $G^{(i,i+1)} \in SkH(Z)$, $\lambda(G^{(i,i+1)}) \leqslant \lambda^{(i,i+1)}$. We say a cubical complex is λ- globally expanding if $\max_{G \in SkH(Z)} \lambda(G) \leqslant \lambda$

6 Combinatorial Constructions of Expanding Complexes

Their is quite a rich literature on constructions of HDXs. One known example is in [DK17], which showed that by picking appropriate parameters of Ramanujan complexes of [LSV05], one can construct constant bounded degree simplicial complexes with global and local expansion bounded by λ, for any $\lambda > 0$. This construction (and to our knowledge, most HDX constructions) is heavily algebraic and relies on quite involved mathematical tools. Contrary to the case of graph expanders, more elementary constructions relying mostly on combinatorics or randomization seem to be harder to find. Notably, the analog of Friedman's theorem isn't true, that is, random simplicial complexes are not good high dimensional expanders. A few constructions obtaining suboptimal parameters (when compared to the algebraic ones mentioned above) have been proposed, of which we will highlight two that follow a product based approach that our main result is inspired by.

Liu-Mohanty-Yang [LMY20] and Golowich [Gol21] defined a variant of a tensor product between a graph and a small simplicial complex and by picking the appropriate parameters construct bounded degree simplicial complexes with global expansion $1/2$ and k-local expansion $\Theta(1/k)$.

As our techniques draw from [LMY20] and [Gol21], we give a short overview on Golowich's HDX construction and its guarantees in the next section. We then present our main result in Section 6.2.1, which is a product-based construction of square complexes with expansion and regularity guarantees. In Section 6.2.2, we define a more general construction of cubical complexes, which reasonably could provide constant expansion. We discuss a line of reasoning on some favorable aspects of this construction and identify some technical hurdles that must be overcome.

6.1 Golowich's HDX Construction

Golowich improved upon a construction by [LMY20] for combinatorially constructing HDXs from graph products. The complex presented in his paper is weighted. For simplicity, we present an unweighted variant of this construction.

Definition 6.1. [Golowich's HDX construction] Let $G = (V(G), E(G), w_G)$ be a simple graph on n vertices. Given positive integers H and s, we define the H-dimensional simplicial complex Z with vertex set $V(G) \times [s]$ so that

$$Z(H) = \{\{(v_1, b_1), \ldots, (v_{H+1}, b_{H+1})\} \subset V(G) \times [s] : \exists \{u, v\} \in E(G) \text{ s.t. } \{v_1, \ldots, v_{H+1}\} = \{u, v\}, \\ b_1, \ldots, b_{H+1} \text{ are all distinct}\}.$$

Their construction provides guarantees on the spectral gap of the resulting simplicial complex. In particular, they have that

Theorem 6.2 (([Gol21])). *Let $W_k^{\uparrow\downarrow}$ be the up-down walk operator for the simplicial complex Z of formed by Definition 6.1. If $H \geqslant 2$, $s \geqslant 2H$, and $n \geqslant 4$, then for every $0 \leqslant k \leqslant H-1$,*

$$\nu_2(W_k^{\uparrow\downarrow}) \geqslant \frac{\nu_2(G)}{(1+\log H)(k+2)(k+1)}.$$

At a very high level, the proof (contained in [Gol21]) functions by analyzing the expansion of the links, and then applying the local-to-global paradigm of HDXs.

6.2 Our Contributions

6.2.1 Generalized Corner Product

Consider two graphs $G = (V_G, E_G)$ and a graph $H = (V_H, E_H)$. We define the *General Corner Product* between these two graphs to take in these graphs as input and output a square complex with:

- $Z(0) = V_G \times V_H$
- $Z(2) = \{((v_1, u_1), (v_2, u_2), (v_3, u_3), (v_2, u_4)) : (v_1, v_2) \in E_G, (v_2, v_3) \in E_G, (u_1, u_3) \in E_H, (u_2, u_4) \in E_H\}$

and $Z(1)$ is defined to be all the edges implied by the existence of the 4-cycles in $Z(2)$. Our result revolves around determining the properties of the square complex resulting from this operation. Indeed, our main result is the below theorem.

Theorem 6.3. *Suppose that $G = (V_G, E_G)$ is a d_G- regular λ_G- expanding graph with $|V_G| = n_G$ and that $H = (V_H, E_H)$ is d_H- regular, λ_H- expanding graph with $|V_H| = n_H$. The square complex Z resulting from the General Corner Product of G and H has the following properties.*

- $d_Z^{|} = d_G n_H$, *in other words, the 1-skeleton of Z is a $d_G n_H$ regular graph.*
- *The (normalized) expansion of the random walk on the 1-skeleton of Z is $\lambda(M_Z^{|}) \leqslant \tilde{\lambda}_G$*
- *The edge regularity $d_Z^{\square}$ is $2(d_G - 1)d_H^2$.*
- *The (normalized) expansion of the parallel random walk on Z is $\lambda(M_Z^{\square}) \leqslant \max(\frac{d_G + d_G\lambda_G - 2}{2d_G - 2}, \lambda(\tilde{M}_H))$*

We prove this Theorem in the next section. An immediate consequence demonstrating its utility is the following.

Corollary 6.4. *There exists a square complex with (un-normalized) $\lambda_Z^{|} \leqslant \mathcal{O}((d_Z^{|})^{0.751})$ and $\lambda_Z^{\square} \leqslant \frac{1}{2} + \epsilon$. The vertex and edge-regularity of this construction will both be $\mathcal{O}(d_G^3)$.*

Proof of Corollary 6.4 (assuming Theorem 6.3): Set G and H to both be Ramanujan expanders with $n_H = d_G^{1+\epsilon}$ and $d_H = d_G$. Take the generalized corner product of G and H. By Theorem 6.3, this results in a square complex with 1-skeleton-expansion $\frac{2}{\sqrt{d_G - 1}}$ and parallel expansion $\max(\frac{d_G + d_G\lambda_G - 2}{2d_G - 2}, \lambda(\tilde{M}_H))$. This complex has $d_Z^{|} = d_G n_H$ and $d_Z^{\square}$ is $2(d_G - 1)d_H^2$, so the un-normalized expansion of $M_Z^{|}$ is $\frac{2d_G n_H}{\sqrt{d_G}}$. Plugging in n_H and d_H and analyzing the asymptotics yields

$$\lambda(M_Z^{|}) = \mathcal{O}\left(d_G^{3/2+\epsilon}\right) \text{ and } d_Z^{|} = \mathcal{O}\left(d_G^{2+\epsilon}\right) \implies \lambda(M_Z^{|}) = \mathcal{O}\left(d_Z^{| \frac{3/2+\epsilon}{2+\epsilon}}\right)$$

as desired □

The rest of this section is devoted to proving Theorem 6.3, which we have broken apart into three smaller proofs.

Proof of Theorem 6.3 for vertex-regularity and 1-skeleton expansion: Note that the definition of $Z(2)$ implies that $Z(1) = \{((v_1, u_1), (v_2, u_2)) : (v_1, v_2) \in E_G\}$. It thus follows that every vertex has exactly $d_G n_H$ neighbors. These observations enable us to quickly deduce that the 1-skeleton of Z is λ_G-expanding. This is because a random walk starting from a vertex (v, u) is equivalent to first sampling a neighbor of $v \in G$ and then a random vertex in H. Because this sampling process is expressed by the operator $\tilde{M_G} \otimes \tilde{J_{n_H}}$, its second largest eigenvalue is exactly $\tilde{\lambda_G}$ (because its eigenvalues are given by the products of the eigenvalues of $\tilde{M_G}$ and $\tilde{J_{n_H}}$) □

Proof of Theorem 6.3 for edge-regularity: We claim that Z is $2(d_G - 1)d_H^2$ edge-regular. To see this, we apply the following constructive counting argument. We begin with some edge $((v_1, u_1), (v_2, u_2))$, so every square containing this edge must be of the form: $((v_1, u_1), (v_2, u_2), (\cdot, \cdot), (\cdot, \cdot))$. Note that every edge has one clone vertex, WLOG, let it be (v_2, u_2), this means that our squares can take the form $((v_1, u_1), (v_2, u_2), (\cdot, \cdot), (v_2, \cdot))$. Note that the first missing coordinate can take on any of the $d_G - 1$ neighbors of v_2 besides v_1, and the second missing coordinate can be any neighbor of u_1, so it has d_H choices and the third missing coordinate is required to be a neighbor of u_2, so it has d_H choices. Putting this together (and multiplying by 2 due to choice of clone vertex) gives us an edge-regularity of $2(d_G - 1)d_H^2$. □

Proof of Theorem 6.3 for parallel-expansion: Our strategy is to simulate the parallel random walk via a combinatorial process $\mathcal{P}$, express the process in terms of operators, and then bound the spectrum. The following process $\mathcal{P}$represents the parallel random walk.

1. We begin with some edge $e = (u, v) \in Z(1)$. Note that $v_1 = (v_{1G}, v_{1H})$ and $v_2 = (v_{2G}, v_{2H})$. Take a step on the line graph of G (using $l(G)$), call the output $e'_G = (v'_{1G}, v'_{2G})$
2. Take a random walk step on the graph H using the vertex v_{1H}, call the output vertex v'_{1H}. Similarly, take a random walk step on the graph H using the vertex v_{2H}, call the output vertex v'_{2H}.
3. Output the edge $((v'_{1G}, v'_{1H}), (v'_{2G}, v'_{2H}))$.

We can see that $\mathcal{P}$ exactly replicates the constructive argument used in the proof for edge-regularity, by noting that we start with the edge $((v_1, u_1), (v_2, u_2))$ and then filling out the blanks uniformly in the set of possible squares: $((v_1, u_1), (v_2, u_2), (\cdot, \cdot), (v_2, \cdot))$. Note that the operator $\tilde{M}_{l(G)} \otimes (\tilde{M}_H \otimes \tilde{M}_H)$ encapsulates $\mathcal{P}$. The operator $\tilde{M}_{l(G)}$ enacts step 1, step 2 is done by $\tilde{M}_H \otimes \tilde{M}_H$. Finally, the coordinates are joined together by the Kronecker product in step 3. The eigenvalues of the matrix $\tilde{M}_{l(G)} \otimes (\tilde{M}_H \otimes \tilde{M}_H)$ are given by the products of the eigenvalues of $\tilde{M}_{l(G)}, \tilde{M}_H$, and$\tilde{M}_H$. It follows that the second largest eigenvalue is $\max(\lambda(\tilde{M}_{l(G)}), \lambda(\tilde{M}_H)) = \max(\frac{d_G + d_G\tilde{\lambda}_G - 2}{2d_G - 2}, \lambda(\tilde{M}_H))$. □

6.2.2 Cubical Complexes from Graph Products

We now show our construction of Cubical Complexes, which can be interpreted as the cubical complex variant of Golowich's HDX construction. Though the constructions are very similar, the proof techniques of [Gol21] are not applicable to the cubical complex setting due to the differing notions of walks and links/associations of simplicial and cubical complexes.

Definition 6.5 (Combinatorial Cubical Complex Construction)**.** Let $G = (V(G), E(G))$ be any simple, connected graph on n vertices. For positive integers H and s, define the H-dimensional cubical complex Z with vertex set $V(G) \times [s]$ such that

- $Z(0) = V(G) \times [s]$
- $Z(H) = \left\{ \{(v_i, b_i)\}_{i=1}^{2^H} \subset V(G) \times [s] \;\middle|\; \begin{array}{l} \exists \{u, v\} \in E(G) \text{ such that } \{v_i\}_{i=1}^{2^H} \subseteq \{u, v\}, \\ b_1, \ldots, b_{2^H} \text{ are all distinct} \end{array} \right\}$

where $Z(1), \cdots, Z(H-1)$ are given directly from the downward closure of Z

We refer to Definition 6.5 as the *C4* construction for alliteration, succinctness and allusion to its explosive nature. It is not unreasonable to suggest that the construction will yield constant expansion (in the size, not the dimension of the faces). We now discuss a methodology for analyzing the expansion of $G^{(i,i+1)} \in SkH(Z)$

The parallel walk starting at σ will output uniformly random $\sigma' \in Z^{\|}(\sigma)$. Observe that this is exactly equivalent to uniformly selecting from $\{\tau \in Z(i+1) : \sigma \in \tau\}$ and then outputting $\tau \setminus \sigma$.

Now, we attempt to follow the strategy for the proof of Theorem 6.3, that is we represent the walk on $G^{(i,i+1)}$ as some combinatorial process, represent it as a tensor product of operators, and analyze the resulting operator spectrum.

Let M denote the operator that acts on the left coordinates $\{v_i\}_{j=1}^{2^i}$ and A the operator that acts on the right coordinates $\{b_i\}_{j=1}^{2^i}$ in one step of the parallel random walk. Observe that the there is no dependence between A and M, and thus the parallel random walk operator is exactly $M \otimes A$. From above discussion and definition of cubic parallelism, note that A exactly maps a 2^i-size subset S of $[s]$ to a uniformly random disjoint subset 2^i-size subset S' of $[s]$. We are able to fully characterize the spectrum of A by identifying it with the random walk operator on the Kneser graph, which is defined as follows.

Definition 6.6 (Kneser Graph)**.** The *Kneser graph* $K(s,k)$ is an undirected graph with its vertex set $V(K(s,k))$ being all $\binom{s}{k}$ k-size subsets of $[s]$ and an edge between subsets v_1, v_2 if and only if $v_1 \cap v_2 = \emptyset$

These graphs have been well-studied, and in fact, there is known to be a closed form for its spectrum.

Lemma 6.7 ([GR01])**.** *Let $K(s,k)$ The eigenvalues of the adjacency matrix of $K(s,k)$ are given by:*

$$\lambda_i = (-1)^i \binom{s-k-i}{k-i}, \quad \text{for } 0 \leqslant i \leqslant k.$$

where the eigenvalue λ_i has multiplicity $m_i = \binom{s}{i} - \binom{s}{i-1}$, and $\binom{s}{-1}$ is defined to be 0.

Thus, to fully compute the spectrum of $G^{(i,i+1)}$, it suffices to characterize the spectrum of M. However, this problem is actually quite difficult due to the asymmetry between different faces of the complex from the C4 construction. In particular, observe that every set of left coordinates $\{v_i\}_{j=1}^{2^i}$ has projection $\{v\}$ or $\{u, v\}$, and that faces with projection being a single-element $v \in V(G)$ have different regularities than faces with projection $\{u, v\} \in E(G)$. We codify this distinction as follows: given a cubical complex Z resulting from the C4 construction, and a face $\sigma \in Z(i)$, denote $L_i(\sigma)$ to be the set of unique elements in $\{v\}_{i=1}^{2^i}$. From our earlier discussion, $\{\sigma \in Z(i) \mid |L_i(\sigma)| = 1\}$ and $\{\sigma \in Z(i) \mid |L_i(\sigma)| = 2\}$ partition $Z(i)$. Now, observe that the following must be true:

Lemma 6.8. *Let Z be the H-dimensional cubical complex arising from the $C4$ construction with d-regular graph G and s as input. Given $G^{(i,i+1)} \in SkH(Z)$, and a vertex $v \in V(^{(i,i+1)})$,*

- *If $v \in \{\sigma \in Z(i) \mid |L_i(\sigma)| = 1\}$, then $deg(v) = \binom{s-k}{k}(d(2^k - 1) + 1)$*
- *If $v \in \{\sigma \in Z(i) \mid |L_i(\sigma)| = 2\}$, then $deg(v) = \binom{s-k}{k}2^k$*

where $k = 2^i$.

Proof. The proof follows a simple constructive counting argument. In both cases, we have that the set of right hand coordinates $\{b_j\}_{j=1}^k$ is exactly the regularity of the Kneser graph, which is $\binom{s-k}{k}$. Now, we breakdown the two cases separately.

- Denote $\sigma \in Z(i)$ to be the face corresponding to $v \in \{\sigma \in Z(i) \mid |L_i(\sigma)| = 1\}$, any face $\sigma' \in Z(i)$ parallel to σ must have either
  - $L_i(\sigma') = \{v\}$
  - $L_i(\sigma') = \{u\}$ where $\{u, v\} \in V(G)$
  - $L_i(\sigma') = \{u, v\}$ where $\{u, v\} \in V(G)$

  Because the choice of right coordinates distinguishes between coordinates, the problem reduces to counting sequences. There is 1 sequence in the first sub-case, d in the second sub-case, and $d(2^k - 2)$ in the third sub-case.
- For the case $v \in \{\sigma \in Z(i) \mid |L_i(\sigma)| = 2\}$, we proceed almost exactly as in the first case.
  - $L_i(\sigma') = \{v\}$, of which there is exactly 1 sequence
  - $L_i(\sigma') = \{u\}$ where $\{u, v\} \in V(G)$, of which there is exactly 1 sequence
  - $L_i(\sigma') = \{u, v\}$ where $\{u, v\} \in V(G)$, of which there are exactly $2^k - 2$ sequences.

Putting together the left and right coordinates yields the desired result. □

Remark 6.9. The main distinction between vertices of the first-type and second-type is that in the first type, we have a degree of freedom to pick a random neighbor in $V(G)$, whereas in the second case, we do not. While this asymmetry makes characterizing M difficult, it is critical for expansion, as uniformly selecting a neighbor $u \sim v$ in the parallel random walk is what enables Z to inherit the expansion properties of G

Observe that the constructive argument utilized in the above proof exactly lends a sampling process for M based on the starting point. The technical hurdle is overcoming the asymmetry between the two types of faces. One approach could be adding a weight function on the faces which could correct the "imbalances" arising from the $C4$ construction. We leave this as future work.

References

[AL20] Vedat Levi Alev and Lap Chi Lau. Improved analysis of higher order random walks and applications. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 1198–1211, New York, NY, USA, 2020. Association for Computing Machinery. 26

[Alo86] N. Alon. Eigenvalues and expanders. volume 6, pages 83–96. 1986. Theory of computing (Singer Island, Fla., 1984). 18

[Alo21] Noga Alon. Explicit expanders of every degree and size. *Combinatorica*, 41(4):447–463, 2021. 18

[Bor20] Charles Bordenave. A new proof of Friedman's second eigenvalue theorem and its extension to random lifts. *Ann. Sci. Éc. Norm. Supér. (4)*, 53(6):1393–1439, 2020. 18

[CY24] Gil Cohen and Tal Yankovitz. Asymptotically-good rlccs with $(\log n)^{2+o(1)}$ queries. In *39th Computational Complexity Conference (CCC 2024)*, volume 300 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 8:1–8:16, 2024. 24

[DEL+22] Irit Dinur, Shai Evra, Ron Livne, Alexander Lubotzky, and Shahar Mozes. Locally testable codes with constant rate, distance, and locality. In *STOC '22—Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 357–374. ACM, New York, 2022. 1, 3, 5, 6, 24, 27, 28, 30, 31

[DK17] Irit Dinur and Tali Kaufman. High dimensional expanders imply agreement expanders. In *58th Annual IEEE Symposium on Foundations of Computer Science—FOCS 2017*, pages 974–985. IEEE Computer Soc., Los Alamitos, CA, 2017. 25, 33

[DLV24] Irit Dinur, Ting-Chun Lin, and Thomas Vidick. Expansion of higher-dimensional cubical complexes with application to quantum locally testable codes. *arXiv preprint arXiv:2402.07476*, 2024. 3, 5, 28, 31

[Fri93] Joel Friedman. Some geometric aspects of graphs and their eigenfunctions. *Duke Math. J.*, 69(3):487–525, 1993. 18

[Fri08] Joel Friedman. A proof of Alon's second eigenvalue conjecture and related problems. *Mem. Amer. Math. Soc.*, 195(910):viii+100, 2008. 18

[Gol21] Louis Golowich. Improved product-based high-dimensional expanders. In *Approximation, randomization, and combinatorial optimization. Algorithms and techniques*, volume 207 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 38, 17. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2021. 3, 4, 6, 33, 35

[GR01] Chris Godsil and Gordon Royle. *Algebraic Graph Theory*, volume 207 of *Graduate Texts in Mathematics*. Springer, 2001. 36

[Gur10] Venkatesan Guruswami. Notes 2: Gilbert–varshamov bound. https://www.cs.cmu.edu/~venkatg/teaching/codingtheory/notes/notes2.pdf, 2010. Introduction to Coding Theory, CMU. 12

[JL00] Bruce W. Jordan and Ron Livné. The Ramanujan property for regular cubical complexes. *Duke Math. J.*, 105(1):85–103, 2000. 28

[LMY20] Siqi Liu, Sidhanth Mohanty, and Elizabeth Yang. High-dimensional expanders from expanders. In *11th Innovations in Theoretical Computer Science Conference*, volume 151 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 12, 32. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2020. 33

[LPS88] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988. 18

[LSV05] Alexander Lubotzky, Beth Samuels, and Uzi Vishne. Explicit constructions of Ramanujan complexes of type $\widetilde{A}_d$. *European J. Combin.*, 26(6):965–993, 2005. 25, 28, 33

[LZ22] Anthony Leverrier and Gilles Zémor. Quantum Tanner codes. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science—FOCS 2022*, pages 872–883. IEEE Computer Soc., Los Alamitos, CA, [2022] ©2022. 28

[Mar88] G. A. Margulis. Explicit group-theoretic constructions of combinatorial schemes and their applications in the construction of expanders and concentrators. *Problemy Peredachi Informatsii*, 24(1):51–60, 1988. 18

[MOP22] Sidhanth Mohanty, Ryan O'Donnell, and Pedro Paredes. Explicit near-Ramanujan graphs of every degree. *SIAM J. Comput.*, 51(3):STOC20–1–STOC20–23, 2022. 18

[Mor94] Moshe Morgenstern. Existence and explicit constructions of $q + 1$ regular Ramanujan graphs for every prime power q. *J. Combin. Theory Ser. B*, 62(1):44–62, 1994. 18

[Nil91] A. Nilli. On the second eigenvalue of a graph. *Discrete Math.*, 91(2):207–210, 1991. 18

[Spi19] Daniel A. Spielman. *Spectral and Algebraic Graph Theory*. 2019. 8