Advances in Mathematics of Communications
Volume X, No. 0X, 200X, X–XX

Web site: http://www.aimSciences.org

EIGENVALUE BOUNDS ON THE PSEUDOCODEWORD WEIGHT OF EXPANDER CODES

Christine A. Kelley
Department of Mathematics
The Ohio State University
Columbus, OH 43210, USA.

Deepak Sridhara
Seagate Technology
1251 Waterfront Place
Pittsburgh, PA 15222, USA.

(Communicated by Aim Sciences)

Abstract. Four different ways of obtaining low-density parity-check codes from expander graphs are considered. For each case, lower bounds on the minimum stopping set size and the minimum pseudocodeword weight of expander (LDPC) codes are derived. These bounds are compared with the known eigenvalue-based lower bounds on the minimum distance of expander codes. Furthermore, Tanner's parity-oriented eigenvalue lower bound on the minimum distance is generalized to yield a new lower bound on the minimum pseudocodeword weight. These bounds are useful in predicting the performance of LDPC codes under graph-based iterative decoding and linear programming decoding.

1. Introduction

Expander graphs are of fundamental interest in mathematics and engineering and have several applications in computer science, complexity theory, derandomization, designing communication networks, and coding theory [1, 2]. A family of highly expanding graphs known as Ramanujan graphs [3, 4] was constructed with excellent graph properties that surpassed the parameters predicted for random graphs. The description of these graphs and their analysis rely on deep results from mathematics using tools from graph theory, number theory, and representation theory of groups [5]. Other authors have investigated non-algebraic approaches to designing expander graphs and one such construction takes an appropriately defined product of small component expander graphs to construct a larger expander graph [1, 6, 7]. Moreover, expander graphs have a special appeal from a geometric viewpoint. Isoperimetric problems in geometry have also been described by analogous problems in graphs, and a close connection exists between the Cheeger constant, defined for Riemannian surfaces, and the expansion constant in graphs. Expander graphs can be viewed as discrete analogues of Riemannian manifolds.

In this paper, we focus on one prominent application of expander graphs – namely, the design of low-density parity-check (LDPC) codes. Low-density parity-check codes are a class of codes that can be represented on sparse graphs and have

2000 *Mathematics Subject Classification:* Primary: 58F15, 58F17; Secondary: 53C35.
Key words and phrases: Expander graphs, LDPC codes, pseudocodewords, pseudocodeword weight, iterative decoding.

The first author is with the Department of Mathematics at The Ohio State University. She was previously with the Fields Institute, Toronto, Canada. The second author is with Seagate Technology, Pittsburgh, USA. He was previously with the Institut für Mathematik, Universität Zürich, Switzerland.

been shown to achieve record breaking performances with graph-based message-passing decoders. Graphs with good expansion properties are particularly suited for the decoder in dispersing messages to all nodes in the graph as quickly as possible. Expander codes are families of graph-based codes where the underlying graphs are expanders. That is, every element of the family is an expander and gives rise to an expander code. The codes are obtained by imposing code-constraints on the vertices (and possibly, edges) of the underlying expander graphs [8, 9, 10]. It has been observed that graphs with good expansion lead to LDPC codes with good minimum distance, a parameter that is fundamental to the error-correction capability of the code. In fact, one method of designing asymptotically good linear block codes is from expander graphs [8].

The popularity of LDPC codes is that they can be decoded with linear time complexity using graph-based message-passing decoders, thereby allowing for the use of large block length codes in several practical applications. In contrast, maximum-likelihood (ML) decoding a generic error-correcting code is known to be NP hard. A parameter that dominates the performance of a graph-based message passing decoder is the minimum pseudocodeword weight, in contrast to the minimum distance for an optimal (or, ML) decoder. The minimum pseudocodeword weight of the graph has been found to be reasonable predictor of the performance of a finite-length LDPC code under graph-based message-passing decoding and also linear programming decoding [11, 12, 13, 14, 15]. In this paper, we consider four different ways of obtaining LDPC codes from expander graphs. For each case, we first present the known lower bounds on the minimum distance of expander codes based on the expansion properties of the underlying expander graph. We then extend the results to lower bound the minimum stopping set size, which is essentially the minimum pseudocodeword weight on the binary erasure channel (BEC), and finally, we lower bound the minimum pseudocodeword weight on the binary symmetric channel (BSC). We also examine a new parity-oriented lower bound on the minimum pseudocodeword weight over the additive white Gaussian noise (AWGN) channel, thereby generalizing the result of Tanner [16] for the minimum distance.

2. Preliminaries

We introduce some preliminary definitions and notation that we will use in this paper.

Definition 1. A graph $G = (X, Y; E)$ is (c, d)-*regular bipartite* if the set of vertices in G can be partitioned into two disjoint sets X and Y such that all vertices in X have degree c and all vertices in Y have degree d and each edge $e \in E$ of G is incident with one vertex in X and one vertex in Y, i.e., $e = (x, y), x \in X, y \in Y$.

We will refer to the vertices of degree c as the *left* vertices, and to vertices of degree d as the *right* vertices.

The adjacency matrix of a d-regular connected graph has d as its largest eigenvalue. Informally, a graph is a good expander if the gap between the first and the second largest eigenvalues is as big as possible. More precise definitions will be given later in the paper as needed. Note that for a (c, d)-regular bipartite graph, the largest eigenvalue is $\sqrt{cd}$.

Definition 2. A *simple* LDPC code is defined by a bipartite graph G (also called, a Tanner graph) whose left vertices are called *variable* (or, *codebit*) nodes and whose right vertices are called *check* (or, *constraint*) nodes and the set of codewords are all

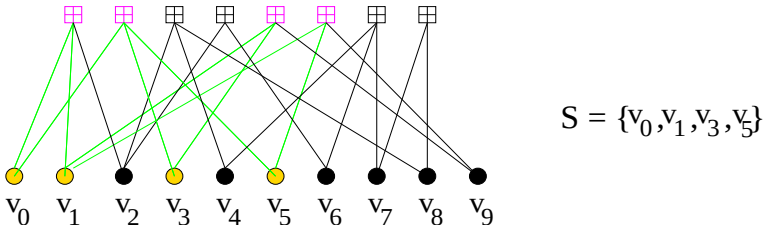


FIGURE 1. A stopping set $S = \{v_0, v_1, v_3, v_5\}$ in G.

binary assignments to the variable nodes such that at each check node, the modulo two sum of the variable node assignments connected to the check node is zero, i.e., the parity-check constraint involving the neighboring variable nodes is satisfied.

The above definition can be generalized by introducing more complex constraints instead of simple parity-check constraints at each constraint node, and the resulting LDPC code will be called a *generalized* LDPC code. Note that equivalently, the LDPC code can be described by a (binary) incidence matrix (or, parity-check matrix) wherein the rows of the matrix correspond to the constraint nodes of G and the columns correspond to variable nodes and there is a one in the matrix at a row-column entry whenever there is an edge between the corresponding constraint node and variable node in G.

To analyze the performance of graph-based message passing decoding, certain combinatorial objects of the LDPC constraint graph have been identified that control the performance of the decoder. When the channel is characterized by the binary erasure channel (BEC), it has been shown that stopping sets in the Tanner graph control the performance of the message-passing decoder.

Definition 3. [17] A *stopping set* is a subset set S of the variable nodes such that every constraint node that is a neighbor of some node $s \in S$ is connected to S at least twice.

Note that the above definition of a stopping set is for simple LDPC codes. The size of a stopping set S is equal to the number of elements in S. A stopping set is said to be *minimal* if there is no smaller sized stopping set contained within it. The smallest minimal stopping set is called a *minimum* stopping set, and its size is denoted by $s_{\min}$. Note that a minimum stopping set is not necessarily unique. Figure 1 shows a stopping set in the graph. Observe that $\{v_4, v_7, v_8\}$ and $\{v_3, v_5, v_9\}$ are two minimum stopping sets of size $s_{\min} = 3$, whereas $\{v_0, v_1, v_3, v_5\}$ is a minimal stopping set of size 4.

On the BEC, if all of the nodes of a stopping set are erased, then the graph-based iterative decoder will not be able to recover the erased symbols associated with the nodes of the stopping set [17]. Therefore, it is advantageous to design LDPC codes with large minimum stopping set size $s_{\min}$.

For other channels, it has been recently observed that so called *pseudocodewords* dominate the performance of the iterative decoder [11, 12]. (In fact, pseudocodewords are a generalization of stopping sets for other channels.) We will now introduce the formal definition of *lift-realizable* pseudocodewords of an LDPC constraint graph G [12]. However, we will also need to introduce the definition of a graph lift. A degree ℓ cover (or, lift) $\hat{G}$ of G is defined in the following manner:

Definition 4. A finite degree ℓ *cover* of $G = (V, W; E)$ is a bipartite graph $\hat{G}$ where for each vertex $x_i \in V \cup W$, there is a *cloud* $\hat{X}_i = \{\hat{x}_{i_1}, \hat{x}_{i_2}, \ldots, \hat{x}_{i_\ell}\}$ of vertices in $\hat{G}$, with $deg(\hat{x}_{i_j}) = deg(x_i)$ for all $1 \leq j \leq \ell$, and for every $(x_i, x_j) \in E$, there are ℓ edges from $\hat{X}_i$ to $\hat{X}_j$ in $\hat{G}$ connected in a $1-1$ manner.

Figure 2 shows a base graph G and a degree four cover of G.

Definition 5. Suppose that $\hat{\mathbf{c}} = (\hat{c}_{1,1}, \hat{c}_{1,2}, \ldots, \hat{c}_{1,\ell}, \hat{c}_{2,1}, \ldots, \hat{c}_{2,\ell}, \ldots)$ is a codeword in the Tanner graph $\hat{G}$ representing a degree ℓ lift of G. A *pseudocodeword* $\mathbf{p}$ *of* G is a vector $(p_1, p_2, \ldots, p_n)$ obtained by reducing a codeword $\hat{\mathbf{c}}$, of the code in the lift graph $\hat{G}$, in the following way:

$$\hat{\mathbf{c}} = (\hat{c}_{1,1}, \ldots, \hat{c}_{1,\ell}, \hat{c}_{2,1}, \ldots, \hat{c}_{2,\ell}, \ldots) \rightarrow (\tfrac{\hat{c}_{1,1}+\hat{c}_{1,2}+\cdots+\hat{c}_{1,\ell}}{\ell}, \tfrac{\hat{c}_{2,1}+\hat{c}_{2,2}+\cdots+\hat{c}_{2,\ell}}{\ell}, \ldots) = (p_1, p_2, \ldots, p_n) = \mathbf{p},$$

where $p_i = \frac{\hat{c}_{i,1}+\hat{c}_{i,2}+\cdots+\hat{c}_{i,\ell}}{\ell}$.

The vector $\hat{c}$ on the left hand side of Figure 2 corresponds to a codeword in the degree four cover that is also a codeword in the base graph G, whereas the vector on the right hand side corresponds to a codeword in the degree four cover that does not correspond to a codeword in the base graph.

From the above definition, it is easy to show that for a simple LDPC constraint graph G, a pseudocodeword $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ is a vector that satisfies the following set of inequalities:

$$0 \leq p_i \leq 1, \quad \text{for } i = 1, 2, \ldots, n. \tag{1}$$

and, if variable nodes i_1, i_2, $\ldots$, i_d participate in a check node of degree d, then the pseudocodeword components satisfy

$$p_{i_j} \leq \sum_{k=1,2,..d, k\neq j} p_{i_k}, \text{ for } j = 1, 2, .., d. \tag{2}$$

Extending the above for generalized LDPC codes, it can similarly be shown that on a generalized LDPC constraint graph G, a pseudocodeword $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ is a vector that satisfies the following set of inequalities:

$$0 \leq p_i \leq 1, \quad \text{for } i = 1, 2, \ldots, n. \tag{3}$$

and, if variable nodes i_1, i_2, $\ldots$, i_d participate in a constraint node of degree d and that constraint node represents a subcode $[d, rd, \epsilon d]$, then the pseudocodeword components satisfy

$$(d\epsilon - 1)p_{i_j} \leq \sum_{k=1,2,..d, k\neq j} p_{i_k}, \text{ for } j = 1, 2, .., d. \tag{4}$$

Remark 1. Note that Equation 4 implies that the pseudocodeword components of the generalized LDPC constraint graph G also satisfy the following set of inequalities at the degree d constraint node representing a $[d, rd, \epsilon d]$ subcode

$$\sum_{\text{any } \frac{d\epsilon}{2} j\text{'s}} p_{i_j} \leq \sum_{\text{remaining terms}} p_{i_k} \tag{5}$$

The set of lift-realizable pseudocodewords can also be described elegantly by means of a polytope, called the fundamental polytope [11]. In particular, lift-realizable pseudocodewords are dense in the fundamental polytope.

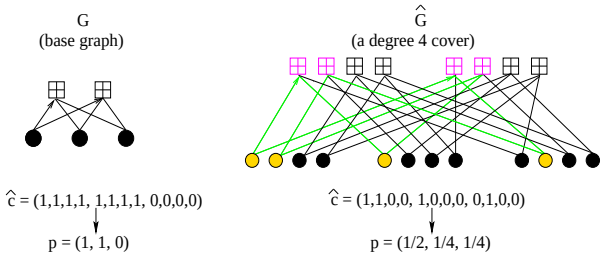


FIGURE 2. A pseudocodeword in the base graph (or a valid codeword in a lift).

It was shown in [11, 12] that a stopping set in a simple LDPC constraint graph is the support of a pseudocodeword as defined above. Thus, generalizing the definition of stopping sets to generalized LDPC code, we have:

Definition 6. A *stopping set* in a generalized LDPC constraint graph G is the support of a pseudocodeword $\mathbf{p}$ of G.

In Sections 3, 4, 5 and 6, we will consider pseudocodewords and their behavior on the binary symmetric channel (BSC), and in Section 7, we will consider pseudocodewords on the additive white Gaussian noise (AWGN) channel. The weight of a pseudocodeword $\mathbf{p}$ on the BSC is defined as follows [18].

Definition 7. Let e be the smallest number such that the sum of the e largest components of $\mathbf{p}$ is at least the sum of the remaining components of $\mathbf{p}$. Then, the *weight* of $\mathbf{p}$ is

$$w_{BSC}(\mathbf{p}) = \begin{cases} 2e, & \text{if } \sum_{\text{e largest}} p_i = \sum_{\text{remaining}} p_i \\ 2e-1, & \text{if } \sum_{\text{e largest}} p_i > \sum_{\text{remaining}} p_i \end{cases}$$

Definition 8. The *minimum pseudocodeword weight* of an LDPC constraint graph G on the BSC is the minimum weight among all pseudocodewords obtainable from all finite-degree lifts of G. This parameter is denoted by $w_{\min}^{BSC}$.

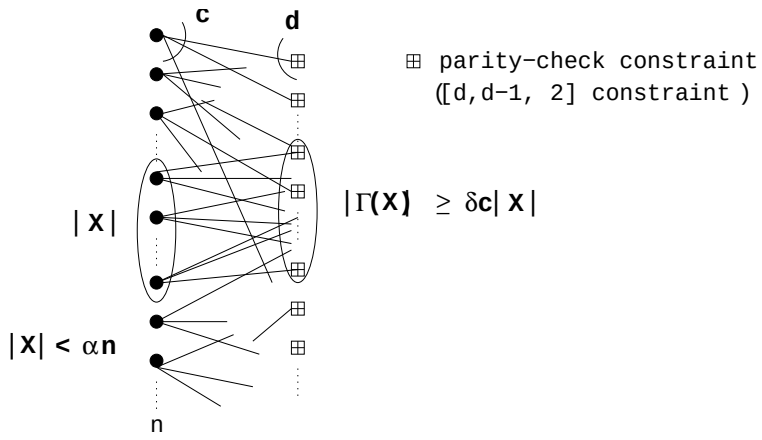
3. CASE A

Definition 9. Let $0 < \alpha < 1$ and $0 < \delta < 1$. A (c, d)-regular bipartite graph G with n degree c nodes on the left and m degree d nodes on the right is an $(\alpha n, \delta c)$ *expander* if for every subset U of degree c nodes such that $|U| < \alpha n$, the size of the set of neighbors of U, $|\Gamma(U)|$ is at least $\delta c|U|$.

Let a (c, d)-regular bipartite graph G with n left vertices and m right vertices be an $(\alpha n, \delta c)$ expander. An LDPC code is obtained from G by interpreting the degree c vertices in G as variable nodes and the degree d vertices as simple parity-check nodes. (See Figure 3.)

3.1. MINIMUM DISTANCE.

Lemma 1. [8] *If $\delta > 1/2$, the LDPC code obtained from the $(\alpha n, \delta c)$ expander graph G as above has minimum distance $d_{\min} \geq \alpha n$.*



Degree c vertices: variable nodes, degree d vertices: simple parity-check constraints.

FIGURE 3. Expander code: Case A.

3.2. MINIMUM STOPPING SET SIZE.

Lemma 2. *If $\delta > 1/2$, the LDPC code obtained from the $(\alpha n, \delta c)$ expander graph G as above has a minimum stopping set size $s_{\min} \geq \alpha n$.*

Proof. Suppose the contrary that there exists a stopping set S of size smaller than αn. Then by the expansion property of the graph, the size of the set of neighbors of S is $|\Gamma(S)| \geq \delta c|S|$. The average number of times a vertex in $\Gamma(S)$ is connected to the set S is $\frac{c|S|}{|\Gamma(S)|} \leq \frac{c|S|}{\delta c|S|} < 2$. This means that there is at least one vertex in $\Gamma(S)$ that is connected to the set S only once, contradicting the fact that S is a stopping set. □

Note that the above proof is just an extension of the proof of Lemma 1 for the lower bound on the minimum distance $d_{\min}$ since it uses the fact that every check node neighbor of a stopping set is connected to the set at least twice, which is a similar requirement for a codeword in the proof of Lemma 1.

3.3. MINIMUM PSEUDOCODEWORD WEIGHT.

Theorem 1. *If $\delta > 2/3 + 1/3c$ such that δc is an integer, the LDPC code obtained from the $(\alpha n, \delta c)$ expander graph G as above has a pseudocodeword weight*

$$w_{\min}^{BSC} > \frac{2(\alpha n - 1)(3\delta - 2)}{(2\delta - 1)} - 1.$$

Proof. Let $\mathbf{p} = (p_1, \ldots, p_n)$ be a pseudocodeword in G. Without loss of generality, let $p_1 \geq p_2 \geq \ldots \geq p_n$. Let $V = \{v_1, v_2, \ldots, v_n\}$ be the set of variable nodes. Let $U = \{v_1, \ldots, v_e\}$ be a set of e variable nodes corresponding to the e largest components of $\mathbf{p}$. Let $\dot{U} = \{v_i \in V | v_i \notin U, |\Gamma(v_i) \cap \Gamma(U)| \geq (1 - \lambda)c + 1\}$, where $\Gamma(X)$ is the set of neighbors of the vertices in X and $\lambda = 2(1 - \delta) + \frac{1}{c}$. Let $U' = U \cup \dot{U}$. Note that since we assume δc to be an integer, λc is also an integer.

We want to show that if $|U'| < \alpha n$, then we can find a set M of edges such that: (i) every node in U is incident with at least δc edges in M, (ii) every node in $\dot{U}$ is incident with at least λc edges in M, and (iii) every node in $\Gamma(U')$ is incident with at most one edge in M. (Such a set M is called a *δ-matching* in [13].) Suppose $e = |U| = \frac{(\alpha n - 1)}{(1+\beta)}$, where $\beta = \frac{(1-\delta)}{(3\delta - 2)}$. Then by Lemma 6 in [13], $|\dot{U}| \leq \beta |U|$. This implies that $|U'| \leq (1+\beta)|U| \leq (\alpha n - 1)$. Since G is an $(\alpha n, \delta c)$-expander, this means $|\Gamma(U')| \geq \delta c |U'| = \delta c |U| + \delta c |\dot{U}|$. Expand each vertex in U to δc copies, and expand each vertex in $\dot{U}$ to δc copies. However, we retain the set of nodes in $\Gamma(U')$ as is. (Note that the δc copies of a node in U correspond to δc edges incident on that node. Furthermore, since the graph does not contain multiple edges, each of those δc edges are connected to a distinct node in $\Gamma(U')$, which means each of the δc copies in the expanded set corresponding to a node in U are connected to a distinct node in $\Gamma(U')$.) Now, for any subset X of vertices from the expanded set of nodes in U and $\dot{U}$, we will always have that $|\Gamma(X)| \geq |X|$ since the graph G is an $(\alpha n, \delta c)$ expander and since $|U \cup \dot{U}| < \alpha n$. By Hall's (Marriage) Theorem, there is a matching of all nodes in expanded sets for U and $\dot{U}$ with the original set of neighbors $\Gamma(U')$. Since $\lambda c < \delta c$ by the choice of λ, this means that there is a δ-matching for the set U' as defined above.

Consider all of the check nodes in $\Gamma(U)$ that are incident with edges from M that are also incident with the vertices in U. We now apply the inequality in equation (2) at each of these check nodes and combine them. By considering a lift graph $\hat{G}$ of G wherein the pseudocodeword $\mathbf{p}$ forms a codeword $\hat{c}$, it can be inferred that the number of ones these vertices in U (in the graph G) contribute to the check nodes in $\Gamma(U)$ that are incident with M is at least $\delta c(p_1 + \cdots + p_e)$. These ones must be balanced by the remaining ones coming into these checks from the other nodes. This means at most $(1-\delta)c$ edges from each vertex in U are incident with these check nodes. Moreover, at most $(1-\lambda)c$ edges from each vertex in $\dot{U}$ are incident with these check nodes. This gives the following inequality

$$\delta c(p_1 + \cdots + p_e) \leq (1-\delta)c(p_1 + \cdots + p_e) + (1-\lambda)c\Big(\sum_{v_i \in \dot{U}} p_i\Big) + (1-\lambda)c\Big(\sum_{v_i \in V \setminus U'} p_i\Big). \quad (*)$$

The above inequality implies that

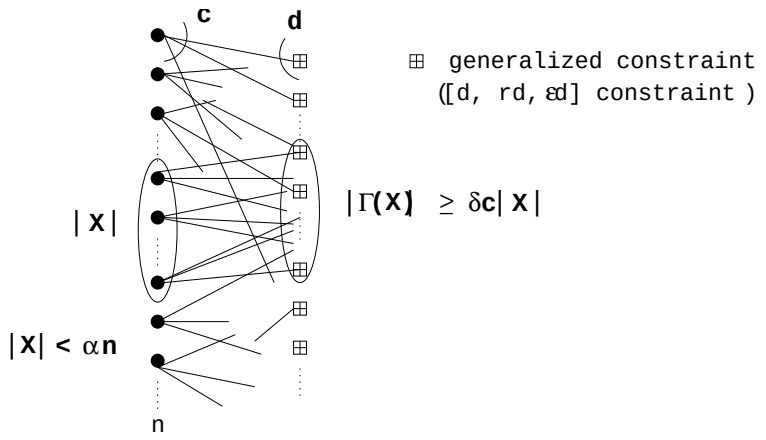
$$p_1 + \cdots + p_e \leq \frac{(1-\lambda)}{(2\delta - 1)}(p_{e+1} + \cdots + p_n) < p_{e+1} + \cdots + p_n,$$

from the choice of λ. From the definition of pseudocodeword weight on the BSC (Definition 7), we have $w^{BSC}(\mathbf{p}) > 2e - 1 = \frac{2(\alpha n - 1)(3\delta - 2)}{(2\delta - 1)} - 1$.

□

Remark 2.

- The proof of the above theorem can also be inferred directly by the result in [13]. However, we believe the proof presented here is somewhat simpler than the indirect approach in [13].
- For the case when $\delta = 3/4$, the lower bound on the minimum pseudocodeword weight $w_{\min}$ matches the lower bound on $d_{\min}$ and $s_{\min}$ presented in Lemmas 1 and 2. This is particularly appealing since an expander code achieving the lower bound on the minimum distance will also achieve the lower bound on



Degree c vertices: variable nodes, degree d vertices: sub-code constraints of a $[d, rd, \epsilon d]$ code.

FIGURE 4. Expander code: Case B.

the minimum pseudocodeword and will have no pseudocodewords of weight less than the minimum distance.

4. CASE B

Let a (c, d)-regular bipartite graph G with n left vertices and m right vertices be a $(\alpha n, \delta c)$ expander. (See Definition 9.) An LDPC code is obtained from G by interpreting the degree c vertices in G as variable nodes and the degree d vertices as sub-code constraints imposed by a $[d, rd, \epsilon d]$ linear block code[1]. A valid assignment of values to the variable nodes is one where the (binary) values assigned to the variable nodes connected to each constraint node satisfy all the constraints imposed by the subcode, meaning that the binary assignments from the variable nodes connected to each constraint node form a codeword in the subcode. (See Figure 4.) Such an LDPC code is called a *generalized LDPC code*.

4.1. MINIMUM DISTANCE.

Lemma 3. [8] *If $\delta > 1/(\epsilon d)$, the LDPC code obtained from the $(\alpha n, \delta c)$ expander graph G as above has minimum distance $d_{\min} \geq \alpha n$.*

4.2. MINIMUM STOPPING SET SIZE. A generalized stopping set is as defined in Definition 6 in Section 2. Under the assumption that the $[d, rd, \epsilon d]$ subcode has no idle components, meaning that there are no components that are zero in all of the codewords of the subcode, Definition 6 reduces to

Definition 10. A stopping set in a generalized LDPC code is a set of variable nodes such that every node that is a neighbor of some node $s \in S$ is connected to S at least ϵd times.

[1]The parameters of an $[n, k, d]$ binary linear block code are the block length n, the dimension k, and the minimum distance d.

Lemma 4. *If $\delta > 1/(\epsilon d)$, the LDPC code obtained from the $(\alpha n, \delta c)$ expander graph G as above has a minimum stopping set size $s_{\min} \geq \alpha n$.*

Proof. Suppose the contrary that there exists a stopping set S of size smaller than αn. Then by Definition 6, there is a pseudocodeword $\mathbf{p}$ whose support has a size smaller than αn. By the expansion property of the graph, the size of the set of neighbors of S is $|\Gamma(S)| \geq \delta c|S|$. The average number of times a vertex in $\Gamma(S)$ is connected to the set S is $\frac{c|S|}{|\Gamma(S)|} \leq \frac{c|S|}{\delta c|S|} < d\epsilon$. This means that there is at least one vertex in $\Gamma(S)$ that is connected to the set S less than $d\epsilon$ times. That means that there are less than $d\epsilon$ non-zero pseudocodeword components connected to that constraint node in $\Gamma(S)$. If we choose the $d\epsilon/2$ largest components among them, then their sum is greater than the sum of the remaining pseudocodeword components at that constraint node. This is a contradiction to the inequality in Equation 5, meaning $\mathbf{p}$ cannot be a pseudocodeword and therefore S cannot be a stopping set. Thus, the size of S cannot be less than αn. □

4.3. Minimum pseudocodeword weight.

Theorem 2. *If $\delta > \frac{2}{(\epsilon d+1)} + \frac{1}{c(\epsilon d+1)}$ such that δc is an integer, then the LDPC code obtained from the $(\alpha n, \delta c)$ expander graph G has a minimum pseudocodeword weight*

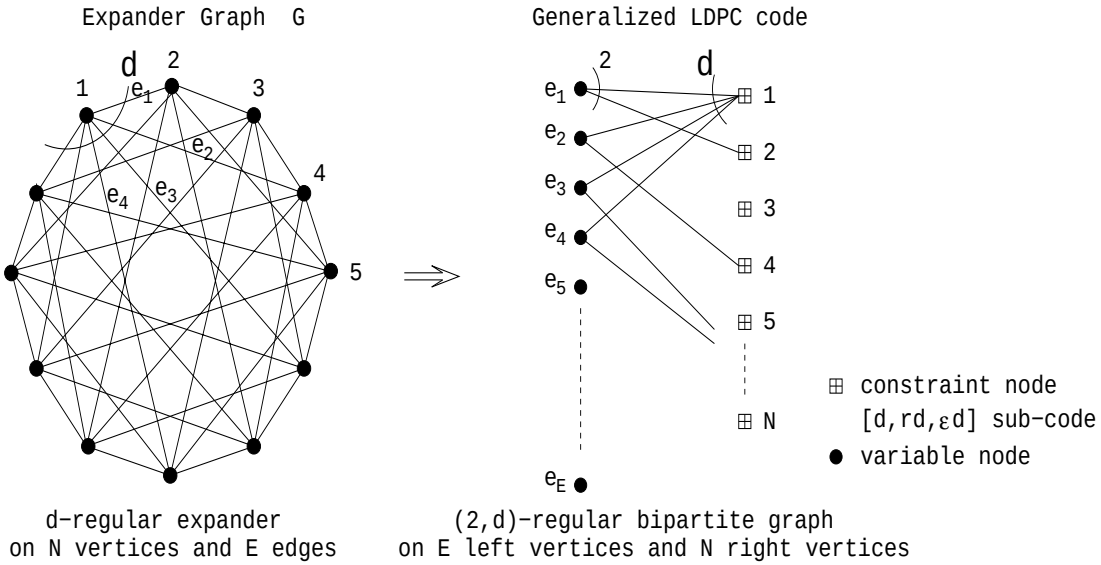
$$w_{\min}^{BSC} > \frac{2(\alpha n - 1)((d\epsilon + 1)\delta - 2)}{(d\epsilon\delta - 1)} - 1.$$

Proof. Suppose G is an $(\alpha n, \delta c)$-expander, where $\delta > \frac{2}{(d\epsilon+1)} + \frac{1}{(d\epsilon+1)c}$. Then, assuming $\mathbf{p}$ is a pseudocodeword of the LDPC constraint graph G, the proof follows that of Case A by choosing a set of variable nodes U corresponding to the e dominant components of the pseudocodeword and letting $|U| = e = \frac{(\alpha n-1)}{1+\beta}$, where $\beta = \frac{1-\delta}{(d\epsilon+1)\delta-2}$. We need to show that $w_{\min}^{BSC} > 2e - 1 = 2\frac{(\alpha n-1)((d\epsilon+1)\delta-2)}{(d\epsilon\delta-1)} - 1$. By using a strong subcode, the δ required is less than that in Case A, thereby allowing α to be larger and yielding a larger bound overall. The argument is the same as in the proof of Case A, where now we set $\lambda = 2 - d\epsilon\delta + \frac{1}{c}$.

Following the proof of Theorem 1, we will first show that if $|U| = e = \frac{(\alpha n-1)}{1+\beta}$, then $|\dot{U}| \leq \beta|U|$. Suppose to the contrary, $|\dot{U}| > \beta|U|$, then that means there is some subset $\ddot{U} \subset \dot{U}$ such that $|\ddot{U}| = \lfloor \beta|U| \rfloor + 1$. This means the size $|U \cup \ddot{U}| = |U| + \lfloor \beta|U| \rfloor + 1 \leq (1 + \beta)|U| + 1 = \alpha n$. Since G is an $(\alpha n, \delta c)$ expander, we then have $|\Gamma(U \cup \ddot{U})| \geq \delta c(|U| + |\ddot{U}|)$. However, observe that $|\Gamma(U \cup \ddot{U})| = |\Gamma(U)| + |\Gamma(\ddot{U}) \backslash \Gamma(U)| \leq c|U| + (\lambda c - 1)|\ddot{U}|$ since $|\Gamma(U)| \leq c|U|$ and $|\Gamma(\ddot{U}) \backslash \Gamma(U)| \leq (\lambda c - 1)|\ddot{U}|$ by definition. Combining the above inequalities, we have $\delta c(|U| + |\ddot{U}|) \leq c|U| + (\lambda c - 1)|\ddot{U}|$, implying $|\ddot{U}| \leq \frac{c(1-\delta)|U|}{c(\delta-\lambda)+1} = \beta|U|$. This contradicts the choice of $\ddot{U}$ above. Thus, if $|U| = e = \frac{(\alpha n-1)}{1+\beta}$, then $|\dot{U}| \leq \beta|U|$.

Following the rest of the proof of Theorem 1 and using the inequality in Equation 4 for the pseudocodeword components, the first inequality $(*)$ in the proof of Case A now becomes

$$\begin{aligned}(d\epsilon - 1)\delta c(p_1 + \cdots + p_e) \leq (1 - \delta)c(p_1 + \cdots + p_e) \\ +(1 - \lambda)c(\sum_{v_i \in \dot{U}} p_i) + (1 - \lambda)c(\sum_{v_i \in V \backslash U'} p_i).\end{aligned}$$



Edges: variable nodes, degree d vertices: constraints of a $[d, rd, \epsilon d]$ code.

FIGURE 5. Expander code: Case C.

This yields

$$p_1 + \cdots + p_e \leq \frac{(1-\lambda)}{(d\epsilon\delta - 1)}(p_{e+1} + \cdots + p_n) < p_{e+1} + \cdots + p_n$$

from the choice of λ. Thus, by Definition 7, the weight of $\mathbf{p}$ is $w(\mathbf{p}) > 2e - 1$. □

Remark 3.
- Since $d\epsilon \geq 2$ for any judicious choice of subcode, the lower bound in Theorem 2 is always greater than the lower bound in Theorem 1. Further, the graph need not be as good an expander in Case B as in Case A for the lower bound to hold. Thus, using strong subcodes is advantageous for constructing good LDPC codes from expander graphs.
- Note that for $\delta = \frac{3}{d\epsilon+2}$, the lower bound on the pseudocodeword weight equals the lower bound on the minimum distance and minimum stopping set size.

5. CASE C

Definition 11. A connected, simple, graph G is said to be a (n, d, μ) *expander* if G has n vertices, is d-regular, and the second largest eigenvalue of G (in absolute value) is μ.

Let a d-regular graph G be an (n, d, μ) expander. An LDPC code is obtained from G by interpreting the edges in G as variable nodes and the degree d vertices as constraint nodes imposing constraints of an $[d, rd, \epsilon d]$ linear block code. (See Figure 5.) The resulting LDPC code has block length $N = nd/2$ and rate $R \geq 2r-1$.

We now state a particularly useful result by Alon and Chung [19, 8] describing the expansion of a d-regular graph.

Lemma 5. *(Alon-Chung) Let G be a d-regular graph on n vertices and let μ be the second largest eigenvalue of its adjacency matrix. Then every subset S of γn vertices contains at most $\frac{nd}{2}(\gamma^2 + \frac{\mu}{d}(\gamma - \gamma^2))$ edges in the subgraph induced by S in G.*

5.1. MINIMUM DISTANCE.

Lemma 6. [8] *The LDPC code obtained from an (n, d, μ) expander graph G as above has minimum distance $d_{\min} \geq N\frac{(\epsilon - \frac{\mu}{d})^2}{(1-\frac{\mu}{d})^2}$.*

Note that the above result of Sipser and Spielman can be improved by a tighter bound in the last step of their proof in [8] to $d_{\min} \geq N\epsilon\frac{(\epsilon - \frac{\mu}{d})}{(1-\frac{\mu}{d})}$.

5.2. MINIMUM STOPPING SET SIZE.

Lemma 7. *The LDPC code obtained from an (n, d, μ) expander graph G has a minimum stopping set size $s_{\min} \geq N\epsilon\frac{(\epsilon - \frac{\mu}{d})}{(1-\frac{\mu}{d})}$.*

Note that we again use Definition 6 for stopping sets in G.

Proof. Let S be a subset of variable nodes (edges in G) of size $\frac{nd}{2}(\gamma^2 + \frac{\mu}{d}(\gamma - \gamma^2))$ representing a stopping set in G. Then S is the support of some pseudocodeword $\mathbf{p}$ in G. By the Alon-Chung lemma, the set S has at least γn constraint node neighbors $\Gamma(S)$. Since each edge in S has two constraint node neighbors in $\Gamma(S)$, this implies that the average number of edges in S connected to a constraint node in $\Gamma(S)$ is $\frac{2\frac{nd}{2}(\gamma^2 + \frac{\mu}{d}(\gamma - \gamma^2))}{\gamma n}$. However if

$$\frac{2\frac{nd}{2}(\gamma^2 + \frac{\mu}{d}(\gamma - \gamma^2))}{\gamma n} = d(\gamma + \frac{\mu}{d}(1 - \gamma)) < \epsilon d, \quad (**)$$

then there is at least one node in $\Gamma(S)$ that is connected to S fewer than ϵd times to S. That means that fewer than ϵd non-zero components of $\mathbf{p}$ are connected to a constraint node. It can now be shown that the inequality in Equation 5 is violated, implying that $\mathbf{p}$ cannot be a pseudocodeword (and, S is not a stopping set.)

The above inequality $(**)$ holds for $\gamma < \frac{\epsilon - \frac{\mu}{d}}{1-\frac{\mu}{d}}$. Substituting the value of γ in $|S| = \frac{nd}{2}(\gamma^2 + \frac{\mu}{d}(\gamma - \gamma^2))$, we infer that the graph cannot contain a stopping set of size less than $\frac{nd}{2}\epsilon\frac{(\epsilon - \frac{\mu}{d})}{(1-\frac{\mu}{d})}$. Hence,

$$s_{\min} \geq \frac{nd}{2}\epsilon\frac{(\epsilon - \frac{\mu}{d})}{(1 - \frac{\mu}{d})} = N\epsilon\frac{(\epsilon - \frac{\mu}{d})}{(1 - \frac{\mu}{d})}.$$

□

5.3. MINIMUM PSEUDOCODEWORD WEIGHT.

Theorem 3. *The LDPC code obtained from an (n, d, μ) expander graph G has a minimum pseudocodeword weight*

$$w_{\min}^{BSC} \geq N\epsilon\frac{(\frac{\epsilon}{2} - \frac{\mu}{d})}{(1 - \frac{\mu}{d})}.$$

Proof. The d-regular graph G can be transformed to a $(2,d)$-regular bipartite graph G' by representing every edge in G by a vertex in G' and every vertex in G by a vertex in G' and connecting the edge nodes to the vertex nodes in G' in a natural way. The edge nodes have degree two and they represent variable nodes of the LDPC code C, whereas the vertex nodes have degree d and each represents a $[d, rd, \epsilon d]$-subcode constraints.

Let $\mathbf{p} = (p_1, p_2, \ldots, p_N)$ be a pseudocodeword, where $N = \frac{nd}{2}$ is the number of edges in G and also the length of the LDPC code. Without loss of generality, let us assume that $p_1 \geq p_2 \geq \cdots \geq p_N$. Let e be the smallest number such that $p_1 + p_2 \cdots + p_e > p_{e+1} + \cdots + p_N$. Let X_e be the set of edges in G that correspond to the support of the e largest components of $\mathbf{p}$, and let S be the set of vertices incident on X_e. Note that in the transformed graph G', X_e is a subset of the variable nodes, and $\Gamma(X_e) = S$.

Let $|X_e| = \frac{nd}{2}(\gamma^2 + \frac{\mu}{d}(\gamma - \gamma^2))$, where $\gamma \leq (\frac{\frac{\epsilon}{2} - \frac{\mu}{d}}{1 - \frac{\mu}{d}})$. Since G is an (n, d, μ) graph, we have $|\Gamma(X_e)| \geq \gamma n$. We now claim that there is a set of edges M in G' called an ϵ-matching such that (i) every vertex in X_e in the graph G' is incident with two edges from M and (ii) every vertex in $\Gamma(X_e)$ in the graph G' is incident with at most $d\epsilon/2$ edges from M.

Given the claim, we can apply the pseudocodeword inequality from equation 5 at each of the vertices in $\Gamma(X_e)$ that is incident with edges from M. Note that each of those inequalities will have the form

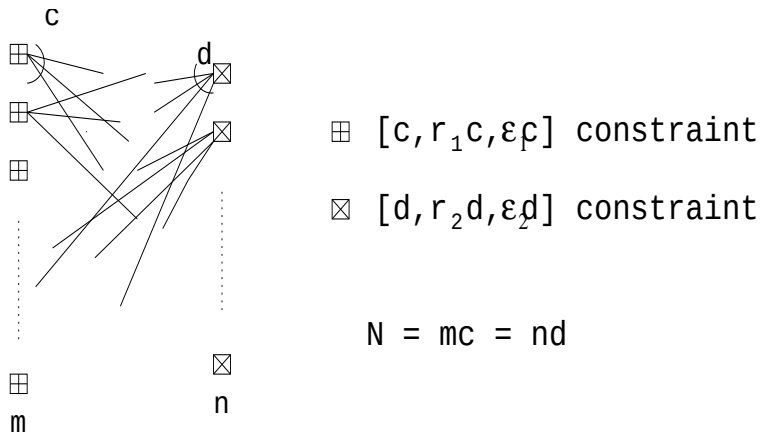
$$\sum_{\text{some } d\epsilon/2 \text{ or less terms from } X_e} p_i \leq \sum_{\text{terms not in } X_e} p_i.$$

Summing all such inequalities, we get

$$2 \sum_{i \in X_e} p_i \leq 2 \sum_{i \notin X_e} p_i.$$

By the definition of the pseudocodeword weight on the BSC channel (see Definition 7), we have that the pseudocodeword weight of $\mathbf{p}$ is $w(\mathbf{p}) \geq 2|X_e|$. Since $|X_e| = \frac{nd}{2}(\gamma^2 + \frac{\mu}{d}(\gamma - \gamma^2))$, for $\gamma \leq (\frac{\frac{\epsilon}{2} - \frac{\mu}{d}}{1 - \frac{\mu}{d}})$, we have $w(\mathbf{p}) \geq 2(\frac{nd}{2})(\frac{\epsilon}{2})(\frac{\frac{\epsilon}{2} - \frac{\mu}{d}}{1 - \frac{\mu}{d}}) = N\epsilon(\frac{\frac{\epsilon}{2} - \frac{\mu}{d}}{1 - \frac{\mu}{d}})$. This proves the desired lower bound on $w_{\min}$.

To prove the claim, observe that for any set X of left vertices in G' such that $|X| = N(\gamma^2 + \frac{\mu}{d}(\gamma - \gamma^2))$ where $\gamma \leq (\frac{\frac{\epsilon}{2} - \frac{\mu}{d}}{1 - \frac{\mu}{d}})$, we have $|\Gamma(X)| \geq \gamma n \geq \frac{4}{d\epsilon}|X|$. In other words, for every X such that $|X| = N(\gamma^2 + \frac{\mu}{d}(\gamma - \gamma^2))$ where $\gamma \leq (\frac{\frac{\epsilon}{2} - \frac{\mu}{d}}{1 - \frac{\mu}{d}})$, we have $\frac{d\epsilon}{2}|\Gamma(X)| \geq 2|X|$. To prove the ϵ-matching for the set X_e, make two copies of every vertex in X_e to denote the two edges in G' that are incident to each vertex in X_e. Similarly, make $\frac{d\epsilon}{2}$ copies of every vertex in $\Gamma(X_e)$ to denote the edges that are incident with each vertex in $\Gamma(X_e)$ that are also incident with vertices in X_e. Let $\hat{X}_e$ denote the expanded set of vertices of X_e and let $\hat{\Gamma}(X_e)$ denote the expanded set of $\Gamma(X_e)$. Form a new graph $\hat{G}'$ that contains vertices $\hat{X}_e$ and $\hat{\Gamma}(X_e)$. Form edges in $\hat{G}'$ in the following manner: if there is an edge from $u \in X_e$ to $v \in \Gamma(X_e)$, then there is an edge from exactly one of the copies of u in $\hat{X}_e$ to exactly one of the copies of v in $\hat{\Gamma}(X_e)$ and vice versa in $\hat{G}'$. This is always possible since $\frac{d\epsilon}{2}|\Gamma(X)| \geq 2|X|$ for any $X \subseteq X_e$ by the above observation. In the new graph $\hat{G}'$, it is easy to see that for any subset $\hat{X}$ of left vertices, we have the size of its neighboring-set $|\Gamma(\hat{X})| \geq |\hat{X}|$



Edges: variable nodes, degree c vertices: $[c, r_1 c, \epsilon_1 c]$ constraints, degree d vertices: $[d, r_2 d, \epsilon_2 d]$ constraints.

FIGURE 6. Expander code: Case D.

by the above observation (and by construction). Thus, by Hall's marriage theorem, there is a matching from the vertices in $\hat{X}_e$ to the vertices in $\hat{\Gamma}(X_e)$. This matching reduces to an ϵ-matching as defined above for the set X_e in G', thereby proving the claim.

□

Remark 4.
- Note that the lower bound on the minimum pseudocodeword weight is almost equal to the lower bound on the minimum distance and the minimum stopping set size.
- The lower bound suggests that if one were to use good expanding graphs such as the Ramanujan graphs from the construction in [3] and choose an appropriate choice of subcodes having minimum distance at least twice the second eigenvalue of the expander then the resulting code will have a good pseudocodeword weight and a good minimum distance. This is interesting for designing codes that are good for iterative decoding or LP decoding.

6. Case D

Definition 12. A (c, d)-regular bipartite graph G on m left vertices and n right vertices is a (c, d, m, n, μ) *expander* if the second largest eigenvalue of G (in absolute value) is μ.

Let a (c, d)-regular bipartite graph G be an (c, d, m, n, μ) expander. An LDPC code is obtained from G by interpreting the edges in G as variable nodes, the degree c left vertices as sub-code constraints imposed by an $[c, r_1 c, \epsilon_1 c]$ linear block code, and the degree d vertices as constraint nodes imposing constraints of an $[d, r_2 d, \epsilon_2 d]$ linear block code. (See Figure 6.) The resulting LDPC code has block length $N = mc = nd$ and rate $R \geq r_1 + r_2 - 1$.

We state a useful result by Janwa and Lal [10] describing the edge-expansion of a regular bipartite graph G.

Lemma 8. *(Janwa-Lal, edge-expansion) Let G be a (c,d)-regular bipartite graph on m vertices on the left and n vertices on the right and let μ be its second largest eigenvalue. If S and T are two subsets of the left and the right vertices, respectively, of G, then the number of edges in the induced sub-graph of S and T in G is at most*

$$|E(S,T)| \leq \frac{d}{m}|S||T| + \frac{\mu}{2}(|S| + |T|).$$

6.1. MINIMUM DISTANCE.

Lemma 9. [10] *If $\epsilon_2 d \geq \epsilon_1 c > \mu/2$, the LDPC code obtained from the (c,d,m,n,μ) expander graph G as above has minimum distance*

$$d_{\min} \geq N\left(\epsilon_1\epsilon_2 - \frac{\mu}{2\sqrt{cd}}(\epsilon_1\sqrt{\frac{c}{d}} + \epsilon_2\sqrt{\frac{d}{c}})\right).$$

6.2. MINIMUM STOPPING SET SIZE. We again use the generalized definition of stopping set in Definition 6. Under the assumption that the $[d, r_2 d, \epsilon_2 d]$ and $[c, r_1 c, \epsilon_1 c]$ subcodes have no idle components, meaning that there are no components that are zero in all of the codewords of either of the subcodes, Definition 6 reduces to the following:

Definition 13. A stopping set in a generalized LDPC code as in Case D is a set of variable nodes such that every node that is a degree c neighbor of some node $s \in S$ is connected to S at least $\epsilon_1 c$ times and every node that is a degree d neighbor of some node $s \in S$ is connected to S at least $\epsilon_2 d$ times.

Lemma 10. *If $\epsilon_2 d \geq \epsilon_1 c > \mu/2$, the LDPC code obtained from the (c,d,m,n,μ) expander graph G has a minimum stopping set size*

$$s_{\min} \geq N\left(\epsilon_1\epsilon_2 - \frac{\mu}{2\sqrt{cd}}(\epsilon_1\sqrt{\frac{c}{d}} + \epsilon_2\sqrt{\frac{d}{c}})\right).$$

Proof. Let X be a stopping set corresponding to a subset of edges in G and let S and T be the set of left and right neighbors, respectively, of X in G. Then X is the support of some pseudocodeword $\mathbf{p}$ in G. Suppose there is some node in S that is connected fewer that $c\epsilon_1$ times to the edges in X, then the inequality in Equation 5 is violated by the pseudocodeword components at that constraint node. Similarly, if some node in T is connected fewer than $d\epsilon_2$ times to the edges in X, then the corresponding pseudocodeword components will not satisfy all the inequalities in Equation 5. Thus, every node in S is connected to X at least $c\epsilon_1$ times and every node in T is connected to X at least $d\epsilon_2$ times. This means $|S| \leq \frac{|X|}{c\epsilon_1}$ and $|T| \leq \frac{|X|}{d\epsilon_2}$. By Lemma 8, we have

$$|X| \leq |E(S,T)| \leq \frac{d}{m}|S||T| + \frac{\mu}{2}(|S| + |T|).$$

This can be further bounded as

$$|X| \leq \frac{d}{m}|S||T| + \frac{\mu}{2}(|S| + |T|) \leq \frac{d}{m}\frac{|X|^2}{cd\epsilon_1\epsilon_2} + \frac{\mu}{2}(\frac{1}{c\epsilon_1} + \frac{1}{d\epsilon_2})|X|.$$

Simplifying, we obtain

$$|X| \geq mc(\epsilon_1\epsilon_2 - \frac{\mu}{2cd}(\epsilon_1 c + \epsilon_2 d)) = N(\epsilon_1\epsilon_2 - \frac{\mu}{2\sqrt{cd}}(\epsilon_1\sqrt{\frac{c}{d}} + \epsilon_2\sqrt{\frac{d}{c}})).$$

□

6.3. Minimum pseudocodeword weight.

Theorem 4. *If $\epsilon_2 d \geq \epsilon_1 c > 2\mu$, the LDPC code obtained from the (c, d, m, n, μ) expander graph G has a minimum pseudocodeword weight*

$$w_{\min}^{BSC} \geq N\frac{c}{d}\epsilon_1(\frac{\epsilon_1}{2} - \frac{\mu}{c}).$$

Proof. Let $\mathbf{p} = (p_1, p_2, \ldots, p_N)$ be a pseudocodeword. Without loss of generality, let us assume that $p_1 \geq p_2 \geq \cdots \geq p_N$. Let e be the smallest number such that

$$p_1 + p_2 \cdots + p_e \geq p_{e+1} + \cdots + p_N \quad (***).$$

Let X_e be the set of edges in G that correspond to the support of the e largest components of $\mathbf{p}$. Now we define a set S as the set of left neighbors (degree c neighbors) to the edges in X_e, and similarly define a set T as the set of right neighbors (degree d neighbors) to X_e. The (c, d)-regular graph G can be transformed to a graph G' by representing every edge in G by a vertex (called a left-vertex) in G', every vertex of degree c in G by a vertex (called a right-left vertex) in G', every vertex of degree d in G by a vertex (called a right-right vertex) in G' and by connecting the edges from the left vertices to the right-left and right-right vertices in G' in a natural way. The left vertices have degree two and they represent variable nodes of the LDPC code C, whereas the right-left vertices have degree c and represent $[c, r_1 c, \epsilon_1 c]$-subcode constraints and the right-right vertices have degree d and represent $[d, r_2 d, \epsilon_2 d]$-subcode constraints. Note that $\Gamma(X_e) = S \cup T$ in G'.

Let $|X_e| = N\frac{c}{2d}\epsilon_1(\frac{\epsilon_1}{2} - \frac{\mu}{c})$. Now let us consider two cases.

Case 1: Suppose $|S \cup T| = |S| + |T| \leq \frac{4}{c\epsilon_1}|X_e|$. Then, Since G is a (c, d, m, n, μ) graph, we have

$$|X_e| \leq \frac{d}{m}|S||T| + \frac{\mu}{2}(|S| + |T|)$$

Note that $|S||T| \leq \frac{(|S|+|T|)^2}{4}$. Hence, we have

$$|X_e| \leq \frac{d}{m}\frac{16|X_e|^2}{4c^2\epsilon_1^2} + \frac{\mu}{2}\frac{4|X_e|}{c\epsilon_1}$$

On simplifying, the above yields

$$|X_e| \geq N\frac{c}{2d}\epsilon_1(\frac{\epsilon_1}{2} - \frac{\mu}{c}).$$

This inequality does not contradict the assumption on the size of X_e.

Case 2: Suppose $|S \cup T| > \frac{4}{c\epsilon_1}|X_e|$. Then we claim that there is a set of edges M in G' called an ϵ_1-matching such that (i) every vertex in X_e in the graph G' is incident with two edges from M and (ii) every vertex in $S \cup T$ in the graph G' is incident with at most $c\epsilon_1/2$ edges from M.

Given the claim, we can apply the pseudocodeword inequality from equation 5 at each of the vertices in $S \cup T$ that is incident with edges from M. Note that each

of those inequalities will have the form

$$\sum_{\text{some } c\epsilon_1/2 \text{ or less terms from } X_e} p_i \leq \sum_{\text{terms not in } X_e} p_i.$$

(Note that we have assumed that $d\epsilon_2 \geq c\epsilon_1$ in the above.) Summing all such inequalities, we get

$$2\sum_{i\in X_e} p_i \leq 2\sum_{i\notin X_e} p_i.$$

This inequality contradicts the choice of X_e in $(***)$. Thus, case 2 cannot occur. Hence only case 1 can occur. This further implies that the minimum pseudocodeword weight is

$$w_{\min}^{BSC} \geq 2|X_e| = 2N\frac{c}{2d}\epsilon_1(\frac{\epsilon_1}{2} - \frac{\mu}{c}) = N\frac{c}{d}\epsilon_1(\frac{\epsilon_1}{2} - \frac{\mu}{c}).$$

To prove the claim in Case 2, observe that for the set X_e of left vertices in G' we have assumed that that $|S \cup T| = |S| + |T| > \frac{4}{c\epsilon_1}|X_e|$.

Furthermore for any subset $X \subseteq X_e$, let S_X be the set of right-left neighbors of X in G' and let T_X be the set of right-right neighbors of X in G'. Note that $|S_X \cup T_X| = |S_X| + |T_X| > \frac{4}{c\epsilon_1}|X|$. Otherwise, using the argument in case 1 and the fact that G is a (c, d, m, n, μ) expander, it can be shown that $|X| \geq N\frac{c}{2d}\epsilon_1(\frac{\epsilon_1}{2} - \frac{\mu}{c}) \geq |X_e|$, which is a contradiction. Thus, for any subset $X \subseteq X_e$, we have $|S_X \cup T_X| > \frac{4}{c\epsilon_1}|X|$. Now to prove the ϵ_1-matching for the set X_e, make two copies of every vertex in X_e to denote the two edges in G' that are incident to each vertex in X_e. Similarly, make $\frac{c\epsilon_1}{2}$ copies of every vertex in $S \cup T$ to denote the edges that are incident with each vertex in $S \cup T$ that are also incident with vertices in X_e. Let $\hat{X}_e$ denote the expanded set of vertices of X_e and let $\hat{S}$ denote the expanded set of S and let $\hat{T}$ denote the expanded set of T. Form a new graph $\hat{G}'$ that contains vertices $\hat{X}_e$ and $\hat{S}$ and $\hat{T}$. Form edges in $\hat{G}'$ in the following manner: if there is an edge from $u \in X_e$ to $v \in S$, then there is an edge from exactly one of the copies of u in $\hat{X}_e$ to exactly one of the copies of v in S and vice versa in $\hat{G}'$. Similarly, if there is an edge from $u \in X_e$ to $v \in T$ in G', then there is an edge from exactly one of the copies of u in $\hat{X}_e$ to exactly one of the copies of v in T and vice versa in $\hat{G}'$. This is always possible since $\frac{c\epsilon_1}{2}|S_X \cup T_X| \geq 2|X|$ for any $X \subseteq X_e$ in G'. In the new graph $\hat{G}'$, it is easy to see that for any subset $\hat{X}$ of left vertices, we have the size of its neighboring-set $|\Gamma(\hat{X})| \geq |\hat{X}|$ by the above observation (and by construction). Thus, by Hall's marriage theorem, there is a matching from the vertices in $\hat{X}_e$ to the vertices in $\Gamma(\hat{X}_e)$ in $\hat{G}'$. This matching reduces to an ϵ_1-matching as defined above for the set X_e in G', thereby proving the claim.

□

Remark 5.

- Note that if $c\epsilon_1 \geq d\epsilon_2 \geq 2\mu$, then it can be shown using a similar proof as in Theorem 4 that $w_{\min}^{BSC} \geq N\frac{d}{c}\epsilon_2(\frac{\epsilon_2}{2} - \frac{\mu}{d})$.
- Observe that the lower bound on the minimum pseudocodeword weight is slightly weaker compared to the lower bound on the minimum distance and the minimum stopping set size, since the proof in Theorem 4 exploits the strength of only one the subcodes – namely, the subcode with the smaller distance. We however believe that this can be improved to give a much stronger result as stated below.

- Note that in the case where $c = d$, $m = n$, and $\epsilon_1 = \epsilon_2 = \epsilon$, the result in Theorem 4 closely resembles the result in Theorem 3 and is almost equal to the lower bounds on the minimum distance and the minimum stopping set size.
- The lower bound suggests that if one were to use good expanding graphs such as the bipartite Ramanujan graphs from the construction in [3] and choose an appropriate choice of subcodes having minimum distance at least twice the second eigenvalue of the expander then the resulting code will have a good pseudocodeword weight and a good minimum distance. Once again, this is interesting for designing codes that are good for iterative decoding or LP decoding. Furthermore, with different choices of c and d, there is greater flexibility in the designing good codes using the construction in Case D than that in Case C.

We believe that Theorem 4 can be improved to a stronger result as follows:

Conjecture 1. *If $\epsilon_2 d \geq \epsilon_1 c > 2\mu$, the LDPC code obtained from the (c, d, m, n, μ) expander graph G has a minimum pseudocodeword weight*

$$w_{\min}^{BSC} \geq N(\frac{\epsilon_1\epsilon_2}{2} - \frac{\mu}{2\sqrt{cd}}(\epsilon_1\sqrt{\frac{c}{d}} + \epsilon_2\sqrt{\frac{d}{c}})).$$

7. A parity-oriented lower bound

Definition 14. The weight of a pseudocodeword $\mathbf{q} = (q_1, q_2, \ldots, q_n)$ of an LDPC constraint graph G on the AWGN channel is defined as [18, 21]

$$w^{AWGN}(\mathbf{q}) = \frac{(\sum_{i=1}^{n} q_i)^2}{(\sum_{i=1}^{n} q_i^2)}.$$

The following bound on the minimum pseudocodeword weight on the AWGN channel is an adaptation of Tanner's parity-oriented lower bound on the minimum distance [16]. Further, this bound complements the bit-oriented bound obtained by Vontobel and Koetter [22] which is also a lower bound on the minimum pseudocodeword weight in terms of the eigenvalues of the adjacency matrix of G, obtained using a slightly different argument.

Theorem 5. *Let G be a (j, m)-regular bipartite graph representing an LDPC code with an $r \times n$ parity check matrix H. Then the minimum pseudocodeword weight on the AWGN channel is lower bounded as*

$$w_{\min}^{AWGN} \geq \frac{n(4j - \mu_2 m)}{(\mu_1 - \mu_2)m},$$

where $\mu_1 = jm$ and μ_2 are the largest and second-largest eigenvalues (in absolute value), respectively, of HH^T.

Proof. Let $\mathbf{q} = (q_1, \ldots, q_n)$ be a pseudocodeword of G, and let $\mathbf{p} = H\mathbf{q}$ be a real-valued vector of length r. The first eigenvector of HH^T is $\mathbf{e}_1 = (1, 1, \ldots, 1)^T/\sqrt{r}$. Let $\mathbf{p}_i$ be the projection of $\mathbf{p}$ onto the ith eigenspace. We will now upper bound $\| H^T\mathbf{p} \|^2$. Converting $\| H^T\mathbf{p} \|^2$ into eigenspace representation, we get

$$\| H^T\mathbf{p} \|^2 = \sum_{i=1}^{r} \mu_i \| \mathbf{p}_i \|^2 = \mu_1 \| \mathbf{p}_1 \|^2 + \sum_{i=2}^{r} \mu_i \| \mathbf{p}_i \|^2$$

$$\leq \mu_1 \parallel \mathbf{p}_1 \parallel^2 + \mu_2(\parallel \mathbf{p} \parallel^2 - \parallel \mathbf{p}_1 \parallel^2).$$

Note that

$$\parallel \mathbf{p}_1 \parallel^2 = \frac{j^2}{r}(\sum_{i=1}^{n} q_i)^2, \text{ and}$$

$$\parallel \mathbf{p} \parallel^2 \leq mj(\sum_{i=1}^{n} q_i^2).$$

The first equality follows from the choice of $\mathbf{p}$ and the regularity of the parity check matrix H. The second inequality follows by applying the identity $(q_1 + q_2 + \cdots + q_t)^2 \leq t(q_1^2 + q_2^2 + \cdots + q_t^2)$ to the terms in the expansion of $\parallel \mathbf{p} \parallel^2$.

The above set of equations yield

$$\parallel H^T\mathbf{p} \parallel^2 \leq (\mu_1 - \mu_2)\frac{j^2}{r}(\sum_{i=1}^{n} q_i)^2 + \mu_2 mj(\sum_{i=1}^{n} q_i^2).$$

We now lower bound $\parallel H^T\mathbf{p} \parallel^2$ as follows

$$\parallel H^T\mathbf{p} \parallel^2 = \sum_{t=1}^{n}(\sum_{i=1}^{r}\sum_{\ell=1}^{n} h_{i,t}h_{i,\ell}q_\ell)^2 \geq (4j^2)(\sum_{t=1}^{n} q_t^2).$$

This bound may be seen by observing that for each t in the outer summation, the inner sums over the indices i and ℓ contribute j q_t terms and $(m-1)j$ terms involving other q_k's. When t is fixed, for each i wherein $h_{it} = 1$, we have q_t and $(m-1)$ other q_k's that contribute to the inner sum. Since q_t and the $(m-1)$ other q_k's are involved in the ith constraint node and since $\mathbf{q}$ is a pseudocodeword, we have q_t+sum of $(m-1)$ other q_k's $\geq 2q_t$. Since there are j values of i wherein $h_{it} = 1$, for a fixed t, the inner sum over i and ℓ can be lower bounded by $2jq_t$. Thus, $\parallel H^T\mathbf{p} \parallel^2 \geq \sum_{t=1}^{n}(2jq_t)^2 = 4j^2(\sum_{t=1}^{n} q_t^2)$.

Combining the upper and lower bounds, we get

$$\frac{(4j^2 - \mu_2 mj)r}{(\mu_1 - \mu_2)j^2} \leq \frac{(\sum_{i=1}^{n} q_i)^2}{(\sum_{i=1}^{n} q_i^2)} = w^{AWGN}(\mathbf{q}).$$

Since $nj = rm$, we obtain the desired lower bound. □

Remark 6. Note that this lower bound is not strong as the bit-oriented bound in [22]. It equals the bit-oriented bound for the case when $m = 2$. However, we believe that by a different but judicious choice of $\mathbf{p}$ in the above proof and by using stronger intermediate bounding steps, a much stronger parity-oriented bound can be obtained.

8. Conclusions

In this paper, the expander-based (i.e., eigenvalue-type) lower bounds on the minimum distance of expander codes were extended to lower bound the minimum stopping set size and the minimum pseudocodeword weight of these codes. A new parity-oriented lower bound in terms of the eigenvalues of the parity-check matrix was also obtained for the minimum pseudocodeword weight of LDPC codes on the AWGN channel. These lower bounds indicate that LDPC codes constructed from expander graphs provide a certain guaranteed level of performance and error-correction capability with graph-based iterative decoding as well as linear programming decoding. Further, the results indicate that if the underlying LDPC constraint graph is a good expander, then the corresponding expander code is likely to have a

very good minimum pseudocodeword weight relative to the block length. This is in general a very hard criterion to ensure in the construction of good error correcting codes at large block lengths. It would be interesting to derive upper bounds on the distance, stopping set size, and pseudocodeword weight of expander codes to examine how tight the derived lower bounds are.

Acknowledgments

We thank Joachim Rosenthal for a careful proof-reading of this paper. We would like to thank the reviewers for their valuable comments. We believe their feedback has greatly improved the paper. We also thank Reviewer 1 for providing the more intuitive definition of a stopping set in a generalized LDPC code.

`<ckelley@math.ohio-state.edu; sridhara@math.unizh.ch>`

References

[1] N. Linial and A. Wigderson, *Expander graphs and their applications*, Lecture notes of a course given at the Hebrew University, 2003. Available under http://www.math.ias.edu/ avi/TALKS/

[2] N. Alon, *Eigenvalues and expanders*, Combinatorica, **6** (1986), 83–96.

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

[4] G. A. Margulis, *Explicit constructions of graphs without short cycles and low-density codes*, Combinatorica, **2**(1) (1982), 71–78.

[5] A. Lubotzky, *Discrete groups, expanding graphs and invariant measures*, Progress in Mathematics, **125**, Birkhauser, Basel, 1994.

[6] N. Alon, A. Lubotzky, and A. Wigderson, *Semi-direct product in groups and zig-zag product in graphs: connections and applications (extended abstract)*. In: 42nd IEEE Symposium on Foundations of Computer Science (Las Vegas, NV, 2001), IEEE Computer Soc., Los Alamitos, CA, (2001), 630–637.

[7] O. Reingold, S. Vadhan, and A. Wigderson, Entropy waves, the zig-zag product, and new constant-degree expanders and extractors in graphs: connections and applications,Annals of Mathematics, **155**(1) (2002), 157–187.

[8] M. Sipser and D. A. Spielman, *Expander codes*, IEEE Trans. Inform. Theory, **42**(6) (1996), 1710–1722.

[9] J. Lafferty and D. Rockmore, *Codes and iterative decoding on algebraic expander graphs*, In: Proceedings of ISITA 2000, Honolulu, Hawaii, available at http://www-2.cs.cmu.edu/afs/cs.cmu.edu/user/lafferty/www/pubs.html, Nov. 2000.

[10] H. Janwa and A. K. Lal,On Tanner codes: Minimum distance and decoding, Proceedings of AAECC, **13** (2003), 335–347.

[11] R. Koetter and P. O. Vontobel, *Graph-covers and iterative decoding of finite length codes*, In: Proceedings of the 2003 Intl. Symposium on Turbo codes, Brest, France.

[12] C. Kelley and D. Sridhara, Pseudocodewords of Tanner graphs. Submitted to IEEE Trans. on Information Theory, June 2005.

[13] J. Feldman, T. Malkin, R. A. Servedio, C. Stein, and M. J. Wainwright, *LP decoding corrects a constant fraction of errors*, CORC Technical Report: TR-2003-08.

[14] J. Feldman, *Decoding Error-Correcting Codes via Linear Programming Decoding*, Ph.D. Thesis, M.I.T., Cambridge, MA, 2003.

[15] J. Feldman, M. J. Wainwright, D. R. Karger, Using Linear Programming to Decode Binary Linear Codes. IEEE Transactions on Information Theory, 51(3), 954 – 972, Mar. 2005.

[16] R. M. Tanner, *Minimum distance bounds by graph analysis*, IEEE Trans. Inform. Theory, **47**(2) (2001), 808–821.

[17] C. Di, D. Proietti, T. Richardson, E. Teletar, and R. Urbanke, *Finite-length analysis of low-density parity-check codes on the binary erasure channel*, IEEE Trans. Inform. Theory, **48** (2002), 1570–1579.

[18] G. D. Forney, Jr., R. Koetter, F. Kschischang, and A. Reznik, On the effective weights of pseudocodewords for codes defined on graphs with cycles, In: B. Marcus and J. Rosenthal, editors, *Codes, Systems and Graphical Models*, IMA Vol. **123** (2001), 101–112. Springer-Verlag.

[19] N. Alon and F. R. K. Chung, *Explicit construction of linear sized tolerant networks*, Discrete Mathematics, **72** (1988), 15–19.

[20] C. Kelley, D. Sridhara, J. Xu, and J. Rosenthal: *Pseudocodeword weights and stopping sets*. In: Proc. of the IEEE International Symposium on Information Theory, (Chicago, USA), page 150, 2004.

[21] N. Wiberg, *Codes and Decoding on General Graphs*. PhD thesis, University of Linköping, Sweeden, 1996.

[22] P. O. Vontobel and R. Koetter, Lower bounds on the minimum pseudo-weight of linear codes, In: Proc. of the IEEE International Symposium on Information Theory, (Chicago, USA), 2004.

[23] R. M. Tanner, *A recursive approach to low complexity codes*, IEEE Trans. Inform. Theory, **27**(5) (1981), 533–547.

Submitted: September 28, 2006. Revised: January 14, 2007.