

Asymptotic Ensemble Enumerators for Protograph-Based Generalized LDPC Codes: Computational Complexity

Shadi Abu-Surra
University of Arizona
Email: shadia@ece.arizona.edu

William E. Ryan
University of Arizona
Email: ryan@ece.arizona.edu

Dariush Divsalar
Jet Propulsion Laboratory
Email: Dariush.Divsalar@jpl.nasa.gov

I. INTRODUCTION

In earlier work [1]–[3] we presented a method for finding ensemble weight enumerator for protograph-based generalized LDPC (G-LDPC) codes, and leveraged this method to find ensemble stopping set enumerator and ensemble trapping set enumerator. The method is conceptually simple, but when the dimensionality of the constraint nodes (number of their codewords) grows, it becomes difficult to handle the computational complexity, which rise while evaluating these enumerators. To deal with this difficulty, we posed a conjecture, which greatly reduce the computational complexity. Trails to proof this conjecture showed that the proof is a challenging problem. Also, proving it will strengthen the theory of enumerating protograph-based G-LDPC code ensembles. Which in turn helps in predicating the average performances for codes drawn from these ensembles.

In Section II we present a review of our method for finding finite and asymptotic weight enumerators for protograph-based G-LDPC code ensembles [1], [2]. Then, we present the conjecture in Section III with some examples.

II. WEIGHT ENUMERATORS FOR PROTOGRAPH-BASED G-LDPC CODE ENSEMBLES

Recall that a protograph [4] is a relatively small bipartite graph, containing variable nodes (VNs) and constraint nodes (CNs), from which a larger graph can be obtained by a copy-and-permute procedure: the protograph is copied N times, and then the edges of the individual replicas are permuted among the replicas to obtain a single, large graph. The copy-and-permute process can be simply represented by replacing each node with a vector of nodes of the same type and replacing each edge with a bundle of (permuted) edges of the same type. This “vectorized” protograph is depicted in Fig. 1.

Following the notation of [1], [2], [5], consider an LDPC protograph, $G = (V, C, E)$, where $V = \{v_1, v_2, \dots, v_{n_v}\}$ is the set of n_v VNs, $C = \{c_1, c_2, \dots, c_{n_c}\}$ is the set of n_c CNs, and E is the set of edges. Denote by q_{v_i} (q_{c_j}) the degree of variable (constraint) node v_i (c_j). Now consider the LDPC code constructed from a protograph G by making N replicas of G and using uniform interleavers, each of size N , to permute the edges among the replicas of the protograph. In order to exploit the results in [6], [7], we treat the VNs and

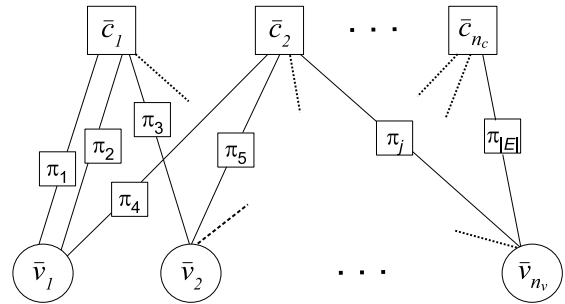


Fig. 1. Vectorized protograph.

CNs as constituent codes in a concatenated coding scheme. More specifically, the group of N VNs of type v_i is considered to be a constituent (repetition) code with a weight- d_i input of length N and q_{v_i} length- N outputs. Also, the group of N CNs of type c_j is considered to be a constituent code with q_{c_j} inputs, each of length N , and a fictitious output of weight zero.

Let $A(\mathbf{d})$ be the average (over the ensemble) number of codewords having weight vector $\mathbf{d} = [d_1, d_2, \dots, d_{n_v}]$ corresponding to the n_v VN N -groups and satisfying the protograph constraints. $A(\mathbf{d})$ is the *vector weight enumerator* for the ensemble of codes of length $N \cdot n_v$ described by the protograph. As shown in [5, Eq. 2], by exploiting the uniform interleaver property, we may write

$$A(\mathbf{d}) = \frac{\prod_{j=1}^{n_c} A^{c_j}(\mathbf{d}_j)}{\prod_{i=1}^{n_v} \binom{N}{d_i}^{q_{v_i}-1}} \quad (1)$$

where $\mathbf{d}_j = [d_{j_1}, d_{j_2}, \dots, d_{j_{q_{c_j}}}]$ is a weight vector which describes the weights of the N -bit words on the edges connected to CN c_j , produced by the VNs neighboring c_j . The elements of \mathbf{d}_j comprise a subset of the elements of \mathbf{d} .

Let S_t be the set of transmitted VNs and let S_p be the set of punctured VNs. Then the average number of codewords of weight d in the ensemble, denoted by A_d , equals the sum of $A(\mathbf{d})$ over all \mathbf{d} for which $\sum_{\{d_i: v_i \in S_t\}} d_i = d$. Notationally,

$$A_d = \sum_{\{d_i: v_i \in S_t\}} \sum_{\{d_k: v_k \in S_p\}} A(\mathbf{d}) \quad (2)$$

under the constraint $\sum_{\{d_i: v_i \in S_t\}} d_i = d$. To evaluate A_d in (2), one first needs to compute the vector weight enumerators, $A^{c_j}(\mathbf{d}_j)$, for the constraint nodes c_j , as seen in (1).

Consider the constituent (μ, κ) linear block code \mathcal{C} . For standard LDPC codes, \mathcal{C} is a single parity-check (SPC) code and $\mu = \kappa + 1$. Let $\mathbf{M}^{\mathcal{C}}$ be the $K \times \mu$ matrix with the codewords of \mathcal{C} as its rows, where $K = 2^r$ is the number of codewords in \mathcal{C} . In [1], [2] we showed that the vector weight enumerator $A^{\mathcal{C}}(\mathbf{w})$ is given by

$$A^{\mathcal{C}}(\mathbf{w}) = \sum_{\{\mathbf{n}\}} C(N; n_1, n_2, \dots, n_K), \quad (3)$$

where $\mathbf{w} = [w_1, w_2, \dots, w_\mu]$ is the weight vector at the input to the constituent code, $\{\mathbf{n}\}$ is the set of integer solutions to $\mathbf{w} = \mathbf{n} \cdot \mathbf{M}^{\mathcal{C}}$, with $n_1, n_2, \dots, n_K \geq 0$ and $\sum_{k=1}^K n_k = N$. In (3) $C(N; n_1, n_2, \dots, n_K)$'s are multinomial coefficients.

In the asymptotic case, let us define the normalized logarithmic asymptotic weight enumerator (we will simply call it the *asymptotic weight enumerator*), $r(\delta)$, as

$$r(\delta) = \limsup_{n \rightarrow \infty} \frac{\ln A_d}{n}, \quad (4)$$

where $\delta = d/n$ (recall n is the number of transmitted variable nodes in the code). Following [5], because the formulas in the previous section involve the number of copies, N , instead of n , we define the function

$$\tilde{r}(\tilde{\delta}) = \limsup_{N \rightarrow \infty} \frac{\ln A_d}{N}, \quad (5)$$

where $\tilde{\delta} = d/N$. Note that $n = |S_t| \cdot N$ and so

$$r(\delta) = \frac{1}{|S_t|} \tilde{r}(|S_t| \cdot \delta). \quad (6)$$

where $\tilde{r}(\tilde{\delta})$ [1], [2] is

$$\tilde{r}(\tilde{\delta}) = \max_{\{\tilde{\delta}_i: v_i \in S_t\}} \left\{ \max_{\{\tilde{\delta}_k: v_k \in S_p\}} \left\{ \sum_{j=1}^{n_c} a^{c_j}(\tilde{\delta}_j) - \sum_{i=1}^{n_v} (q_{v_i} - 1) H(\tilde{\delta}_i) \right\} \right\}, \quad (7)$$

under the constraint $\sum_{\{\tilde{\delta}_i: v_i \in S_t\}} \tilde{\delta}_i = \tilde{\delta}$. In (7), $a^{c_j}(\tilde{\delta}_j)$ is the asymptotic vector weight enumerator of the constraint node c_j , $H(\tilde{\delta}_i) = -(1 - \tilde{\delta}_i) \ln(1 - \tilde{\delta}_i) - \tilde{\delta}_i \ln \tilde{\delta}_i$, and $\tilde{\delta}_j = \mathbf{d}_j/N$. For a generic constituent CN code \mathcal{C} , this enumerator is defined as

$$a^{\mathcal{C}}(\boldsymbol{\omega}) = \limsup_{N \rightarrow \infty} \frac{\ln A^{\mathcal{C}}(\mathbf{w})}{N}, \quad (8)$$

where $\boldsymbol{\omega} = \mathbf{w}/N$, and is evaluated [1], [2] as

$$a^{\mathcal{C}}(\boldsymbol{\omega}) = \max_{\{P_{\boldsymbol{\omega}}\}} \{H(P_{\boldsymbol{\omega}})\}, \quad (9)$$

under the constraint that $\{P_{\boldsymbol{\omega}}\}$ is the set of solutions to $\boldsymbol{\omega} = P_{\boldsymbol{\omega}} \cdot \mathbf{M}^{\mathcal{C}}$, $p_1, p_2, \dots, p_K \geq 0$ and $\sum_{k=1}^K p_k = 1$. These

are the asymptotic equivalents of the constraints mentioned below (3). Note that, $P_{\boldsymbol{\omega}} = [p_1, p_2, \dots, p_K]$ is the empirical probability distribution of the codewords in \mathcal{C} given a sequence of N such codewords, where $p_k = n_k/N$ and n_k is the number of occurrences of the k^{th} codeword.

III. ON ASYMPTOTIC ENSEMBLE ENUMERATORS: THE CONJECTURE

The drawback of the method described in Section II in evaluating the asymptotic enumerators can be seen from (9): the number of the maximization arguments equals the number of the codewords K in the CN code \mathcal{C} . As an example, for the (15, 11) Hamming code, $K = 2^{11} = 2048$.

To alleviate this issue of having to maximize over a large number of variables, we did the following: First, we partitioned the protograph's VNs into subsets based on their neighborhoods. That is, two VNs belong to the same subset if and only if they have the same *type-neighborhood*, meaning, they are connected to an identical distribution of CN types. Given this partitioning of the set of VNs into subsets, the bits for a given CN code can themselves be partitioned into *bit subsets* in accordance with their membership in the VN subsets. We define the *subset-weight vector* (SWV) of a CN codeword as the vector whose components are the weights of bit subsets of the CN codeword. As an example, let $\bar{x} = [1010111]$ be a codeword of a length-7 CN code and let the CN code's bit subsets be $\{1, 2, 7\}$, $\{3, 4\}$, $\{5, 6\}$; then $\text{SWV}(\bar{x}) = [2, 1, 2]$. We define the *SWV enumerator* as the number of CN codewords that have the same SWV.

Next, we examined several examples, all of which led us to make the following conjecture:

Conjecture 1: In the maximization in (9), the optimal point occurs when codewords of equal SWV have the same proportion of occurrence in the constituent CN code.

The intuition behind this conjecture is the following: It is known that the distribution which maximizes the entropy, with no constraints, is the uniform distribution. However, in the case where the distribution has to satisfy some constraints, the one which maximizes the entropy is the "most" uniform distribution which satisfies the constraints (This can be proved by showing that any transfer of probability that makes the distribution more uniform increases the entropy. Note, the relative entropy to the uniform distribution can be used as a measure of uniformity.). Now, the problem of enumerating the codeword weights of a protograph-based code ensemble implies two kinds of constraints, the normalized weight δ , and the neighborhood. For a given CN, the weight constraint forces codewords of different weights to have different probabilities. Consequently, the most uniform distribution, which can be achieved, is the one where codewords of equal weight have the same proportion of occurrence in the constituent CN code. Actually, this is the case when all the CN's variable nodes have the same neighborhood. However, when they have different neighborhood, the neighborhood constraint forces codewords of different SWV to have different probabilities (even if they

have the same weight). Therefore, the most uniform distribution, which can be achieved, is the one where codewords of equal SWV have the same proportion of occurrence in the constituent CN code.

We use this conjecture with some simple linear algebra to rewrite (9) as

$$a^c(\omega) = \max_{\{\hat{P}\}} \left\{ H^*(\hat{P}) \right\}, \quad (10)$$

under the constraint that $\{\hat{P}\}$ is the set of solutions to $\omega = \hat{P} \cdot \hat{\mathbf{M}}^c$, $p^{(1)}, p^{(2)}, \dots \geq 0$ and $\Psi \cdot \hat{P}^T = 1$, where $\hat{P} = [p^{(1)}, p^{(2)}, \dots]$ is a vector of the distinct p_i 's in $P\omega$, $\Psi = [\psi_1, \psi_2, \dots]$ is a vector of the SWV enumerators of \mathcal{C} , and $\hat{\mathbf{M}}^c$ is constructed from \mathbf{M}^c by adding the rows of \mathbf{M}^c having the same SWV. Note that it is possible to have identical columns in $\hat{\mathbf{M}}^c$. This implies that the corresponding ω_i 's, in $\omega = [\omega_1, \omega_2, \dots, \omega_\mu]$, are equal. Finally, $H^*(\hat{P})$ is related to $H(P\omega)$ as follows:

$$\begin{aligned} H(P\omega) &= - \sum_{k=1}^K p_k \ln p_k \\ &= - \sum_{l=1}^L \left(p^{(l)} \ln p^{(l)} \right) \cdot \psi_l \\ &\triangleq H^*(\hat{P}). \end{aligned}$$

Example 1: Consider the protograph in Fig. 2, but with (15, 11) Hamming codes for the constraints \mathbf{H}_1 and \mathbf{H}_2 , where

$$\mathbf{H}_1 = [\mathbf{M}_1 \ \mathbf{M}_2] = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\mathbf{H}_2 = [\mathbf{M}_2 \ \mathbf{M}_1].$$

Note that there are $K = 2048$ codewords in each constituent code, so it would be difficult to evaluate (9) for this code. But, applying the conjecture, all VNs in this protograph have the same type-neighborhood. Also, after finding $\hat{\mathbf{M}}$, one finds that all of its columns are identical. Consequently, the VNs have the same normalized weight δ . The asymptotic weight enumerator is

$$r(\delta) = \frac{1}{15} \max_{\delta} \left\{ 2a^{H_1}(\tilde{\delta}) - 15H(\delta) \right\}.$$

where $\tilde{\delta} = [\delta, \delta, \dots, \delta]$ (15 of them). Now, to find $a^{H_1}(\tilde{\delta})$, define $p^{(\rho)}$ as the proportion of occurrence of a codeword of weight ρ in the constituent CN code, and so $\hat{P} = [p^{(0)}, p^{(3)}, p^{(4)}, p^{(5)}, p^{(6)}, p^{(7)}, p^{(8)}, p^{(9)}, p^{(10)}, p^{(11)}, p^{(12)}, p^{(15)}]$ and $\Psi = [1, 35, 105, 168, 280, 435, 435, 280, 168, 105, 35, 1, 1]$. Consequently,

$$a^{H_1}(\tilde{\delta}) = \max_{\{\hat{P}\}} \left\{ H^*(\hat{P}) \right\}, \quad (11)$$

under the constraint $\{\hat{P}\}$ is the set of solutions to $\tilde{\delta} = \hat{P} \cdot \hat{\mathbf{M}}^c$, $p^{(\rho)} \geq 0$, for all $p^{(\rho)}$ in \hat{P} , and $\Psi \cdot \hat{P}^T = 1$. Clearly, under these assumptions, the computation of $a^{H_1}(\tilde{\delta})$, hence $r(\delta)$, is vastly simplified.

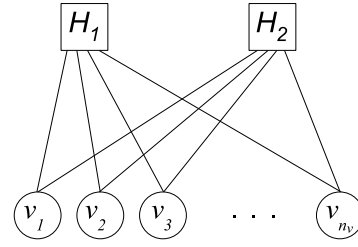


Fig. 2. Rate-7/15, $n_v = 15$ G-LDPC protograph.

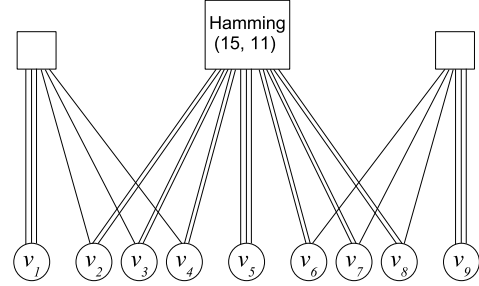


Fig. 3. Rate-1/3 G-LDPC protograph.

Example 2: Consider the rate-1/3 G-LDPC protograph in Fig. 3. The different subsets of VNs, according to their type-neighborhoods, are $\{v_2, v_3, v_4, v_6, v_7, v_8\}$, $\{v_1, v_9\}$, $\{v_5\}$. From the $\hat{\mathbf{M}}^c$ matrices of the CNs, we noticed that VNs in the same subset have the same normalized weight. So we associate the normalized weights $\{\tilde{\delta}_0, \tilde{\delta}_1, \tilde{\delta}_2\}$ with the VNs of the sets. The asymptotic weight enumerator is given by $r(\delta) = \tilde{r}(9\delta)/9$, where

$$\begin{aligned} \tilde{r}(\tilde{\delta}) &= \max_{\tilde{\delta}_0, \tilde{\delta}_1, \tilde{\delta}_2} \left\{ 2a^{SPC}(\tilde{\delta}_1) + a^H(\tilde{\delta}_2) \right. \\ &\quad \left. - 12H(\tilde{\delta}_0) - 4H(\tilde{\delta}_1) - 2H(\tilde{\delta}_2) \right\}. \end{aligned} \quad (12)$$

such that $6\tilde{\delta}_0 + 2\tilde{\delta}_1 + \tilde{\delta}_2 = \tilde{\delta}$, where $\tilde{\delta}_1$ and $\tilde{\delta}_2$ are the vectors of the associated normalized weights.

REFERENCES

- [1] S. Abu-Surra, W. E. Ryan, and D. Divsalar, "Ensemble weight enumerators for protograph-based generalized LDPC codes," in *UCSD Workshop on Information Theory and Its Applications*, February 2007. http://ita.ucsd.edu/workshop/07/files/paper/paper_218.pdf.
- [2] S. Abu-Surra, W. E. Ryan, and D. Divsalar, "Ensemble enumerators for protograph-based generalized LDPC codes," in *IEEE Global Telecomm. Conf., GLOBECOM '07*, 2007.
- [3] S. Abu-Surra, W. E. Ryan, and D. Divsalar, "Ensemble trapping set enumerators for protograph-based LDPC codes," in *Proc. of the 45th Annual Allerton Conf. on Commun., Control, and Computing*, Monticello, Illinois, September 2007.
- [4] J. Thorpe, "Low-density parity-check (LDPC) codes constructed from protographs," Tech. Rep. 42-154, IPN Progress Report, August 2003.
- [5] D. Divsalar, "Ensemble weight enumerators for protograph LDPC codes," in *IEEE Int. Symp. on Inform. Theory*, pp. 1554-1558, July 2006.
- [6] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: Performance analysis, design, and iterative decoding," *IEEE Trans. on Inform. Theory*, vol. 44, pp. 909-926, May 1998.
- [7] D. Divsalar, H. Jin, and R. McEliece, "Coding theorems for "turbo-like" codes," in *Proc. of 36th Allerton Conf.*, September 1998.