

# Strong Secrecy on the Binary Erasure Wiretap Channel Using Large-Girth LDPC Codes

Arunkumar Subramanian\*, *Student Member, IEEE*, Andrew Thangaraj, *Member, IEEE*,  
Matthieu Bloch, *Member, IEEE*, Steven W. McLaughlin, *Fellow, IEEE*,

**Abstract**—For an arbitrary degree distribution pair (DDP), we construct a sequence of low-density parity-check (LDPC) code ensembles with girth growing logarithmically in block-length using Ramanujan graphs. When the DDP has minimum left degree at least three, we show using density evolution analysis that the expected bit-error probability of these ensembles, when passed through a binary erasure channel with erasure probability  $\epsilon$ , decays as  $\mathcal{O}(\exp(-c_1 n^{c_2}))$  with the block-length  $n$  for positive constants  $c_1$  and  $c_2$ , as long as  $\epsilon$  is less than the erasure threshold  $\epsilon_{\text{th}}$  of the DDP. This guarantees that the coset coding scheme using the dual sequence provides strong secrecy over the binary erasure wiretap channel for erasure probabilities greater than  $1 - \epsilon_{\text{th}}$ .

## I. INTRODUCTION

The notion of information-theoretic security on a communication system with a passive eavesdropper was first introduced by Shannon in [1]. This model consists of three parties — Alice, Bob and Eve; Alice wants to convey a secret message  $S$  to Bob without revealing it to Eve, who can passively intercept the transmission. Shannon’s model involves noiseless communication channels and secret communication can be achieved only if Alice and Bob share an encryption key that is not known to Eve. Alice converts the message  $S$  into an  $n$ -symbol cryptogram  $X^n$  using the key  $K$  and transmits the cryptogram to Bob. The communication scheme is said to attain *perfect secrecy* if  $\mathbb{I}(S; X^n) = 0$ . Shannon proved that perfect secrecy is guaranteed only if  $\mathbb{H}(K) \geq \mathbb{H}(S)$ . In other words, Alice and Bob must share a secret key that is at least as long as the confidential message.

Wyner [2] introduced an alternate model called the wiretap channel where communication occurs over noisy channels, with Eve receiving a degraded version of the signal received by Bob. Csiszár-Körner [3] considered a generalization of this model where Eve’s reception need not be a degraded version of Bob’s received signal. In these models, it is possible to achieve secret communication without using a pre-shared encryption

key. This is done by exploiting the fact that the wiretapper’s channel is “noisier” than the legitimate user’s channel.

Since perfect secrecy may not be achievable over the wiretap channel for short block-lengths, Wyner introduced the asymptotic notion of *weak secrecy*. If  $Z^n$  is the length- $n$  symbol vector received by Eve, weak secrecy is said to be achieved if the rate of information leakage  $\frac{1}{n}\mathbb{I}(S; Z^n)$  vanishes as  $n \rightarrow \infty$ . The authors of [2], [3] calculated the secrecy capacity of the respective channels under the weak secrecy condition. A shortcoming of weak secrecy is that the amount of leaked information can be unbounded even if the rate at which it is leaked goes to zero. Maurer and Wolf [4] highlighted this shortcoming and introduced the notion of *strong secrecy*, which requires that the total amount of leaked information  $\mathbb{I}(S; Z^n)$  must vanish as  $n \rightarrow \infty$ . Though the strong secrecy condition is more stringent, it does not reduce the secrecy capacity [4], [5].

In this paper, we consider the binary erasure wiretap channel (BEWC) model, where Bob’s channel is noiseless and Eve’s channel is a binary erasure channel (BEC). Note that this wiretap model is also called a “binary-erasure-channel wiretap” (BEC-WT) [6] in literature<sup>1</sup>. The BEWC model is important because other wiretap scenarios can be modeled based on the BEWC. For example, the scenarios with a noiseless main channel and a binary symmetric or an additive white Gaussian noise wiretap channel can be modeled [6] as degraded BEWCs. Moreover, the Erasure Decomposition Lemma [10, Lemma 4.78] lets us model wiretap systems with a noiseless main channel and an arbitrary binary-input memoryless symmetric-output wiretap as degraded BEWCs. We employ the forward coding approach to achieve secrecy on our wiretap model. Other approaches, like the ones in [4], [11], use public discussion on authenticated channels in addition to communication on the wiretap channel to achieve secrecy on the overall system.

In [12], Thangaraj, et al. proposed using the duals of low-density parity-check (LDPC) codes in a “coset coding scheme” [2], [7] to achieve weak secrecy on the BEWC. They showed

A. Subramanian and S. W. McLaughlin are with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA. E-mail: arunkumar@gatech.edu, swm@ece.gatech.edu. Phone: +1-404-385-3383. Fax: +1-404-894-7883.

A. Thangaraj is with the Department of Electrical Engineering, Indian Institute of Technology Madras, Chennai - 600036, India. E-mail: andrew@iitm.ac.in. Phone: +91-44-2257-4424.

M. Bloch is with Georgia Tech - CNRS UMI 2958 and with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, 57070 Metz, France. E-mail: matthieu.bloch@ece.gatech.edu. Phone: +33-3-87-20-39-26.

\* Corresponding author

<sup>1</sup>Liu, et al. [6] considered a generalized version of our BEWC model where there is a binary-input memoryless symmetric-output wiretap, and called it a type-II wiretap model. The original type-II wiretap model introduced by Ozarow-Wyner [7] has a noiseless main channel with a fixed number of transmitted bits revealed to the eavesdropper. The eavesdropper is also able to choose which bit locations are revealed to her. These two wiretap models are different — in [6], the wiretapper’s channel is memoryless, whereas in [7] it is not.

In [6], the BEWC model is called a BEC-WT. Contrastingly, Rathi, et al. [8], [9] use BEC-WT to denote the wiretap model with a BEC main channel and an independent BEC wiretapper’s channel.

that this scheme achieves weak secrecy over  $\text{BEWC}(\xi)$  for  $\xi > 1 - \epsilon_{\text{th}}$ , where  $\epsilon_{\text{th}}$  is the BEC threshold of the LDPC code ensembles under message passing (MP) decoding. As an extension to this result, Suresh, et al. [13] showed that strong secrecy on the BEWC can be achieved using the duals of short-cycle-free LDPC codes in the coset coding scheme. They first show that a sufficient condition for strong secrecy is to have the MP block-error probability decay as  $\mathcal{O}(1/n^2)$ . Using a stopping set based analysis, they prove that short-cycle-free LDPC code ensembles satisfy this condition. Specifically, they show that strong secrecy is achieved on  $\text{BEWC}(\xi)$  for  $\xi > 1 - \epsilon_{\text{ef}}$ , where  $\epsilon_{\text{ef}}$  is the lower bound on error-floor of LDPC codes defined in [14]. Since  $\epsilon_{\text{ef}} < \epsilon_{\text{th}}$ , there is a gap between the strong and weak secrecy thresholds for finite-girth ensembles.

The work presented in this paper is based on the LDPC code based coset coding scheme of [12], [13]. We show that the duals of “large-girth” LDPC codes achieve strong secrecy on the BEWC with no gap between the strong and weak secrecy thresholds. We do this by first analyzing the asymptotic behaviour of the bit-error probability estimate given by density evolution as the number of iterations increases monotonically. We then construct irregular Tanner graphs for arbitrary degree distribution pairs such that the girth of these graphs increases logarithmically in the number of vertices. This construction, which can potentially create disconnected Tanner graphs, is based on the large-girth regular Ramanujan graphs constructed by Lubotzky, et al. [15]. We show that the LDPC codes based on our graphs have a bit-error probability that closely follows the density evolution estimate for increasing iterations. This property, together with the logarithmic increase in the girth of the underlying graphs, guarantees that the duals of our LDPC codes will achieve strong secrecy on the BEWC.

In recent work [16], [17], polar codes have been suggested as methods for approaching the secrecy capacity of general degraded and symmetric wiretap channels, of which the erasure wiretap channel is a special case. However, since the threshold phenomenon of LDPC codes is observed at shorter block-lengths than polarization, there is enough interest in studying the strong secrecy properties of LDPC code ensembles. Also, the mechanism of security using polar and LDPC codes is different. Security of polar codes is proved using the capacity-approaching properties of these codes. In the case of LDPC code ensembles over erasure wiretap channels, we use the duals of codes with a threshold property that need not be close to capacity.

This paper is organized as follows. In Section II, we give a brief introduction of the channel model and the coset coding scheme, and relate the strong secrecy condition to the bit-error probability of the duals of the codes used in the coset coding scheme. In Section III, we give a brief overview of the density evolution analysis for LDPC codes and state the result regarding the double exponential decay of the density evolution bit-error probability estimate as the number of iterations increases. We then show that this result translates to strong secrecy on the BEWC using the duals of large-girth regular LDPC codes. In Section IV, we provide a quick overview of existing constructions for graphs with good girth. We then describe our construction of large-girth graphs

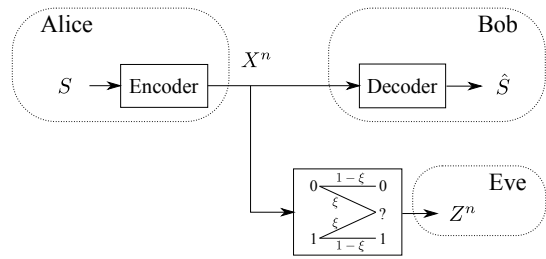


Fig. 1. The binary erasure wiretap channel model.

and prove that the duals of the resulting LDPC codes achieve strong secrecy.

## II. COSET CODING SCHEME AND STRONG SECRECY

We consider the binary erasure wiretap channel (BEWC) model introduced in [12], which consists of two legitimate parties, Alice and Bob, and an eavesdropper, Eve (Fig. 1). The channel from Alice to Bob is noiseless and Eve sees the bits sent to Bob through a binary erasure channel (BEC) with erasure probability  $\xi$ .

Prior to transmission, Alice and Bob publicly agree on a  $(n, n(1 - R))$  binary linear code  $C$ . For each possible value  $\mathbf{s}$  of the  $nR$  bit secret vector  $S$ , we associate a coset of  $C$  given by  $C(\mathbf{s}) = \{\mathbf{x}^n \in \{0, 1\}^n : \mathbf{x}^n \mathbf{H}^T = \mathbf{s}\}$ , where  $\mathbf{H}$  is the parity check matrix of  $C$ . Note that all the vectors in this paper are assumed to be row vectors. To convey the message  $S$  to Bob, Alice picks a vector  $X^n$  from one of the  $2^{n(1-R)}$  vectors of  $C(S)$  at random and transmits it over  $\text{BEWC}(\xi)$ . Bob obtains the secret message from the received vector by calculating  $X^n \mathbf{H}^T$ .

The amount of information that is leaked to Eve through her observation  $Z^n$  can be bounded [13] as

$$\mathbb{I}(S; Z^n) \leq n R P_B^{\text{MAP}}(C^\perp, 1 - \xi)$$

where  $P_B^{\text{MAP}}(C^\perp, 1 - \xi)$  is the block-error probability under maximum a posteriori (MAP) decoding of the dual code  $C^\perp$  transmitted over  $\text{BEC}(1 - \xi)$ . A weakened form of this upper bound can be obtained by substituting  $P_B^{\text{MAP}}$  with  $P_B^{\text{MP}}$ , the block-error probability using the message passing (MP) decoder with infinite number of iterations.

The security condition on a sequence of encoding schemes of increasing block-length  $n$  and a constant rate  $R$  to achieve strong secrecy on a wiretap channel is

$$\mathbb{I}(S; Z^n) \rightarrow 0 \quad \text{as } n \rightarrow \infty$$

**Lemma 1** (adapted from [13, Corollary 1]). *If  $(C_n^\perp)$  is a sequence of binary linear block codes of rate  $R$  with increasing block length  $n$  such that for some  $\alpha > 1$ ,*

$$P_B^{\text{MP}}(C_n^\perp, 1 - \xi) = \mathcal{O}\left(\frac{1}{n^\alpha}\right)$$

*then strong secrecy is achieved on  $\text{BEWC}(\xi)$  when the dual sequence  $(C_n)$  is used under the coset coding scheme.*

We use the standard order notations  $\mathcal{O}$ ,  $o$ ,  $\Theta$ ,  $\omega$  and  $\Omega$  as defined in [18]. From Lemma 1, it is clear that the

sequence  $(C_n)$  achieves strong secrecy on BEWC( $\xi$ ) if the dual sequence has

$$P_B^{\text{MP}}(C_n^\perp, 1 - \xi) = \mathcal{O}\left(\frac{1}{n^2}\right)$$

Using the union bound on the block-error probability,

$$P_B^{\text{MP}}(C_n^\perp, 1 - \xi) \leq nP_b^{\text{MP}}(C_n^\perp, 1 - \xi)$$

where  $P_b^{\text{MP}}$  is the bit-error probability using the MP decoder.

**Corollary 2.** *If a sequence  $(C_n)$  of binary linear codes with increasing block length  $n$  and rate  $1 - R$  is such that the dual sequence  $(C_n^\perp)$  has a bit-error probability such that*

$$P_b^{\text{MP}}(C_n^\perp, 1 - \xi) = \mathcal{O}\left(\frac{1}{n^3}\right)$$

*then strong secrecy is achieved on BEWC( $\xi$ ) when  $(C_n)$  is used under the coset coding scheme.*

### III. ASYMPTOTIC BEHAVIOUR OF THE BEC DENSITY EVOLUTION FORMULA

The asymptotic behaviour of the bit-error probability of LDPC codes under MP decoding can be tracked using density evolution [10, Sec. 3.9]. The main result of our paper is based on the asymptotic behaviour of the BEC density evolution expression as the number of iterations goes to infinity. Since density evolution gives only an approximate value of the bit-error probability and our ultimate aim is to study the asymptotic behaviour of the bit-error probability, it is important to understand where and how approximations are made in density evolution.

#### A. Density Evolution Analysis - a Background

Let  $\mathcal{H}_n$  be an arbitrary ensemble of Tanner graphs with  $n$  variable nodes. Suppose a graph  $G$  is selected uniformly at random from  $\mathcal{H}_n$  and a random codeword from the associated block code is transmitted over BEC( $\epsilon$ ). The receiver, with the knowledge of  $G$ , tries to decode the transmitted word using the MP algorithm. For a family of ensembles  $(\mathcal{H}_n)$  with increasing  $n$ , let

- $x(t, n)$  = the probability that a randomly selected edge in the Tanner graph transmits an erasure message from its variable node to its check node at the  $t^{\text{th}}$  iteration
- $y(t, n)$  = the probability that a randomly selected codeword bit is unknown after  $t$  iterations

1) *Computation Graphs* [10, Sec. 3.7.1]: To evaluate  $x(t, n)$  and  $y(t, n)$  explicitly, the computation graphs associated with the Tanner graph ensemble may be considered. Suppose a graph  $G$  is selected from  $\mathcal{H}_n$  uniformly at random and a random edge  $e$  is picked from  $G$ . Let  $v$  be the variable node connected to  $e$ . The level- $t$  *edge-rooted computation graph*  $\vec{C}_t$  of  $\mathcal{H}_n$  is defined as the subgraph obtained by traversing from  $v$  up to iteration depth  $t$  in all directions except along  $e$ .  $\vec{C}_t$  is a random graph whose distribution depends only on  $t$  and  $\mathcal{H}_n$ . Also,  $x(t, n)$  can be uniquely determined given the possible realizations of  $\vec{C}_t$  and their probabilities (regardless of what  $\mathcal{H}_n$  is).

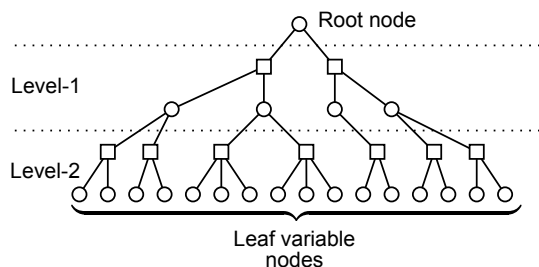


Fig. 2. An example of a level-2 decoding neighbourhood tree.

To evaluate  $y(t, n)$ , the level- $t$  *node-rooted computation graph*  $\hat{C}_t$ , defined subsequently, may be considered. As before, a graph  $G$  is selected from  $\mathcal{H}_n$  uniformly at random. Then, a random variable node  $v$  is picked from  $G$ .  $\hat{C}_t$  is defined as the subgraph obtained by traversing from  $v$  up to iteration depth  $t$  in all directions. Like  $\vec{C}_t$ ,  $\hat{C}_t$  is also dependent only on  $t$  and  $\mathcal{H}_n$ .  $y(t, n)$  can be uniquely determined given the possible realizations of  $\hat{C}_t$  and their probabilities (again, regardless of  $\mathcal{H}_n$ ).

2) *Tree Ensembles* [10, Sec. 3.7.2]: While studying the error-correcting performance of LDPC codes, the codes corresponding to the socket permutation ensemble of Tanner graphs, denoted by  $\mathcal{G}(n, \lambda, \rho)$ , are usually considered. The graphs in this ensemble contain  $n$  variable nodes, whose degrees are determined by the degree distribution polynomial  $\lambda(x) = \sum_i \lambda_i x^{i-1}$ , where  $\lambda_i$  is the fraction of edges that are connected to degree- $i$  variable nodes. The check-node degree distribution is determined by the polynomial  $\rho(x) = \sum_j \rho_j x^{j-1}$ , where  $\rho_j$  is the fraction of edges connected to degree- $j$  check nodes.

In the classical setting,  $\mathcal{H}_n = \mathcal{G}(n, \lambda, \rho)$  is considered, and  $x(t, n)$  and  $y(t, n)$  are analyzed while keeping  $t$  fixed and letting  $n$  grow monotonically. The possible computation graphs of  $\mathcal{G}(n, \lambda, \rho)$  are not cycle-free and hence enumerating them is cumbersome. Doing an exact analysis of the bit-error probability  $y(t, n)$  for this ensemble is therefore difficult. Density evolution resorts to an approximate analysis by considering *tree ensembles*, which are asymptotic approximations of computation graphs.

The *node-rooted tree ensemble*  $\vec{T}_t$  is an approximation of the computation graph  $\vec{C}_t$  of  $\mathcal{G}(n, \lambda, \rho)$ .  $\vec{T}_t$  is a random graph which takes all possible level- $t$  decoding neighbourhood trees permitted by  $(\lambda, \rho)$  and it is generated by the following rules.

- The degrees of all nodes are chosen independently.
- The root variable node has degree  $i$  with probability  $L_i$ , where  $L_i$  is the fraction of degree- $i$  variable nodes in  $\mathcal{G}(n, \lambda, \rho)$ .
- All the leaf variable nodes have degree one.
- All other variable nodes have degree  $i$  with probability  $\lambda_i$ .
- Check nodes have degree  $j$  with probability  $\rho_j$ .

The *edge-rooted tree ensemble*  $\vec{T}_t$ , which is an asymptotic approximation of  $\vec{C}_t$ , is defined in a similar manner, except for the fact that the root variable node has degree  $i$  with probability  $\lambda_{i+1}$ .

Suppose the block code corresponding to the Tanner graph  $\vec{T}_t$  is transmitted over BEC( $\epsilon$ ). The probability that the root

node is unknown after  $t$  iterations of the MP decoder, denoted by  $x_t$ , is given by the recursive equation

$$x_t = \epsilon \lambda (1 - \rho(1 - x_{t-1})) =: f(\epsilon, x_{t-1})$$

with  $x_0 = \epsilon$ . The threshold  $\epsilon_{\text{th}}$  is defined as the supremum of all values of  $\epsilon$  for which the sequence  $(x_t)$  converges to zero.

For the ensemble  $\tilde{\mathcal{T}}_t$  a similar probability  $y_t$  may be defined. We will have

$$y_t = \epsilon L(1 - \rho(1 - x_{t-1}))$$

where  $L(x) = \sum_i L_i x^i$  is the degree distribution polynomial from the node perspective.

### B. Density Evolution - Asymptotic Behaviour

It is a well-known result that  $x_t$  and  $y_t$  exhibit a double-exponential decay as  $t$  goes to infinity for  $\epsilon < \epsilon_{\text{th}}$ . A proof of this result for regular codes is provided in [19, Sec. V-A]. For the sake of completeness, we state the more general result for irregular codes and provide an alternative proof (see Appendix A) involving mathematical induction.

**Lemma 3.** *For a distribution pair  $(\lambda, \rho)$  with minimum variable node degree  $l_{\min} \geq 3$  and  $\epsilon < \epsilon_{\text{th}}$ , we have*

$$x_t, y_t = \mathcal{O}(\exp(-\beta(l_{\min} - 1)^t)) \quad (1)$$

as  $t \rightarrow \infty$ , where  $\beta > 0$  is a constant.

It is important to note that a similar *double exponential* decay result is not true for DDPs that have degree-2 variable nodes. Working out the expressions for this case, we can see that  $x_t$  (and  $y_t$ ) exhibits only an *exponential* decay in the number of iterations. Note that  $x_t$  is the expectation of the root-node bit-error probability taken over the possible outcomes of  $\vec{\mathcal{T}}$ . The dominating term in this expectation is the contribution of the worst-case trees, namely, the trees that contain *only* variable nodes of the least degree. DDPs with degree-2 variable nodes form a special case where the contribution by the worst-case trees decays only exponentially.

### C. Asymptotic Decay of Bit-Error Probability

Suppose we are given a DDP  $(\lambda, \rho)$  with  $l_{\min} \geq 3$ . For  $k = 1, 2, 3, \dots$ , let  $(n_k)$  be a strictly increasing sequence of positive integers and let  $t_k$  be such that

$$t_k = \left\lceil \frac{\log \log n_k + \log a - \log \beta}{\log(l_{\min} - 1)} \right\rceil$$

for any positive integer  $a$ . This means that  $y_{t_k} = \mathcal{O}(1/n_k^a)$ . In particular, we have  $y_t = \mathcal{O}(1/n^3)$  for  $a = 3$  (we drop the subscript  $k$  for convenience). Since  $y_t$  is only an approximation of  $y(t, n)$ , this does not necessarily mean that the actual bit-error probability  $y(t, n)$  itself decays as  $\mathcal{O}(1/n^3)$ .

Our ultimate aim is to prove an information-theoretic result and this requires rigorous mathematical proofs. There are only a few rigorous results regarding the ‘‘closeness’’ of the density evolution approximation. For example, we know the following results

- For  $\mathcal{G}(n, \lambda, \rho)$

$$\lim_{n \rightarrow \infty} x(t, n) = x_t, \quad \lim_{n \rightarrow \infty} y(t, n) = y_t$$

as long as  $t$  remains constant [10, Thm. 3.49].

- The ‘‘exchange of limits’’ result by Korada-Urbanke [20].

To achieve strong secrecy, we must find some ensemble  $\mathcal{H}_n$  for which  $y(t, n) = \mathcal{O}(1/n^3)$ , where  $t$  is growing with  $n$  at least as fast as  $\log \log n$ . In general, this is not true for  $\mathcal{G}(n, \lambda, \rho)$ . For example, any irregular DDP with  $l_{\min} = 3$  does not satisfy  $y(t, n) = \mathcal{O}(y_t)$  any  $\epsilon > 0$  (see [14, Thm. 16]).

1) *Strong Secrecy on the BEWC Using Large-Girth Regular LDPC Codes:* Let  $\mathcal{G}_g(n, \lambda, \rho)$  denote the subset of Tanner graphs in  $\mathcal{G}(n, \lambda, \rho)$  whose girth is at least  $g$ . Clearly, the level- $t$  computation graphs of  $\mathcal{G}_{4t+2}(n, \lambda, \rho)$  are cycle free. This means that any possible outcome of  $\hat{\mathcal{C}}_t$  is also a possible outcome of  $\tilde{\mathcal{T}}_t$ . This does not necessarily mean that  $\hat{\mathcal{C}}_t$  and  $\tilde{\mathcal{T}}_t$  are identically distributed and therefore,  $y(t, n) = y_t$  is not necessarily true for  $\mathcal{G}_{4t+2}(n, \lambda, \rho)$ .

The regular LDPC code ensemble  $\mathcal{G}_{4t+2}(n, x^{c-1}, x^{d-1})$  is a special case for which  $\hat{\mathcal{C}}_t$  and  $\tilde{\mathcal{T}}_t$  are equal to a unique tree  $T$ . Since  $y(t, n)$  is calculated from  $\hat{\mathcal{C}}_t$  in the same way as  $y_t$  is calculated from  $\tilde{\mathcal{T}}_t$ , we have  $y(t, n) = y_t$ . Using a similar reasoning, we can also say that  $x(t, n) = x_t$ .

In essence, density evolution analysis is approximate because it makes the following assumptions.

- 1) The decoding neighbourhood is a tree
- 2) The degrees of the nodes in the decoding tree can be chosen independently.

For large-girth Tanner graphs, the first assumption is justified. However, the second assumption is not justified for large-girth irregular Tanner graphs. For large-girth *regular* Tanner graphs, there is a unique decoding neighbourhood with only one choice for variable-node degrees and only one choice for check-node degrees, which means that the second assumption is also justified. Therefore, we are able to assert that the density evolution estimate is exact in the case of large-girth regular LDPC codes, but are unable to do the same for the irregular counterpart.

Assume that there exists a sequence  $(C_n^\perp)$  of  $(c, d)$ -regular LDPC codes with  $c \geq 3$  such that their Tanner graphs have girth at least  $4t + 2$  with

$$t = \left\lceil \frac{\log \log n + \log 3 - \log \beta}{\log(c - 1)} \right\rceil$$

(The existence of such codes will be proved in the next section). For these codes, we have

$$P_b^{\text{MP}}(C_n^\perp, \epsilon, t) := y(t, n) = y_t \quad (2)$$

$$= \mathcal{O}\left(\frac{1}{n^3}\right) \quad (3)$$

for  $\epsilon < \epsilon_{\text{th}}$ . Here,  $P_b^{\text{MP}}(C_n^\perp, \epsilon, t)$  denotes the bit-error probability after  $t$  iterations. By the above equation, the dual sequence  $(C_n)$  will achieve strong secrecy on BEWC( $\xi$ ) for  $\xi > 1 - \epsilon_{\text{th}}$ .

## IV. LARGE-GIRTH GRAPHS

### A. Existence of Large-Girth Graphs

For our scheme to achieve strong secrecy, we require a sequence of regular bipartite graphs whose girth increases

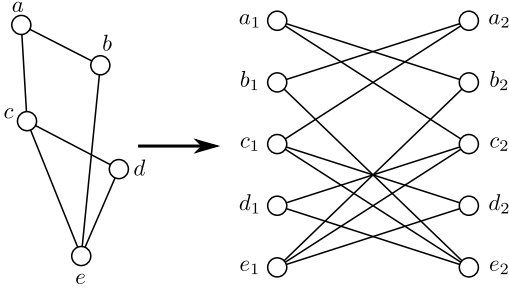


Fig. 3. An illustration of Algorithm 1 to create bipartite graphs.

faster than  $\log \log n$ , where  $n$  is the number of vertices. We define a sequence of *large-girth graphs* as one with a prescribed degree distribution whose girth increases as  $\log n$ . These large-girth graphs satisfy the girth condition required for strong secrecy. The existence of these graphs is related to the problem of *cages* [21] in graph theory. A  $\nu$ -regular (simple) graph is one where each vertex has exactly  $\nu$  neighbours. A  $(\nu, g)$  *cage* is a vertex minimal  $\nu$ -regular graph of girth  $g$ . Erdős and Sachs [22] showed that cages exist for all  $\nu \geq 3$  and  $g \geq 3$ . For a given  $\nu \geq 4$ , let  $(R_g)_{g \in \mathbb{N}}$  be a sequence of  $(\nu, g)$  cages in  $n(g)$  vertices. From the upper [23] and lower [22] lower bounds on  $n$ , we have  $g = \Theta(\log n)$ . This means that  $(R_g)$  is a sequence of large-girth graphs. Since cages are not necessarily bipartite and we require a sequence of large-girth *bipartite* graphs, we make use of Algorithm 1 [24, Sec. 3.1]. For convenience, we denote the operation performed by this algorithm by  $B(\cdot)$ .

---

**Algorithm 1** Construction of a bipartite graph given any graph [24, Sec. 3.1].

---

- 1: Given a graph  $G$  in  $n$  vertices, create an identical copy  $G'$  with  $V(G) \cap V(G') = \emptyset$ . Let  $f : V(G) \rightarrow V(G')$  be a graph homomorphism.
  - 2: Create a graph  $H$  with vertex set  $V(H) = V(G) \cup V(G')$  and edge set  $E(H) = \{\{x, y\} : x \in V(G), y \in V(G'), f(x) \sim y \text{ in } G'\}$ . That is, if  $a_1, b_1 \in V(G)$ ,  $a_2 = f(a_1)$ ,  $b_2 = f(b_1)$  and  $a_1 b_1 \in E(G)$  (or equivalently, if  $a_2 b_2 \in E(G')$ ), then  $a_1 b_2, a_2 b_1 \in E(H)$ .
- 

**Lemma 4.** *Given a graph  $G$ , if  $H = B(G)$  then  $g(H) \geq g(G)$ .*

*Proof:* For any cycle  $C$  in  $H$  with the vertices in the order  $(u_0, v_0, u_1, v_1, \dots, u_{r-1}, v_{r-1}, u_0)$  there exists a closed walk

$$W = (u_0, f^{-1}(v_0), u_1, f^{-1}(v_1), \dots, u_{r-1}, f^{-1}(v_{r-1}), u_0)$$

in  $G$ . Note that  $r \geq 2$ .  $W$  can either contain a cycle, or it can be a path tracing itself back after some point. We show that this closed walk has a cycle.

Suppose this closed walk does not have a cycle. Then, without loss of generality, we can assume that it traces itself back at some point  $v_i$ . Therefore, the sequence  $u_i, v_i, u_{(i+1) \bmod r}$  is such that  $u_i = u_{(i+1) \bmod r}$ . This is a contradiction since all the vertices in the original cycle are distinct and  $r \geq 2$ .

Therefore,  $W$  contains a cycle  $C^*$ . We have

$$\text{length}(C) \geq \text{length}(C^*) \geq g(G)$$

which proves that  $g(H) \geq g(G)$ . Note that the second inequality in the above equation follows from the fact that  $C^*$  is a cycle in  $G$ . ■

By the above lemma,  $(B(R_g))_{g \in \mathbb{N}}$  is a sequence of large-girth  $\nu$ -regular bipartite graphs. Using this sequence, it is possible to construct (see Algorithm 4) large-girth  $(c, d)$  regular Tanner graphs for arbitrary  $c, d$ .

However, it should be noted that there is no standard construction of  $(\nu, g)$ -cages that works for all  $\nu$  and  $g$ . Moreover, the structures of cages for  $\nu > 20$  or for  $g > 15$  are not currently known. Though the existence of cages proves the existence of large-girth regular LDPC codes, it does not result in a generalized construction algorithm for these codes. In the later sections, we will construct large-girth LDPC codes from large-girth graphs that are not necessarily cages.

### B. Existing Constructions for Tanner Graphs With Good Girth

A construction of regular large-girth LDPC codes was proposed by Gallager in his monograph [25, Appendix C]. The progressive edge growth (PEG) algorithm [26] constructs LDPC codes with a prescribed left (variable-node) degree distribution and rate. Though empirical evidence shows that PEG creates codes of good girth, we are unable to prove that they are large-girth codes. A PEG-like algorithm to construct *almost* regular large-girth graphs was proposed in [27]. A modification of this algorithm to create almost  $(c, d)$ -regular large-girth LDPC codes was published as the Almost Regular large-Girth (ARG) algorithm [28]. Though ARG creates large-girth LDPC codes, it cannot be used to create LDPC codes for a pre-defined DDP.

Based on the large-girth regular graph construction in [29], [30], Kim, et al. [31] constructed large-girth regular LDPC codes of rate  $1/q$ , for any prime power  $q$ . In other work, Margulis [32] constructed  $2r$ -regular large-girth graphs and based on his idea, Rosenthal and Vontobel [33] constructed large-girth  $(3, 6)$ -regular LDPC codes using the algebraic structure behind the construction of Ramanujan graphs proposed by [15].

It can be noted that the above constructions produce only large-girth LDPC codes of specific rates and specific (regular) degree distributions. On the other hand, the construction described in the next section produces large-girth graphs of arbitrary rates and degree distributions.

### C. Proposed Construction of Large-Girth Graphs

The adjacency matrix of a simple graph with  $n$  vertices is an  $n \times n$  matrix  $[a_{i,j}]$  such that  $a_{i,j} = 1$  whenever vertices  $i$  and  $j$  are adjacent, and  $a_{i,j} = 0$  otherwise. We consider the eigenvalues of the adjacency matrix. For a  $k$ -regular graph, any eigenvalue  $\mu$  is such that  $|\mu| \leq k$ . A Ramanujan graph is a  $k$ -regular graph such that if  $\mu$  is an eigenvalue and  $|\mu| \neq k$ , then  $|\mu| \leq 2\sqrt{k-1}$ . For a detailed discussion of Ramanujan graphs, see the book Davidoff, et al. [34].

Lubotzky, et al. [15] provided a Cayley graph based construction of certain Ramanujan graphs. For primes  $p$  and  $q$ , they construct a family of graphs  $X^{p,q}$  with the following properties.

**Theorem 5** ([34, Thm. 4.2.2]). *Let  $p, q$  be distinct, odd primes, with  $q > 2\sqrt{p}$ . The graphs  $X^{p,q}$  are  $(p+1)$ -regular graphs that are connected and Ramanujan. Moreover,*

- 1) *If  $p$  is a quadratic residue modulo  $q$ , then  $X^{p,q}$  is a non-bipartite graph with  $\frac{q(q^2-1)}{2}$  vertices, satisfying the girth estimate*

$$g(X^{p,q}) \geq 2 \log_p q$$

- 2) *If  $p$  is a quadratic non-residue modulo  $q$ , then  $X^{p,q}$  is a bipartite graph with  $q(q^2-1)$  vertices, satisfying*

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

For our purposes, we will not be using the Ramanujan property of the graphs  $X^{p,q}$ ; we will merely use the above lower bounds on the girth.

When  $p$  is a quadratic residue modulo  $q$ , we can use the construction in Algorithm 1 to generate a bipartite  $(p+1)$ -regular graph in  $q(q^2-1)$  vertices with girth at least  $2 \log_p q$ . Using Algorithm 1, we now have the following corollary.

**Corollary 6.** *Given a prime  $p$ , for any  $n \in \mathbb{N}$  it is possible to construct a  $(p+1)$ -regular bipartite graph in  $q(q^2-1)$  vertices with girth at least  $2 \log_p q$  for some prime  $q \geq n$  based on the construction of Lubotzky, et al.*

We would like to construct  $k$ -regular bipartite graphs of large girth where  $k$  is some arbitrary natural number, i.e., it is not necessarily the successor of a prime number. We do this as follows. We first find an integer  $s$  such that  $sk-1$  is a prime number, say  $p$ . The existence of  $s$  (and  $p$ ) is guaranteed by the following.

**Theorem 7** (Dirichlet's Theorem on Arithmetic Progressions [35, Chapter 7]). *Given two positive integers  $a, b$  that are relatively prime, i.e.,  $\gcd\{a, b\} = 1$ , the sequence  $(an+b)_{n \in \mathbb{N}}$  contains an infinite number of primes.*

**Corollary 8.** *Given any positive integer  $k$ , it is always possible to find  $s \in \mathbb{N}$  such that  $sk-1$  is a prime. Moreover, there are infinite such  $s$ .*

*Proof:* It can easily be seen that  $\gcd\{k, k-1\} = 1$ . Therefore, there are infinite prime numbers of the form  $rk + (k-1)$ , where  $r \in \mathbb{N}$ . Therefore, there are infinite prime numbers of the form  $sk-1$ , where  $s \in \mathbb{N}$ . ■

Now, we know that for any arbitrary natural number  $k$ , we can create a family of  $sk$ -regular graphs of large girth for some natural number  $s$ . Using this family, we can create a family of large-girth  $k$ -regular graphs by using Algorithm 2.

In this paper, we will only consider the creation of equal sized partitions  $N_i$  in Step 1 of Algorithm 2. Under this restriction, the partitioning of  $N(v)$  can be done in two different ways.

- **Deterministic version.** We assume that the edges in the graph  $G$  are in some simple ordering  $(e_1, e_2, \dots, e_M)$ . If

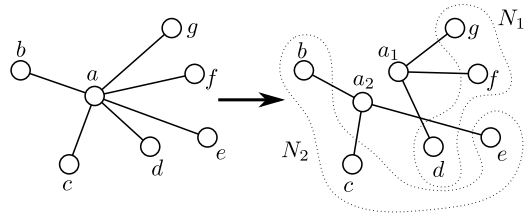


Fig. 4. An illustration of Algorithm 2 to split a vertex.

---

**Algorithm 2** Splitting a vertex into vertices of smaller degrees.

---

- 1: Given a vertex  $v$  in a graph  $G$ , we partition the set of all its neighbours  $N(v)$  into  $N_1, N_2, \dots, N_k$ .
  - 2: We create a new graph  $H$  by deleting  $v$  from  $G$  and adding new vertices  $v_1, v_2, \dots, v_k$  and connecting  $v_i$  to the vertices in  $N_i$  for all  $i$ .
- 

$N(v) = \{e_{i_1}, e_{i_2}, \dots, e_{i_{j_k}}\}$  with  $i_1 \leq i_2 \leq \dots \leq i_j$ , let

$$N_1 = \{e_{i_1}, e_{i_2}, \dots, e_{i_k}\}$$

$$N_2 = \{e_{i_{k+1}}, e_{i_{k+2}}, \dots, e_{i_{2k}}\}$$

and so on.

- **Random version.** The partitioning of  $N(v)$  is done in a random fashion.

Though the girth properties of the graphs obtained by the deterministic and the random versions of Algorithm 2 are similar, it is easier to count graphs of a particular configuration when we use the deterministic version than when we use the random version.

**Lemma 9.** *Given a graph  $G$ , if  $H$  is a graph obtained by splitting an arbitrary vertex of  $G$  according to either version of Algorithm 2, then  $g(H) \geq g(G)$ .*

*Proof:* Suppose  $H$  does not have any cycles. In this case,  $g(H) = \infty$  and the lemma is true.

We are now left with the case where  $H$  has cycles. Consider any cycle  $C$  in  $H$ . Let  $v$  be the vertex of  $G$  that is being split and let  $V_{\text{new}} = \{v_1, v_2, \dots, v_k\}$  be the set of new vertices created. By traversing along  $C$  and identifying vertices  $v_i$  with  $v$ , we will get a closed walk  $W$  in  $G$ . We show that  $W$  contains a cycle.

If  $C$  has less than two vertices from the set  $V_{\text{new}}$ , then  $W$  is a cycle and we are done. Otherwise,  $C$  has at least two of these vertices. We can pick  $v_i, v_j$  such that while traversing from one to the other along  $C$ , we don't encounter any other vertices from  $V_{\text{new}}$ . Let this path (excluding  $v_i, v_j$ ) be denoted by  $P$ . Since  $v_i$  and  $v_j$  are not adjacent and  $N(v_i) \cap N(v_j) = \emptyset$ , this path has at least two vertices. Therefore,  $vPv$  is a cycle  $C^*$  in  $W$  that is smaller than  $C$ . Since  $G$  contains the cycle  $C^*$ , we have

$$\text{length}(C) \geq \text{length}(C^*) \geq g(G)$$

which shows that  $g(H) \geq g(G)$ . ■

Note that Algorithm 2 can sometimes create a disconnected graph. That is,  $H$  may be disconnected even if  $G$  is connected. However, we can see that Lemma 9 is valid regardless of any disconnections introduced by node splitting. Furthermore, the

proof of our main theorem relies only on Lemma 9 and is valid even though some of the Tanner graphs in our ensemble may be disconnected.

---

**Algorithm 3** Constructing large-girth  $k$ -regular bipartite graphs.

---

- 1: Given a positive integer  $k$ , find the smallest solution for  $s \in \mathbb{N}$  such that  $sk - 1$  is a prime. The existence of  $s$  is guaranteed by Dirichlet's Theorem on Arithmetic Progressions. Denote  $sk - 1$  by  $p$ .
  - 2: Pick a sequence of primes greater than  $2\sqrt{p}$ . For each such prime  $q$ , generate the graph  $X^{p,q}$  described in [15].
  - 3: If  $p$  is a quadratic residue modulo  $q$ , then  $G = B(X^{p,q})$ . Otherwise,  $G = X^{p,q}$ . In either case,  $G$  is an  $(sk)$ -regular bipartite graph on  $q(q^2 - 1)$  vertices and  $g(G) \geq 2 \log_p q$ .
  - 4: Split each vertex of  $G$  successively into  $s$  vertices of degree  $k$  according to either version of Algorithm 2. Call this new graph  $H$ .
  - 5:  $H$  is a  $k$ -regular bipartite graph with  $sq(q^2 - 1)$  vertices and  $g(H) \geq g(G) \geq 2 \log_p q$ .
- 

---

**Algorithm 4** Constructing large-girth  $(c, d)$ -regular bipartite graphs.

---

- 1: Let  $k = \text{LCM}\{c, d\}$ . Construct a sequence of  $k$ -regular bipartite graphs of large girth according to Algorithm 3.
  - 2: Given a  $k$ -regular bipartite graph  $G$  with  $sq(q^2 - 1)$  vertices and  $g(G) \geq 2 \log_p q$ , let  $(V_s, V_c)$  be the bipartition of the vertices. We have  $|V_s| = |V_c| = \frac{sq(q^2 - 1)}{2}$ .
  - 3: Split each vertex in  $V_s$  into  $k/c$  new vertices of degree  $c$  each according to either version of Algorithm 2 to get  $\frac{k}{c} \frac{sq(q^2 - 1)}{2}$  left vertices of degree  $c$ .
  - 4: Split each vertex in  $V_c$  into  $k/d$  new vertices of degree  $d$  each according to either version of Algorithm 2 to get  $\frac{k}{d} \frac{sq(q^2 - 1)}{2}$  right vertices of degree  $d$ . The resultant graph  $H$  is a  $(c, d)$ -regular bipartite graph with  $g(H) \geq 2 \log_p q$ .
- 

#### D. Strong Secrecy Using Irregular LDPC Codes Based on Ramanujan Graphs

For a given DDP  $(\lambda, \rho)$ , we can create a sequence of large-girth  $(\lambda, \rho)$ -irregular LDPC codes  $(C_n)$  of increasing block-length  $n$  using Algorithm 5. We denote the large-girth graphs associated with  $C_n$  by  $\mathcal{R}_n$ .

**Theorem 10.** *For a given DDP  $(\lambda, \rho)$  with minimum left degree  $l_{\min} \geq 3$ , the sequence of large-girth  $(\lambda, \rho)$ -irregular LDPC codes  $(C_n)$  created using Algorithm 5 is such that whenever  $\epsilon < \epsilon_{\text{th}}$  we have*

$$\mathbb{E}P_b^{\text{MP}}(C_n, \epsilon) = \mathcal{O}(\exp(-c_1 n^{\epsilon_2})) \quad (4)$$

for some positive constants  $c_1, \epsilon_2$ .

*Proof:* See Appendix B ■

The asymptotic decay of the bit-error probability achieved by the codes in Thm. 10 is faster than the inverse cubic decay

---

**Algorithm 5** Constructing large-girth  $(\lambda, \rho)$  irregular bipartite graphs.

---

- 1: Let  $k$  be the least common multiple of all the left and right degrees. Let  $a$  be the smallest positive integer such that  $a\lambda_i, a\rho_j \in \mathbb{N}$  for all  $i, j$ .
  - 2: Let  $s$  be the smallest natural number such that  $sak - 1$  is a prime number. Call this prime number  $p$ . Choose an arbitrarily prime  $q > 2\sqrt{p}$ . Construct an  $(ak)$ -regular bipartite graph  $G_0$  according to Algorithm 3.  $G_0$  has  $sq(q^2 - 1)$  vertices and  $g(G_0) \geq 2 \log_p q$ .
  - 3: Split each vertex of  $G_0$  into  $a$  vertices of degree  $k$  by successively applying Algorithm 2 (either version) and denote the resulting  $k$ -regular bipartite graph by  $G$ .  $G$  has  $n_0$  vertices on the left and  $n_0$  vertices on the right, where  $n_0 = \frac{asq(q^2 - 1)}{2}$ , and  $g(G) \geq 2 \log_p q$ .
  - 4: Let  $(v_1, v_2, \dots, v_{n_0})$  be some ordering of the ‘‘left’’ vertices in  $G$  and let  $(c_1, c_2, \dots, c_{n_0})$  be some ordering of the ‘‘right’’ vertices in  $G$ . Also, let  $(e_1, e_2, \dots, e_{n_0 k})$  be some ordering of the edges in  $G$ .
  - 5: Let  $\sigma$  and  $\pi$  be two randomly chosen permutation functions over the set  $\{1, 2, \dots, n_0\}$ .
  - 6: Consider the ordered set  $(v'_1, v'_2, \dots, v'_{n_0})$ , where  $v'_i = v_{\sigma(i)}$ . In this ordered set,
    - split the first  $n_0 \lambda_{l_{\min}}$  vertices into  $n_0 k \lambda_{l_{\min}} / l_{\min}$  vertices of degree  $l_{\min}$ ,
    - split the next  $n_0 \lambda_{l_{\min} + 1}$  vertices into  $n_0 k \lambda_{l_{\min} + 1} / (l_{\min} + 1)$  vertices of degree  $l_{\min} + 1$ ,
    - ...
    - split the last  $n_0 \lambda_{l_{\max}}$  vertices into  $n_0 k \lambda_{l_{\max}} / l_{\max}$  vertices of degree  $l_{\max}$ .
- In the above, we split the vertices according to the deterministic version of Algorithm 2.
- 7: Do a similar operation for the check nodes using the ordered set  $(c'_1, c'_2, \dots, c'_{n_0})$ , where  $c'_j = c_{\pi(j)}$ , and the distribution  $\rho$ . The resulting graph  $H$  is a  $(\lambda, \rho)$  irregular bipartite graph with
 
$$n = \frac{aksq(q^2 - 1)}{2} \int_0^1 \lambda dx$$
 vertices and girth at least  $2 \log_p q$ .
- 

required for strong secrecy. This directly implies that the duals of our Ramanujan graph LDPC codes achieve strong secrecy on the BEWC under the coset coding scheme.

#### E. Discussion

For a given DDP  $(\lambda, \rho)$ , we have constructed a sequence  $(C_n)$  of large-girth LDPC codes based on Ramanujan graphs. For minimum left degree at least three, we showed that for  $\epsilon < \epsilon_{\text{th}}$ , we have

$$\mathbb{E}P_b^{\text{MP}}(C_n, \epsilon) = \mathcal{O}(\exp(-\beta n^a \log(l_{\min} - 1)))$$

By Corollary 2, the dual sequence  $(C_n^\perp)$  achieves strong secrecy on BEWC( $\xi$ ) for  $\xi > 1 - \epsilon_{\text{th}}$ .

1) *Difference Between Regular and Irregular Codes:* For any large-girth regular LDPC code sequence  $(C_n)$ , we have

$$\mathbb{P}_b^{\text{MP}}(C_n, \epsilon) = \mathcal{O}\left(\exp(-\beta n^a \log(l_{\min}-1))\right)$$

for  $\epsilon < \epsilon_{\text{th}}$ . This means that the dual sequence achieves strong secrecy on BEWC( $\xi$ ) for  $\xi > 1 - \epsilon_{\text{th}}$ .

For irregular codes *constructed by Algorithm 5*, we have shown that

$$\mathbb{E}\left(\mathbb{P}_b^{\text{MP}}(C_n, \epsilon)\right) = \mathcal{O}\left(\exp(-\beta n^a \log(l_{\min}-1))\right)$$

for  $\epsilon < \epsilon_{\text{th}}$ . This means that *on an average*, the dual sequence achieves strong secrecy on BEWC( $\xi$ ) for  $\xi > 1 - \epsilon_{\text{th}}$ . Moreover, most codes in the dual sequence are concentrated around this average strong secrecy property. This is because the Markov inequality guarantees that, with probability tending to 1, any LDPC code from the ensemble will have probability of bit error smaller than  $\frac{1}{n^k}$  (for any positive integer  $k$ ).

Our result for regular LDPC codes is stronger than that of irregular LDPC codes. However, irregular LDPC codes are important because they have  $1 - \epsilon_{\text{th}}$  very close to their rate. Therefore, irregular codes are instrumental in achieving a secrecy rate very close to the BEWC secrecy capacity.

2) *Gap Between Achievable Region and Secrecy Capacity:* For a secret information rate  $R$ , we are interested in the minimum value of Eve's erasure probability  $\epsilon$  for which we can ensure strong secrecy over the BEWC using our scheme. Since our proof works only for  $l_{\min} \geq 3$ , this involves finding an optimal DDP of rate  $R$  and  $l_{\min} \geq 3$  for which the BEC threshold  $\epsilon_{\text{th}}$  is as high as possible. It can be noted that  $\epsilon_{\text{th}} < 1 - R$ . Most of the capacity achieving DDP sequences require  $l_{\min} = 2$  (e.g., the tornado sequence and the right regular sequence in [36]). Therefore, there is a small gap between the strong secrecy rate achievable by our technique and the secrecy capacity of the BEWC.

For example, when we performed a search using the LDP-COPT online tool [37] for  $R = 0.5$  and  $l_{\min} \geq 3$ , we found that the maximum value of  $\epsilon_{\text{th}} = 0.4619$  is achieved by the DDP

$$\begin{aligned} \lambda(x) &= 0.9043388x^2 + 0.03300419x^{16} + 0.01434268x^{17} \\ &\quad + 0.03535427x^{18} + 0.01296008x^{99} \\ \rho(x) &= x^{10} \end{aligned}$$

This means that the duals of the LDPC codes constructed using Algorithm 5 will achieve a strong secrecy rate of 0.5 over BEWC( $\epsilon$ ) for all  $\epsilon > 0.5381$ . Note that for  $\epsilon$  close to 0.5381, the secrecy capacity of the BEWC is close to 0.5381. Our coding scheme will achieve a secrecy rate of 0.5 over this channel, which is 7% less than the secrecy capacity.

## V. CONCLUSION AND FUTURE DIRECTIONS

In this work, we have constructed LDPC codes whose girth increases logarithmically in block-length using Ramanujan graphs. In contrast to existing large-girth constructions, our construction works for arbitrary irregular degree distribution pairs. To our knowledge, this is the first such construction. We have shown that the duals of these LDPC codes achieve strong

secrecy on the binary erasure wiretap channel (BEWC), when their minimum left degree is at least three. To achieve secrecy capacity on the BEWC, we require LDPC code ensembles with degree-2 variable nodes. Since our current proof does not apply to these codes, we must look for new techniques to analyze them. A multiedge-type construction, similar to the approach of [8], [9], might be required to achieve secrecy capacity on the BEWC.

In addition, the Ramanujan graph ensemble is interesting in the general area of LDPC coding even without the secrecy application. In particular, one can show (through the relationship between the girth and the stopping distance) that the minimum distance of these codes grows at least as  $n^b$ , for some  $b$  such that  $0 < b < 1$ . Further properties of this construction could be explored in future work.

## APPENDIX A PROOF OF LEMMA 3

For any  $x \in [0, 1]$ , we have

$$\begin{aligned} (1-x)^{d-1} &\geq 1 - (d-1)x, \quad \forall d \in \mathbb{N} \\ \Rightarrow \rho(1-x) &= \sum_{d=2}^{r_{\max}} \rho_d (1-x)^{d-1} \\ &\geq \sum_{d=2}^{r_{\max}} (1 - (d-1)x) \rho_d \\ &= 1 - (r_{\text{avg}} - 1)x \\ \Rightarrow 1 - \rho(1-x) &\leq (r_{\text{avg}} - 1)x \end{aligned}$$

where  $r_{\text{avg}}$  is the average check-node degree and  $r_{\max}$  is the maximum check-node degree. For  $0 \leq (r_{\text{avg}} - 1)x \leq 1$ ,

$$\begin{aligned} f(\epsilon, x) &= \epsilon \lambda(1 - \rho(1-x)) \\ &\stackrel{\text{a}}{\leq} \epsilon \lambda((r_{\text{avg}} - 1)x) \\ &= \epsilon \sum_{i=l_{\min}}^{l_{\max}} \lambda_i ((r_{\text{avg}} - 1)x)^{i-1} \\ &\stackrel{\text{b}}{\leq} \epsilon \sum_{i=l_{\min}}^{l_{\max}} \lambda_i ((r_{\text{avg}} - 1)x)^{l_{\min}-1} \\ \Rightarrow f(\epsilon, x) &\leq \epsilon ((r_{\text{avg}} - 1)x)^{l_{\min}-1} =: g(\epsilon, x) \quad (5) \end{aligned}$$

Note that (a) follows from the monotonicity of  $\lambda(x)$ , and (b) follows from the given condition  $0 \leq (r_{\text{avg}} - 1)x \leq 1$ . To make the notation easier, let us denote  $A = \epsilon (r_{\text{avg}} - 1)^{l_{\min}-1}$ . Since we are operating in the region  $\epsilon < \epsilon_{\text{th}}$  where  $(x_t)$  converges to zero, there exists an  $R$  such that  $Ax_R^{l_{\min}-2} \leq 1$  and  $(r_{\text{avg}} - 1)x_R \leq 1$ . The first inequality will be used later in the proof.

Let us construct a sequence  $z_{R+i+1} = g(\epsilon, z_{R+i})$  with  $z_R = x_R$ . It is immaterial what  $z_i$  takes when  $i < R$ . We then claim that  $x_{R+i} \leq z_{R+i}$  for any non-negative integer  $i$ . We can prove this by induction. The base case is when  $i = 0$  and it is true by our choice of  $z_R$ . Assuming the claim is true for some integer  $i \geq 0$ , we have

$$x_{R+i+1} = f(\epsilon, x_{R+i}) \leq g(\epsilon, x_{R+i}) \leq g(\epsilon, z_{R+i}) = z_{R+i+1}$$



The first inequality is due to Eqn. (5) and the second inequality is due to the monotonicity of  $g$  and the induction hypothesis. This proves the claim.

We have,

$$\begin{aligned}
z_{R+1} &= Az_R^{l_{\min}-1} \\
z_{R+i} &= A^{1+(l_{\min}-1)+(l_{\min}-1)^2+\dots+(l_{\min}-1)^{i-1}} z_R^{(l_{\min}-1)^i} \\
&= A^{\frac{(l_{\min}-1)^i-1}{l_{\min}-2}} z_R^{(l_{\min}-1)^i} \\
&= A^{\frac{-1}{l_{\min}-2}} \left( A^{\frac{1}{l_{\min}-2}} x_R \right)^{(l_{\min}-1)^i} \\
&= A^{\frac{-1}{l_{\min}-2}} \exp \left( (l_{\min}-1)^i \left( \frac{\log A}{l_{\min}-2} + \log x_R \right) \right) \\
&= A^{\frac{-1}{l_{\min}-2}} \exp \left( -\alpha_R (l_{\min}-1)^i \right)
\end{aligned}$$

Due to our choice of  $R$ ,  $\alpha_R \triangleq \frac{-1}{l_{\min}-2} \log A - \log x_R$  is positive. In fact, we can choose  $R$  arbitrarily large, making  $x_R$  arbitrarily small and  $\alpha_R$  arbitrarily large. For  $t \geq R$ , we have

$$\begin{aligned}
x_t &\leq z_t \\
&= A^{\frac{-1}{l_{\min}-2}} \exp \left( -\alpha_R (l_{\min}-1)^{t-R} \right) \\
&= A^{\frac{-1}{l_{\min}-2}} \exp \left( -\frac{\alpha_R}{(l_{\min}-1)^R} (l_{\min}-1)^t \right) \\
&= A^{\frac{-1}{l_{\min}-2}} \exp \left( -\beta (l_{\min}-1)^t \right)
\end{aligned}$$

Note that  $\beta \triangleq \frac{\alpha_R}{(l_{\min}-1)^R} > 0$ . Therefore, we have

$$x_t = \mathcal{O} \left( \exp(-\beta (l_{\min}-1)^t) \right) \quad \text{as } t \rightarrow \infty$$

To prove the second half, we note that for  $x \in [0, 1]$

$$\begin{aligned}
L(x) &= \sum L_i x^i \leq \sum L_i x^{i-1} \\
&= \frac{1}{\int_0^1 \lambda(x) dx} \sum_{i=l_{\min}}^{l_{\max}} \frac{\lambda_i}{i} x^{i-1} \\
&\leq \frac{1}{l_{\min} \int_0^1 \lambda(x) dx} \lambda(x) \\
\Rightarrow y_t &\leq \frac{1}{l_{\min} \int_0^1 \lambda(x) dx} x_t \\
\Rightarrow y_t &= \mathcal{O} \left( \exp(-\beta (l_{\min}-1)^t) \right)
\end{aligned}$$

## APPENDIX B PROOF OF THM. 10

The only sources of randomness in Algorithm 5 for a given large-girth graph  $G$  (at the end of Step 3) are the permutation functions  $\sigma$  and  $\pi$ . The probability distribution of  $\mathcal{R}_n$  given  $G$  is easier to analyze than that of  $\mathcal{R}_n$  when  $G$  is not specified. Clearly, (4) is true whenever

$$\mathbb{E} \left( P_b^{\text{MP}}(C_n, \epsilon) | G \right) = \mathcal{O} \left( \exp(-c_1 n^{c_2}) \right) \quad (6)$$

is true *uniformly* for all possible  $G$  in Step 3 of Algorithm 5.

Note that  $P_b^{\text{MP}}(C_n, \epsilon)$  denotes the probability of bit-error after infinite iterations of the MP algorithm (or equivalently, when a stopping set is encountered). This probability is clearly

less than the probability of bit-error after a finite number of iterations. Therefore (6) is true whenever

$$\mathbb{E} \left( P_b^{\text{MP}}(C_n, \epsilon, t(n)) | G \right) \leq A(n) = \mathcal{O} \left( \exp(-c_1 n^{c_2}) \right) \quad (7)$$

is true for some function  $t(n)$ . The role played by the quantity  $A(n)$  is to ensure that we are able to upper bound the left hand side uniformly in  $G$ . We pick  $t(n) = a \log n$ , where  $a > 0$  is such that  $g(\mathcal{R}_n) \geq 4a \log n + 2$ . We know that  $a$  exists because of the large-girth property of  $\mathcal{R}_n$ . Let  $a_{\max}$  be the maximum possible value for  $a$ .

**Proposition 11.** *For any  $\delta \in (0, 1)$ , there exists a natural number  $N$  such that for all  $n \geq N$  we have*

$$\mathbb{E} \left( P_b^{\text{MP}}(C_n, \epsilon, t(n)) | G \right) \leq \frac{1}{1-\delta} y_{t(n)}(\epsilon) \quad (8)$$

where  $y_{t(n)}(\epsilon)$  is the quantity defined in Lemma 3.

We know from Lemma 3 that

$$\begin{aligned}
y_{t(n)}(\epsilon) &= \mathcal{O} \left( \exp(-\beta (l_{\min}-1)^{t(n)}) \right) \\
&= \mathcal{O} \left( \exp(-\beta (l_{\min}-1)^{a \log n}) \right) \\
&= \mathcal{O} \left( \exp(-\beta n^{a \log(l_{\min}-1)}) \right)
\end{aligned}$$

The above equation, along with Proposition 11, completes the proof of the theorem.

*Proof of Proposition 11:* Consider the computation graph  $\check{C}_t$  of  $\mathcal{R}_n$  (we write  $t$  for  $t(n)$  to make the notation less cumbersome). Clearly,  $\mathbb{P}(\check{C}_t = T) > 0$  if and only if  $\mathbb{P}(\check{\mathcal{T}}_t = T) > 0$ .

Let  $T$  be any valid level- $t$  tree in the sense that  $\mathbb{P}(\check{\mathcal{T}}_t = T) > 0$ . Let  $P_e(T, \epsilon)$  be the probability that the root node of  $T$  is in error when the tree code associated with  $T$  is transmitted over BEC( $\epsilon$ ) and decoded with  $t$  iterations of the MP decoder. Note the following two equations

$$\begin{aligned}
y_t(\epsilon) &= \sum \mathbb{P}(\check{\mathcal{T}}_t = T) P_e(T, \epsilon) \\
\mathbb{E} \left( P_b^{\text{MP}}(C_n, \epsilon, t) | G \right) &= \sum \mathbb{P}(\check{C}_t = T | G) P_e(T, \epsilon)
\end{aligned}$$

From the above, we can see that the proof is complete once we show that for some natural number  $N$ , we have

$$\mathbb{P}(\check{C}_t = T | G) \leq \frac{1}{1-\delta} \mathbb{P}(\check{\mathcal{T}}_t = T)$$

for all  $n \geq N$ .

Let  $T$  be a valid level- $t$  tree with  $i_0$  being the degree of the root node. Let this tree have  $p_i$  variable nodes of degree  $i$  (including the root node, but excluding the leaf nodes) and  $q_j$  check nodes of degree  $j$ . We have

$$\mathbb{P}(\check{\mathcal{T}}_t = T) = L_{i_0} \lambda_{i_0}^{p_{i_0}-1} \prod_{i=3, i \neq i_0}^{l_{\max}} \lambda_i^{p_i} \prod_{j=2}^{r_{\max}} \rho_j^{q_j} \quad (9)$$

Now, consider  $\check{C}_t$ . The probability that the root node  $v$  has degree  $i_0$  is clearly  $L_{i_0}$ . The  $i_0$  edges incident with  $v$  in  $\mathcal{R}_n$  will correspond to  $i_0$  edges in  $G$  incident with  $u$ , the parent node of  $v$ . Let  $b(1), b(2), \dots, b(i_0)$  be the  $i_0$  neighbours of  $u$  in  $G$  corresponding to those edges. Let  $c(1), c(2), \dots, c(i_0)$  be the daughter nodes in  $\mathcal{R}_n$  corresponding to the same edges. The number of ways of choosing the permutation function  $\pi$

such that node  $c(1)$  has degree  $j$  is equal to the number of ways of putting  $b(1)$  into a slot that corresponds to degree  $j$ , which is  $n_0 \rho_j$ . Note that these slots are numbered. Here,  $n_0 = 2n / \left(k \int_0^1 \lambda dx\right)$  is the number of left (right) vertices in  $G$ , where  $k$  is the LCM of all the degrees in  $(\lambda, \rho)$ .

In general, whenever  $T$  is a valid level- $t$  tree, we have

$$\begin{aligned} \mathbb{P}(\mathring{\mathcal{C}}_t = T) &= L_{i_0} \binom{n_0 \lambda_{i_0} - 1}{p_{i_0} - 1} (p_{i_0} - 1)! \\ &\times \frac{(n_0 - 1 - \sum_{i=2}^{l_{\max}} p_i)!}{n_0!} \prod_{\substack{i=l_{\min} \\ i \neq i_0}}^{l_{\max}} \binom{n_0 \lambda_i}{p_i} p_i! \\ &\times \frac{(n_0 - \sum_{j=r_{\min}}^{r_{\max}} q_j)!}{n_0!} \prod_{j=r_{\min}}^{r_{\max}} \binom{n_0 \rho_j}{q_j} q_j! \end{aligned} \quad (10)$$

We note the following inequality.

$$\begin{aligned} &\frac{(n_0 - 1 - \sum_{i=2}^{l_{\max}} p_i)! (n_0 - \sum_{j=r_{\min}}^{r_{\max}} q_j)!}{n_0! n_0!} \\ &< \frac{1}{(n_0 - \sum p_i)^{(\sum p_i) - 1} (n_0 - \sum q_j)^{\sum q_j}} \end{aligned} \quad (11)$$

We also see that

$$\begin{aligned} &L_{i_0} \binom{n_0 \lambda_{i_0} - 1}{p_{i_0} - 1} (p_{i_0} - 1)! \prod_{\substack{i=l_{\min} \\ i \neq i_0}}^{l_{\max}} \binom{n_0 \lambda_i}{p_i} p_i! \prod_{j=r_{\min}}^{r_{\max}} \binom{n_0 \rho_j}{q_j} q_j! \\ &< L_{i_0} (n_0 \lambda_{i_0})^{p_{i_0} - 1} \prod_{\substack{i=l_{\min} \\ i \neq i_0}}^{l_{\max}} (n_0 \lambda_i)^{p_i} \prod_{j=r_{\min}}^{r_{\max}} (n_0 \rho_j)^{q_j} \\ &= n_0^{(\sum p_i + \sum q_j) - 1} L_{i_0} \lambda_{i_0}^{p_{i_0} - 1} \prod_{\substack{i=l_{\min} \\ i \neq i_0}}^{l_{\max}} \lambda_i^{p_i} \prod_{j=r_{\min}}^{r_{\max}} \rho_j^{q_j} \\ &= n_0^{(\sum p_i + \sum q_j) - 1} \mathbb{P}(\mathring{\mathcal{T}}_t = T) \end{aligned} \quad (12)$$

Substituting (11) and (12) in (10), we get

$$\mathbb{P}(\mathring{\mathcal{C}}_t = T) < \frac{\mathbb{P}(\mathring{\mathcal{T}}_t = T)}{\left(1 - \frac{\sum p_i}{n_0}\right)^{(\sum p_i) - 1} \left(1 - \frac{\sum q_j}{n_0}\right)^{\sum q_j}} \quad (13)$$

The proof is complete once we show that

$$\left(1 - \frac{\sum p_i}{n_0}\right)^{(\sum p_i) - 1} \left(1 - \frac{\sum q_j}{n_0}\right)^{\sum q_j} \rightarrow 1 \quad \text{as } n \rightarrow \infty \quad (14)$$

First, we note that  $\sum p_i$  and  $\sum q_j$  grow exponentially in  $t$ . This means that there exist constants  $\alpha_1, \alpha_2, \beta_1, \beta_2 > 0$  such that

$$\alpha_1 n^{\beta_1} < \sum p_i, \sum q_j < \alpha_2 n^{\beta_2} \quad (15)$$

We have

$$\begin{aligned} 1 &> \left(1 - \frac{\sum p_i}{n_0}\right)^{(\sum p_i) - 1} \left(1 - \frac{\sum q_j}{n_0}\right)^{\sum q_j} \\ &> \left(1 - \frac{\alpha_2 n^{\beta_2}}{n_0}\right)^{\alpha_2 n^{\beta_2} - 1} \left(1 - \frac{\alpha_2 n^{\beta_2}}{n_0}\right)^{\alpha_2 n^{\beta_2}} \\ &= \left(1 - \frac{\alpha_2 n^{\beta_2}}{n_0}\right)^{2\alpha_2 n^{\beta_2} - 1} \end{aligned}$$

The proof is complete once we show that

$$\left(1 - \frac{\frac{1}{2}\alpha_2 k \int_0^1 \lambda dx}{n^{1-\beta_2}}\right)^{n^{\beta_2}} \rightarrow 1 \quad (16)$$

For this, we pick the constant  $a \in (0, a_{\max}]$  small enough so that  $\beta_2 < 0.5$ . Observe that for any  $\theta > 1$  and  $\alpha > 0$ , we have

$$\lim_{n \rightarrow \infty} \left(1 - \frac{\alpha}{n^\theta}\right)^n = 1 \quad (17)$$

Substituting  $m = n^{\beta_2}$  in the left hand side of (16), we have

$$\left(1 - \frac{\frac{1}{2}\alpha_2 k \int_0^1 \lambda dx}{m^{(1-\beta_2)/\beta_2}}\right)^m$$

which goes to 1 as  $m \rightarrow \infty$ .  $\blacksquare$

## REFERENCES

- [1] C. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal*, vol. 28, pp. 656–715, Oct. 1949.
- [2] A. D. Wyner, "The wire-tap channel," *Bell System Technical Journal*, vol. 54, no. 8, pp. 1355–1367, Oct. 1975.
- [3] I. Csiszár and J. Körner, "Broadcast channels with confidential messages," *IEEE Trans. Inf. Theory*, vol. 24, no. 3, pp. 339–348, May 1978.
- [4] U. M. Maurer and S. Wolf, "Information-theoretic key agreement: From weak to strong secrecy for free," in *Advances in Cryptology - Eurocrypt 2000*, Lecture Notes in Computer Science. B. Preneel, 2000, p. 351.
- [5] I. Csiszár, "Almost independence and secrecy capacity," *Problems of Information Transmission*, vol. 32, no. 1, pp. 40–47, Jan.–Mar. 1996.
- [6] R. Liu, Y. Liang, H. V. Poor, and P. Spasojević, "Secure nested codes for type II wiretap channels," in *Proceedings of IEEE Information Theory Workshop*, Lake Tahoe, California, USA, Sep. 2007, pp. 337–342.
- [7] L. H. Ozarow and A. D. Wyner, "Wire-tap channel II," *Bell Labs Tech. J.*, vol. 63, no. 10, pp. 2135–2157, Dec. 1984.
- [8] V. Rathi, M. Andersson, R. Thobaben, J. Kliewer, and M. Skoglund, "Two edge type LDPC codes for the wiretap channel," in *Signals, Systems and Computers, 2009 Conference Record of the Forty-Third Asilomar Conference on*, 2009, pp. 834–838.
- [9] —, "Performance analysis and design of two edge type LDPC codes for the BEC wiretap channel," 2010, submitted to *IEEE Transactions on Information Theory*. [Online]. Available: <http://arxiv.org/abs/1009.4610>
- [10] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.
- [11] C. H. Bennett, G. Brassard, C. Crépeau, and U. Maurer, "Generalized privacy amplification," *IEEE Trans. Inf. Theory*, vol. 41, no. 6, pp. 1915–1923, Nov. 1995.
- [12] A. Thangaraj, S. Dihidar, A. Calderbank, S. McLaughlin, and J.-M. Merolla, "Applications of LDPC codes to the wiretap channel," *Information Theory, IEEE Transactions on*, vol. 53, no. 8, pp. 2933–2945, Aug. 2007.
- [13] A. T. Suresh, A. Subramanian, A. Thangaraj, M. Bloch, and S. W. McLaughlin, "Strong secrecy for erasure wiretap channels," in *Information Theory Workshop, 2010. ITW 2010. IEEE*, Aug. 2010. [Online]. Available: <http://arxiv.org/abs/1004.5540>
- [14] A. Orlitsky, K. Viswanathan, and J. Zhang, "Stopping set distribution of LDPC code ensembles," *IEEE Transactions on Information Theory*, vol. 51, no. 3, pp. 929–953, Mar. 2005.
- [15] A. Lubotzky, R. Phillips, and P. Sarnak, "Ramanujan graphs," *Combinatorica*, vol. 8, no. 3, pp. 261–277, 1988.

- [16] H. Mahdaviifar and A. Vardy, "Achieving the secrecy capacity of wiretap channels using polar codes," in *Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on*, Jun. 2010, pp. 913–917.
- [17] E. Hof and S. Shamai, "Secrecy-achieving polar-coding," in *Information Theory Workshop, 2010. ITW 2010. IEEE*, 2010, to appear. [Online]. Available: <http://arxiv.org/abs/1005.2759>
- [18] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms (MIT Electrical Engineering and Computer Science)*. The MIT Press, 1990.
- [19] M. Lentmaier, D. Truhachev, K. Zigangirov, and D. Costello, "An analysis of the block error probability performance of iterative decoding," *Information Theory, IEEE Transactions on*, vol. 51, no. 11, pp. 3834–3855, Nov. 2005.
- [20] S. Korada and R. Urbanke, "Exchange of limits: Why iterative decoding works," in *Information Theory, 2008. ISIT 2008. IEEE International Symposium on*, 2008, pp. 285–289.
- [21] P. Wong, "Cages - a survey," *Journal of Graph Theory*, vol. 6, no. 1, pp. 1–22, 1982.
- [22] P. Erdos and H. Sachs, "Reguläre graphen gegebener taillenweite mit minimaler knotenzahl (Regular graphs with given girth and minimal number of nodes)," *Wiss. Z. Martin-Luther-Univ. Halle-Wittenberg Math. Natur. Reihe*, vol. 12, pp. 251–257, 1963.
- [23] N. Sauer, "Extremaleigenschaften regulärer graphen gegebener taillenweite (Extremal regular graphs of a given girth), I and II, sitzungsberichterreich," *Acad. Wiss. Math. Natur. Kl., SB II*, vol. 176, pp. 9–25, 1967.
- [24] B. Bollobas, *Extremal Graph Theory*. Dover Publications, 2004.
- [25] R. G. Gallager, *Low-Density Parity-Check Codes*. The MIT Press, 1963.
- [26] X.-Y. Hu, E. Eleftheriou, and D. Arnold, "Regular and irregular progressive edge-growth Tanner graphs," *Information Theory, IEEE Transactions on*, vol. 51, no. 1, pp. 386–398, Jan. 2005.
- [27] L. S. Chandran, "A high girth graph construction," *SIAM J. Discret. Math.*, vol. 16, no. 3, pp. 366–370, 2003.
- [28] K. M. Krishnan, R. Singh, L. S. Chandran, and P. Shankar, "A combinatorial family of near regular LDPC codes," in *Information Theory, 2007. ISIT 2007. IEEE International Symposium on*, Jun. 2007, pp. 761–765.
- [29] F. Lazebnik and V. A. Ustimenko, "Explicit construction of graphs with an arbitrary large girth and of large size," *Discrete Appl. Math.*, vol. 60, no. 1-3, pp. 275–284, 1995.
- [30] F. Lazebnik, V. Ustimenko, and A. Woldar, "A new series of dense graphs of high girth," *American Mathematical Society*, vol. 32, no. 1, 1995.
- [31] J.-L. Kim, U. Peled, I. Perepelitsa, V. Pless, and S. Friedland, "Explicit construction of families of LDPC codes with no 4-cycles," *Information Theory, IEEE Transactions on*, vol. 50, no. 10, pp. 2378–2388, Oct. 2004.
- [32] G. Margulis, "Explicit constructions of graphs without short cycles and low density codes," *Combinatorica*, vol. 2, no. 1, pp. 71–78, 1982.
- [33] J. Rosenthal and P. Vontobel, "Constructions of regular and irregular LDPC codes using Ramanujan graphs and ideas from Margulis," in *Information Theory, 2001. Proceedings. 2001 IEEE International Symposium on*, 2001, p. 4.
- [34] G. Davidoff, P. Sarnak, and A. Valette, *Elementary Number Theory, Group Theory and Ramanujan Graphs (London Mathematical Society Student Texts)*. Cambridge University Press, 2003.
- [35] T. M. Apostol, *Introduction to Analytic Number Theory (Undergraduate Texts in Mathematics)*. Springer, 1976.
- [36] P. Oswald and A. Shokrollahi, "Capacity-achieving sequences for the erasure channel," *Information Theory, IEEE Transactions on*, vol. 48, no. 12, pp. 3017–3028, Dec. 2002.
- [37] R. Urbanke and A. Amraoui. (2010, Jun.) LDPCOPT - a fast and accurate degree distribution optimizer for LDPC code ensembles. Online database. [Online]. Available: <http://ipgdemos.epfl.ch/ldpcopt/>