

RAMANUJAN GRAPHS IN THE CONSTRUCTION OF LDPC CODES

WALTER H. CHEN

ABSTRACT. Low-density parity-check (LDPC) codes have recently become a popular interdisciplinary area of research. Widely unknown after their invention by Gallager in 1965, the existence of efficient encoding and decoding algorithms coupled with performance that operates near theoretical limits has led to the rediscovery of LDPC codes. This paper will address the reasoning and construction of LDPC codes with Ramanujan graphs. Most of the material here can be found in [1] and [5].

1. INTRODUCTION

Low Density Parity Check (LDPC) codes were invented in 1963 by Robert Gallager at MIT as part of his PhD thesis [2]. Although largely unregarded at its time of publication, in the last decade, LDPC codes have drawn significant attention of many researchers.

Interest has been rekindled because of the development of the computational power required to utilize LDPC codes to their fullest potential. Specifically, LDPC codes came equipped with an efficient decoding algorithm that has a natural parallel implementation.

1.1. Definitions.

1.1.1. *Low-Density Parity-Check Codes.* An LDPC code is a binary linear block code that is specified by a sparse $n \times m$ parity-check matrix, H . H can be considered to be part of the adjacency matrix for a bipartite graph (known as a Tanner graph) whose rows and columns represent the vertices of two disjoint vertex sets, one of size m and the other of size n , respectively.

Definition 1.1. A binary linear block code is a subspace of $\{0, 1\}^n$ whose code words are of fixed length.

Definition 1.2. A word $w \in \{0, 1\}^n$ is a codeword of a code C if

$$w \cdot H = \mathbf{0}$$

where H is the parity-check matrix specifying C .

Figure 1 is a Tanner graph representing a parity check matrix of an LDPC code. Taken with the above definition, the Tanner graph determines that $a \oplus c \oplus e = 0$ (where \oplus is addition over \mathbb{F}_2) where nodes a , c , and e have the values of the corresponding entries in the codeword w . Accordingly the n right vertices are called the constraints, and the m left vertices are called the variables.

Thanks to Goong Chen, Winnie Li, and Dan Zaffran. The work was done at the Penn State MASS program.

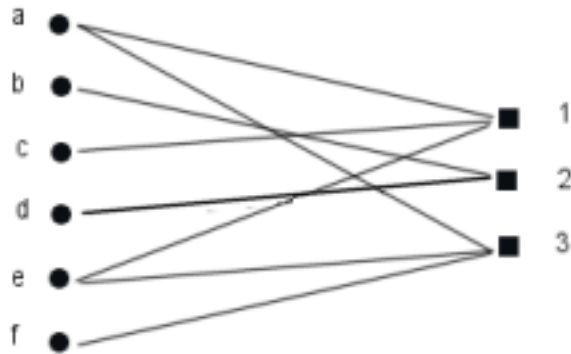


FIGURE 1. The underlying Tanner graph of an LDPC code

Definition 1.3. A (j, k) – regular code has its parity-check matrix H having exactly j non-zero entries in its columns and exactly k non-zero entries in its rows. As H is sparse, $j \ll m, k \ll n$. There are regular and irregular LDPC codes, but we will only consider regular LDPC codes.

Example 1.4. One reason that LDPC codes are of interest because of fast sequential decoding algorithms. Here is a simple example (Sipser and Spielman [6]):

- (1) If there is a variable with more violated constraints than satisfied constraints, complement the value of the variable
- (2) Repeat
- (3) Terminate when no such variable remains

Each iteration of this algorithm would resolve one variable that had more violated constraints than satisfied constraints. However, a natural optimization would be to have all such variables resolved in parallel. This is a tremendous boost to the speed of decoding. Modern computers are able to utilize this configuration, making the implementation of LDPC codes particularly advantageous in similar (more realistic) decoding algorithms.

1.1.2. *Ramanujan Graphs.* So, now that we know that LDPC codes are an efficient choice for our code, the question arises: how do we construct good LDPC codes? From the above discussion, we can see that the problem can be viewed from the perspective of the underlying graph. What graph structure will result in good LDPC codes? One answer is – Ramanujan graphs.

Definition 1.5. A finite, connected, k -regular graph G is a Ramanujan graph if $\mu_1 \leq 2\sqrt{k-1}$, where μ_1 is the largest non-trivial eigenvalue of the adjacency matrix representing G .

There are many ways to construct Ramanujan graphs. The construction presented in this paper is due to Lubotzky, Phillips, and Sarnak [3].

The nodes of their graph are elements of PGL_2 . They construct this by a rather in depth (see [1]) procedure of commutation, starting with the quaternions, and taking homomorphisms and quotients until they reach PGL_2 . In this sense, it is no surprise that there are many properties of these graphs relating to the Legendre symbol.

Definition 1.6. The Legendre symbol $\left(\frac{p}{q}\right)$ is defined as follows:

$$\left(\frac{p}{q}\right) = \begin{cases} 0 & \text{if } p \text{ divides } m \\ 1 & \text{if } p \text{ does not divide } m \text{ and } m \text{ is a square modulo } p \\ -1 & \text{if } p \text{ does not divide } m \text{ and } m \text{ is not a square modulo } p \end{cases}$$

where $p, q \in \mathbb{Z}$

Remark 1.7. A property we will use results from the fact that the group whose elements are the squares in \mathbb{F}_q^x (ie, those $m \in \mathbb{F}_q^x$ whose Legendre symbol is 1) has index 2 in \mathbb{F}_q^x :

$$\left(\frac{x}{q}\right)\left(\frac{y}{q}\right) = \left(\frac{xy}{q}\right)$$

The next section will focus on why the properties resulting from Ramanujan criterion are valuable to LDPC codes.

1.2. Motivation. Why would we like to construct LDPC codes with Ramanujan graphs? Gallager's early work focused around randomly constructed LDPC codes. However, explicitly constructed codes have the benefit of having provable properties that allow us to better understand its performance.

So now that we are looking at explicit LDPC codes, why use Ramanujan graphs?

First, we must know what properties of the underlying Tanner graph make for a good LDPC code.

- Criterion 1: The graph is a good expander.

Definition 1.8. A graph $G = (V, E)$ is called an ϵ -expander if for any $S \subset V$ with $|S| \leq |V|/2$, $|\partial S| \geq \epsilon|S|$, where $\partial S = \{v \in V \setminus S : (v, s) \in E \text{ for some } s \in S\}$. Namely,

$$h(G) \equiv \min \left\{ \frac{|\partial S|}{|S|} : S \subseteq V, |S| \leq |V|/2 \right\} \geq \epsilon$$

So a graph is a "good expander" if ϵ is a "large expansion factor." Sipster and Spielman show that asymptotically good LDPC codes require good expansion. This is because a graph with a large $h(G)$ value can be considered to be one that disseminates information efficiently. As the graphs we are considering are bipartite, this implies that in the iterative sum-product decoding process similar to the one given in Example 1.4, a large number of breached constraints will arise from a small subset of erroneous bits. Many of these constraints will consequently be able to correct their adjacent bits as the algorithm iterates.

In general, $h(G)$ is difficult to compute, but the following lower bound allows us to reformulate the problem of maximizing $h(G)$:

Theorem 1.9. A finite, connected, k -regular graph without loops has

$$\frac{k - \mu_1}{2} \leq h(G)$$

where μ_1 is the largest non-trivial eigenvalue of G , and $k - \mu_1$ is called the spectral gap.

Proof. See [1]. □

The following theorem gives an asymptotic lower bound on the spectral gap.

Theorem 1.10. *Let $(X_m)_{m \geq 1}$ be a family of connected, k -regular, finite graphs, with $|V_m| \rightarrow +\infty$ as $m \rightarrow +\infty$. Then,*

$$\liminf_{m \rightarrow +\infty} \mu_1(X_m) \geq 2\sqrt{k-1}$$

Proof. See [1]. □

Definition 1.5 taken with Theorem 1.10 shows that Ramanujan graphs have the largest asymptotic spectral gaps, and subsequently, are the best asymptotic expanders. So, we will be exploiting these properties to construct good LDPC codes.

- Criterion 2: The girth is as large as possible.

The performance of these iterative decoding algorithms improves as the girth grows. Particularly, let us consider belief-propagation algorithms. These iterative algorithms calculate probabilities at the nodes, and then send a message containing this information to all of its incident nodes. This happens each iteration. The receiving node then uses these probabilities for subsequent calculations. One of the assumptions underlying this procedure is that all probabilities are independent.

Unfortunately, cycles in a graph introduces dependence among nodes. Accordingly, graphs of small girth suffer in their decoding efficiency because those graphs have a higher level of dependence among nodes. So, maximization of the girth is desirable, as it maximizes the performance of the decoding algorithm. Randomly generated LDPC codes rely on the sparsity of the parity-check matrix to avoid cycles, but our explicit algebraic constructions will allow us to talk about the structure of our codes. Namely, we can provide lower bounds on the length of the girth.

Theorem 1.11. *For the graph $X^{p,q}$ where $X^{p,q}$ is a Ramanujan graph to be described later, and $\binom{p}{q} = -1$,*

$$g(X^{p,q}) \geq 4 \log_p q - \log_p 4$$

Proof. See [3]. □

There was also suspected to be a connection between the Ramanujan criterion and graphs of large girth, but that was disproved. Nevertheless, there are many Ramanujan graphs with large girth, which makes the construction of LDPC codes based on these graphs to be a well-founded decision.

As Ramanujan graphs satisfy these two criteria for good LDPC codes, we will use them for their construction.

1.3. Code Construction. This section requires background that can be developed by referencing [1] and [5].

As mentioned before, the Ramanujan graph construction we are using is due to Lubotzky, Phillips, and Sarnak [3]. Particularly, our graph is a Cayley graph, $X(PGL_2(F_q), S_{p,q})$. This satisfies the Theorem 1.11; thus, it has large girth. We will see in the following theorem that it also satisfies the Ramanujan condition, and has other properties that make it particularly useful for LDPC codes (namely bipartiteness and regularity).

Theorem 1.12. *The Cayley graph $X(PGL_2(F_q), S_{p,q})$ (denoted $X^{p,q}$) where $\binom{p}{q} = -1$ is a $p+1$ regular, bipartite, Ramanujan graph.*

Proof. Define:

$$\varphi: PGL_2(\mathbb{F}_q) \longrightarrow \{-1, 1\}, AD \longmapsto \left(\frac{\det(A)}{q} \right)$$

where $A \in GL_2(\mathbb{F}_q)$ and \mathcal{D} are the diagonal matrices over \mathbb{F}_q .

Suppose $AD = BD$.

That implies $AD_1 = BD_2$ where D_1 and D_2 are diagonal matrices

$$\begin{aligned} \Rightarrow A &= BD_2D_1^{-1} \\ \Rightarrow \det(A) &= \det(BD_2D_1^{-1}) \\ \Rightarrow \det(A) &= \det(B) \det(D_2) \det(D_1^{-1}) \\ \Rightarrow \det(A) &= \det(B)x^2y^2 \text{ where } x, y \in \mathbb{F}_q \\ \Rightarrow \left(\frac{\det(A)}{q} \right) &= \left(\frac{\det(B)x^2y^2}{q} \right) \\ \Rightarrow \left(\frac{\det(A)}{q} \right) &= \left(\frac{\det(B)}{q} \right) \left(\frac{x^2}{q} \right) \left(\frac{y^2}{q} \right) \\ \Rightarrow \left(\frac{\det(A)}{q} \right) &= \left(\frac{\det(B)}{q} \right) \\ \Rightarrow \varphi(AD) &= \varphi(BD) \end{aligned}$$

So φ is well-defined. Using the properties exploited in the above proof, specifically that $\det(AB) = \det(A)\det(B)$ and $\left(\frac{x}{q}\right)\left(\frac{y}{q}\right) = \left(\frac{xy}{q}\right)$, it is clear that φ is a homomorphism. This defines a bipartition on $X^{p,q}$.

The remaining proofs of regularity and the Ramanujan condition can be found in [1]. □

Remark 1.13. Conveniently, as a consequence, $\varphi^{-1}(1) = PSL_2(\mathbb{F}_q)$. Which means that $PSL_2(\mathbb{F}_q)$ is one half of the graph, one of the two disjoint vertex sets.

We demonstrate the construction of a (3,6)-regular code due to Rosenthal and Votobel:

Let us construct the LDPC code using $X^{p,q}$, the Cayley graph $X(PGL_2(\mathbb{F}_q), S_{p,q})$, with $p = 5, q = 17$.

Using Theorem 1.12, $\left(\frac{p}{q}\right) = -1$ gives that the graph is bipartite, so it has the natural structure to use for LDPC codes. However, Theorem 1.12 also gives that the graph is $p + 1$ regular, but LDPC codes must have imbalanced bipartite graphs with different regularities for the variable and constraint nodes. In $X^{5,17}$, each node is 6 regular. By taking two copies of one of the vertex sets, we can create a (3, 6) – regular LDPC code.

So, as “left vertices” we take two copies of $PSL_2(\mathbb{F}_{17})$ and as “right vertices” we take $PGL_2(\mathbb{F}_{17}) \setminus PSL_2(\mathbb{F}_{17})$ as given by Remark 1.13.

Each element of $S_{p,q}$ has its multiplicative inverse. As $|S_{p,q}|$ has six elements, let them be denoted by $A, A^{-1}, B, B^{-1},$ and C, C^{-1} . We then construct the graph as shown in Figure 2.

The resulting code is a (3, 6) – regular LDPC code having block length $|PGL_2(\mathbb{F}_q)| = q(q^2 - 1)$ which with $q = 17$ gives codewords of length 4896. As the construction of

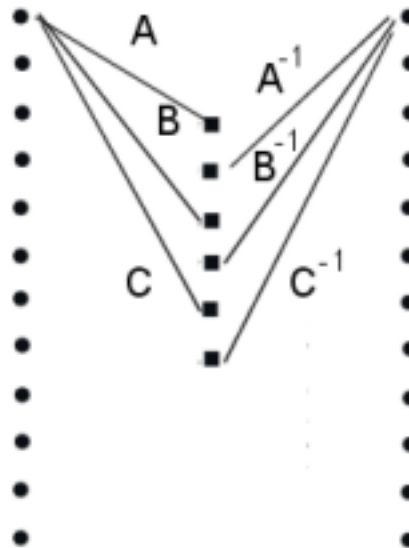


FIGURE 2. The left and right nodes are the two copies of $PSL_2(\mathbb{F}_q)$, the left nodes connecting via edges A, B, C, and the right nodes connecting via those inverses

the code largely maintains the original graph's structure, the resulting code maintains the criteria for large girth. The girth of this graph was calculated to be 12 [5].

1.4. Conclusions. Through simulations, this construction was shown by Rosenthal and Vontobel to perform better than randomly constructed LDPC codes of equal regularity and codeword length. However, [4] shows that this construction results in the weakness of having low-weight codewords that result in decoding errors. Further work on algebraic constructions are necessary.

REFERENCES

1. G. Davidoff, P. Sarnak, and A. Valette. *Elementary Number Theory, Group Theory, and Ramanujan Graphs*. Cambridge University Press, 2003.
2. R. G. Gallager. *Low-Density Parity-Check Codes*. MIT Press, 1963.
3. A. Lubotzky, R. Phillips, and P. Sarnak. *Ramanujan Graphs*. *Combinatorica*, vol. 8, no. 3, pp. 261–277, 1988.
4. D. MacKay and M. Postol. *Weaknesses of Margulis and Ramanujan-Margulis Low-Density Parity-Check Codes*. *Electronic Notes in Theoretical Computer Science*, vol. 74, www.elsevier.nl/locate/entcs/volume74.html, 2003.
5. J. Rosenthal and P. O. Vontobel. *Constructions of LDPC Codes using Ramanujan Graphs and Ideas from Margulis*. *Proceedings of the 38th Annual Allerton Conference on Communication, Control, and Computing*, pp. 248–257, 2000.
6. M. Sipser and D. A. Spielman. *Expander codes*. *IEEE Trans. Inform. Theory*, 42(6, part 1):1710–1722, 1996.
7. R. M. Tanner. *A recursive approach to low complexity codes*. *IEEE Trans. Inform. Theory*, 27(5):533–547, 1981.

CORNELL UNIVERSITY, ITHACA, NY, 14853
E-mail address: whc7@cornell.edu