Ensemble Weight Enumerators for Protograph-Based Generalized LDPC Codes

Shadi Abu-Surra[a], William E. Ryan[a], and Dariush Divsalar[b]
[a]{shadia,ryan}@ece.arizona.edu, University of Arizona
[b]Dariush.Divsalar@jpl.nasa.gov, Jet Propulsion Laboratory

***Abstract*— Protograph-based LDPC codes have the advantages of a simple design (or search) procedure and highly structured encoders and decoders. These advantages have also been exploited in the design of protograph-based generalized LDPC (G-LDPC) codes. Recently, a technique for computing ensemble weight enumerators for protograph-based LDPC codes has been published. In the current paper, we extend those results to protograph-based G-LDPC codes. That is, we first derive ensemble weight enumerators for finite-length G-LDPC codes based on protographs, and then we consider the asymptotic case. The asymptotic results allow us to determine whether or not the typical minimum distance in the ensemble grows linearly with codeword length.**

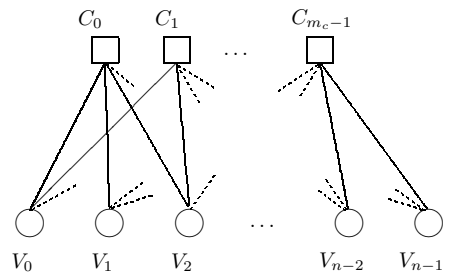

Fig. 1. Tanner graph of generalized LDPC code.

I. Introduction

Weight enumerating functions (or weight enumerators) for specific codes are useful for bounding or estimating the decoding error probability of channel codes. As noted by Gallager in his seminal monograph on LDPC codes [1], it is generally impractical to calculate the weight enumerator for a given code. In this case, Gallager and others have calculated the average performance for ensembles of codes. In many cases, it is often easier to calculate the asymptotic weight enumerators, that is, the weight enumerators for code ensembles with code length tending to infinity. The asymptotic results allows us to make statistical statements about the distance properties of code ensembles (particularly, the minimum distance) and on the minimum operable signal-to-noise-ratio (i.e., SNR threshold) for maximum-likelihood decoders.

Gallager derived asymptotic weight enumerators for the "Gallager ensembles" of regular LDPC codes in [1]. This result was extended to the irregular LDPC ensembles in [2]–[5]. In [6], [7], ensemble weight enumerators for serially concatenated codes and turbo-like codes were derived. Asymptotic weight enumerators for ensembles of protograph-based LDPC codes were computed in [8], [9]. In the latter paper, Divsalar first derived the ensemble weight enumerators for finite-length protograph-based LDPC codes [10] and then obtained the asymptotic results by letting the code length go to infinity.

In the current paper, we extend the work in [9] to protograph-based generalized LDPC (G-LDPC) codes. G-LDPC codes, for which more complex constraints than single parity-check (SPC) constraints are permissible, were first proposed by Tanner [11]. The Tanner graph of a G-LDPC code with length n and m_c constraints (n variable nodes and m_c constraint nodes) is depicted in Fig. 1. Selected types of such codes were investigated in [12]–[15]. G-LDPC codes based on protographs were studied in [16]–[21]. In this paper, we follow [9] to derive the finite-length weight enumerators and the asymptotic weight enumerators for protograph-based G-LDPC code ensembles.

This paper proceeds as follows: In Section II, we define our notation and derive the weight enumerators for the protograph G-LDPC codes. In Section III, we extend this result to the asymptotic case and provide some examples. In Section IV, we discuss topics which are related to asymptotic weight enumerators.

II. Ensemble weight enumerator for finite-length protograph-based G-LDPC codes

We start by defining a G-LDPC protograph. A protograph [10], [22] is a relatively small bipartite graph, containing variable nodes (VNs) and generalized 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 (under restrictions) to obtain a single, large graph. The edge permutations cannot be arbitrary. In particular, the nodes of the protograph are labeled so that if variable node v is connected to constraint node c in the protograph, then variable node v in a replica can only connect to one of the N replicated c constraint nodes. Doing so preserves the decoding threshold properties of the protograph. A protograph can possess parallel edges, i.e., two nodes can be connected by more than one edge. The copy-and-permute procedure must eliminate such parallel connections in order to obtain a derived graph appropriate for a parity-check matrix. 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

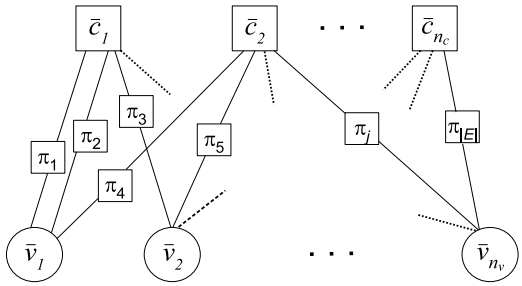

Fig. 2. Vectorized protograph.

bundle of (permuted) edges the same type. This "vectorized" protograph is depicted in Fig. 2.

Following the notation of [9], consider a G-LDPC protograph, $G = (V,C,E)$, where $V = \{v_1,v_2,\ldots,v_{n_v}\}$ is the set of n_v VNs, $C = \{c_1,c_2,\ldots,c_{n_c}\}$ is the set of n_c CNs, and E is the set of edges. Each edge $e \in E$ connects a variable node in V to a constraint node in C. Denote by q_{v_i} (q_{c_j}) the degree of variable (constraint) node v_i (c_j). Now consider the G-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 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 $\mathbf{d} = [d_1,\ d_2,\ \ldots\ ,d_{n_v}]$ 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. Let us further define

- $A^{v_i}(\mathbf{w}_i) = \binom{N}{d_i}\delta_{d_i,w_{i,1}} \cdots \delta_{d_i,w_{i,q_{v_i}}}$ = the vector weight enumerator for the type-v_i (VN) constituent code for a weight-d_i input, where $\mathbf{w}_i = \left[w_{i,1}, w_{i,2}, ..., w_{i,q_{v_i}}\right]$ is a weight vector describing the constituent code's output, and
- $A^{c_j}(\mathbf{z}_j)$ = the vector weight enumerator for the type-c_j (CN) constituent code and $\mathbf{z}_j = \left[z_{j,1}, z_{j,2}, ..., z_{j,q_{c_j}}\right]$, where $z_{j,l} = w_{i,k}$ if the lth edge of CN c_j is the kth edge of VN v_i.

As shown in [9, Eq. 2], by exploiting the uniform interleaver property, we may write

$$\begin{aligned} A(\mathbf{d}) &= \sum_{w_{m,u}} \frac{\prod_{i=1}^{n_v} A^{v_i}(\mathbf{w}_i) \prod_{j=1}^{n_c} A^{c_j}(\mathbf{z}_j)}{\prod_{s=1}^{n_v} \prod_{r=1}^{q_{v_s}} \binom{N}{w_{s,r}}} \\ &= \frac{\prod_{j=1}^{n_c} A^{c_j}(\mathbf{d}_j)}{\prod_{i=1}^{n_v} \binom{N}{d_i}^{q_{v_i}-1}} \end{aligned} \tag{1}$$

where the summation in the first line is over all weights $w_{m,u}$, $m = 1,...,n_v$ and $u = 1,...,q_{v_m}$, and $\mathbf{d}_j = \left[d_{j_1}, d_{j_2}, ..., d_{j_{q_{c_j}}}\right]$ 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 $\mathbf{d}_j$ are 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,

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

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). This was done for SPC nodes in [9]. In this section, we extend this computation to constraint nodes which correspond to any linear block code.

Consider the constituent (μ, κ) linear block code $\mathcal{C}$. We need to find its vector weight enumerator $A^{\mathcal{C}}(\mathbf{w})$, where $\mathbf{w} = [w_1, w_2, \ldots, w_\mu]$ is the weight vector at the input to the constituent code. Following [9], the $\{A^{\mathcal{C}}(\mathbf{w})\}$ may be found as the coefficients of the multi-dimensional z-transform of $\{A^{\mathcal{C}}(\mathbf{w})\}$, which is easy to obtain. Exploiting the uniform interleaver property and the fact that the multi-dimensional z-transform of a single constraint node is $\sum_{\mathbf{x}\in\mathcal{C}} W_1^{x_1} W_2^{x_2} \ldots W_\mu^{x_\mu}$, the multi-dimensional z-transform for N copies of the protograph is

$$A^{\mathcal{C}}(W_1, W_2, \ldots, W_\mu) = \left(\sum_{\mathbf{x}\in\mathcal{C}} W_1^{x_1} W_2^{x_2} \ldots W_\mu^{x_\mu}\right)^N, \tag{3}$$

where the W_i's are indeterminate bookkeeping variables and $\mathbf{x} = [x_1, x_2, \ldots, x_\mu]$, $x_i \in \{0,1\}$, is a codeword in $\mathcal{C}$. Expanding the righthand side of (3) will yield the form

$$A^{\mathcal{C}}(W_1, W_2, \ldots, W_\mu) = \sum_{\mathbf{w}} A^{\mathcal{C}}(\mathbf{w}) W_1^{w_1} W_2^{w_2} \ldots W_\mu^{w_\mu}, \tag{4}$$

from which we may obtain $A^{\mathcal{C}}(\mathbf{w})$. The direct application of the multinomial theorem on the righthand side of (3) gives

$$\begin{aligned} A^{\mathcal{C}}(W_1, W_2, \ldots, W_\mu) = \sum_{\substack{n_1,n_2,\ldots,n_K\geq 0 \\ n_1+n_2+\ldots+n_K=N}} & C\left(N; n_1, n_2, \ldots, n_K\right) \\ & \times \prod_{\mathbf{x}\in\mathcal{C}} \left(W_1^{x_1} W_2^{x_2} \ldots W_\mu^{x_\mu}\right)^{n_i} \end{aligned} \tag{5}$$

where $K = 2^\kappa$ is the number of codewords in $\mathcal{C}$ and $C\left(N; n_1, n_2, \ldots, n_K\right)$ is the multinomial coefficient, given by

$$C\left(N; n_1, n_2, \ldots, n_K\right) = \frac{N!}{n_1! n_2! \ldots n_K!}.$$

Let $\mathbf{M}^{\mathcal{C}}$ be the $K \times \mu$ matrix with the codewords of $\mathcal{C}$ as its rows and $\mathbf{n} = [n_1, n_2, \ldots, n_K]$. Then (5) can be written as

$$\begin{aligned} A^{\mathcal{C}}(W_1, W_2, \ldots, W_\mu) = \sum_{\mathbf{w}} \sum_{\{\mathbf{n}\}} & C\left(N; n_1, n_2, \ldots, n_K\right) \\ & \times W_1^{w_1} W_2^{w_2} \ldots W_\mu^{w_\mu}, \end{aligned} \tag{6}$$

where $\{\mathbf{n}\}$ is the set of integer solutions to $\mathbf{w} = \mathbf{n} \cdot \mathbf{M}^{\mathcal{C}}$, under the constraints $n_1, , n_2, , \ldots, n_K \geq 0$ and $\sum_{k=1}^{K} n_k = N$. To see the last step, note that the product in (5) can be manipulated as follows

$$\prod_{\mathbf{x}\in\mathcal{C}} \left(W_1^{x_1} W_2^{x_2} \ldots W_\mu^{x_\mu}\right)^{n_i} = W_1^{w_1} W_2^{w_2} \ldots W_\nu^{w_\nu},$$

where $w_j = \sum_{\mathbf{x}\in\mathcal{C}} x_j n_i$, $j = \{1,\ 2,\ \ldots,\ \mu\}$. Also, if $\mathbf{w} = \mathbf{n}\cdot\mathbf{M}^{\mathcal{C}}$ has more than one solution, the term $W_1^{x_1} W_2^{x_2} \ldots W_\mu^{x_\mu}$ will appear as a common factor in all of the terms that are associated with these solutions. This explains the presence of the second summation in (6). Finally, comparing (4) and (6) leads to the expression of the vector weight enumerator,

$$A^{\mathcal{C}}(\mathbf{w}) = \sum_{\{\mathbf{n}\}} C\left(N; n_1, n_2, \ldots, n_K\right), \tag{7}$$

where $\{\mathbf{n}\}$ is the set of integer solutions to $\mathbf{w} = \mathbf{n} \cdot \mathbf{M}^{\mathcal{C}}$, with $n_1, n_2, \ldots, n_K \geq 0$ and $\sum_{k=1}^{K} n_k = N$.

Example 1: Consider the degree-4 SPC constraint node. The codeword set is $\mathcal{C} = \{$0000, 1001, 0101, 1100, 0011, 1010, 0110, 1111$\}$, and so $K = 8$.
(a) Consider $N = 3$ protograph copies and the weight vector $\mathbf{w} = [2,\ 2,\ 2,\ 2]$. From $\mathbf{w} = \mathbf{n} \cdot \mathbf{M}^{\mathcal{C}}$ and the associated constraints on $\mathbf{n}$, it is easy to see that $\{\mathbf{n}\} = \{$[0, 0, 0, 1, 1, 0, 0, 1], [0, 0, 1, 0, 0, 1, 0, 1], [0, 1, 0, 0, 0, 0, 1, 1], [1, 0, 0, 0, 0, 0, 0, 2]$\}$, from this $A^{\mathcal{C}}(\mathbf{w}) = 21$ (via (7)).
(b) When $N = 4$ and $\mathbf{w} = [4,\ 2,\ 2,\ 2]$, $\{\mathbf{n}\} = \{$[0, 1, 0, 1, 0, 1, 0, 1]$\}$ and so $A^{\mathcal{C}}(\mathbf{w}) = 24$.
(c) When $N = 4$, and $\mathbf{w} = [3,\ 2,\ 2,\ 2]$, $\{\mathbf{n}\}$ is empty and so $A^{\mathcal{C}}(\mathbf{w}) = 0$. $\square$

Example 2: Consider the protograph with a single (7, 4) Hamming constraint node and 7 degree-1 VNs, all transmitted. Noting that the denominator in (1) is unity (since $q_{v_i} = 1$ for all i) and the numerator is $A^{c_1}(\mathbf{d}_1)$ (since $n_c = 1$), we will compute the first four A_d's assuming $N = 4$ copies of the protograph. The Hamming code is generated by

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix},$$

from which the matrix $\mathbf{M}^{\mathcal{C}}$ may be obtained; we assume the codewords are listed in the natural binary order with respect to the 16 input words. From (1) and (2), with $n_v = 7$ and $n_c = 1$, $A_d = \sum_{\mathbf{d}} A^{Hamm}(\mathbf{d})$ such that $\sum d_j = d$. Thus, $A_0 = A_{[0,0,0,0,0,0,0]} = C(4; 4, 0, \ldots, 0) = 1$. $A_1 = \sum_{\mathbf{d}} A^{Hamm}(\mathbf{d})$ such that $\sum d_j = 1$, but any such $\mathbf{d}$ must result in $\{\mathbf{n}\}$ empty. Consequently, $A_1 = 0$. Similarly, one finds $A_2 = 0$. With $A_3 = \sum_{\mathbf{d}} A^{Hamm}(\mathbf{d})$ such that $\sum d_j = 3$, $\mathbf{d} = \{$[1,0,0,0,0,1,1], [0,1,0,0,1,1,0], [1,0,1,0,1,0,0], [0,1,1,0,1,0,0], [0,0,0,1,1,0,1], [1,1,0,1,1,0,0,0], [0,0,1,1,0,1,0]$\}$. Each $\mathbf{d}$ yields only one solution $\mathbf{n}$ to the equation $\mathbf{d} = \mathbf{n}\cdot\mathbf{M}^{\mathcal{C}}$. Each solution has $n_1 = 3$ together with $n_k = 1$, where k corresponds to a row in $\mathbf{M}^{\mathcal{C}}$ containing one of the 7 weight-3 codewords. Note, the other $\mathbf{d}$'s that achieve $\sum d_j = 3$ result in $\{\mathbf{n}\}$ empty. Using (1), (2), and (7), $A_d = \sum_{\mathbf{d}} A^{Hamm}(\mathbf{d}) = \sum_{\{\mathbf{n}\}} C\left(4; 3, 0, ..., 1, ..., 0\right) = 28$. $\square$

III. Asymptotic ensemble weight enumerators

In this section we will extend the results of the previous section to the asymptotic case. Let us start by defining the asymptotic weight enumerator, $r(\delta)$, as

$$r(\delta) = \lim_{n\to\infty} \sup \frac{\ln A_d}{n}, \tag{8}$$

where $\delta = d/n$ (recall n is the number of transmitted variable nodes in the code). Following [9], because the formulas in the previous section involve the number of copies, N, instead of n, we define the function

$$\tilde{r}(\tilde{\delta}) = \lim_{N\to\infty} \sup \frac{\ln A_d}{N}, \tag{9}$$

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

$$r(\delta) = \frac{1}{|S_t|} \tilde{r}(|S_t| \cdot \delta). \tag{10}$$

We also define $\max^*(x, y) \triangleq \log\left(e^x + e^y\right)$ and we similarly define $\max^*$ when more than two variables are involved. When x and y are large and distinct (so that e^x and e^y are vastly different), then $\max^*(x, y) \simeq \max(x, y)$, and similarly for more than two variables.

From (2), we have

$$\begin{aligned} \ln A_d &= \max^*_{\{d_l : v_l \in S_t\}} \left\{ \max^*_{\{d_k : v_k \in S_p\}} \{\ln A(\mathbf{d})\} \right\}, \\ &= \max_{\{d_l : v_l \in S_t\}} \left\{ \max_{\{d_k : v_k \in S_p\}} \{\ln A(\mathbf{d})\} \right\}, \\ &= \max_{\{d_l : v_l \in S_t\}} \left\{ \max_{\{d_k : v_k \in S_p\}} \left\{ \sum_{j=1}^{n_c} \ln A^{c_j}\left(\mathbf{d}_j\right) \right.\right. \\ &\qquad \left.\left. - \sum_{i=1}^{n_v} (q_{v_i} - 1) \ln \binom{N}{d_i} \right\} \right\}, \end{aligned} \tag{11}$$

under the constraint $\sum_{\{d_i : v_i \in S_t\}} d_i = d$. The second equality holds when N is very large and the third equality follows by invoking (1). Taking the limit as $N \to \infty$ and applying the result (from Stirling's formula) $\lim_{N\to\infty} \sup \ln \binom{N}{d_i}/N = H(\tilde{\delta}_i) = -(1 - \tilde{\delta}_i) \ln(1 - \tilde{\delta}_i) - \tilde{\delta}_i \ln \tilde{\delta}_i$, where $\tilde{\delta}_i = d_i/N$, we obtain

$$\begin{aligned} \tilde{r}(\tilde{\delta}) = \max_{\{\tilde{\delta}_l : v_l \in S_t\}} \left\{ \max_{\{\tilde{\delta}_k : v_k \in S_p\}} \left\{ \sum_{j=1}^{n_c} a^{c_j}(\tilde{\boldsymbol{\delta}}_j) \right.\right. \\ \left.\left. - \sum_{i=1}^{n_v} (q_{v_i} - 1) H(\tilde{\delta}_i) \right\} \right\}, \end{aligned} \tag{12}$$

under the constraint $\sum_{\{\tilde{\delta}_i : v_i \in S_t\}} \tilde{\delta}_i = \tilde{\delta}$. In (12), $a^{c_j}(\tilde{\boldsymbol{\delta}}_j)$ is the asymptotic vector weight enumerator of the constraint node c_j, and $\tilde{\boldsymbol{\delta}}_j = \mathbf{d}_j/N$. For a generic constituent CN code $\mathcal{C}$, this enumerator is defined as

$$a^{\mathcal{C}}(\boldsymbol{\omega}) = \lim_{N\to\infty} \sup \frac{\ln A^{\mathcal{C}}(\mathbf{w})}{N}, \tag{13}$$

$\boldsymbol{\omega} = \mathbf{w}/N$.

We may obtain a simple expression for $a^{\mathcal{C}}(\boldsymbol{\omega})$ using the method of types [23, Ch. 12]. We define the type $P_{\boldsymbol{\omega}}$ as

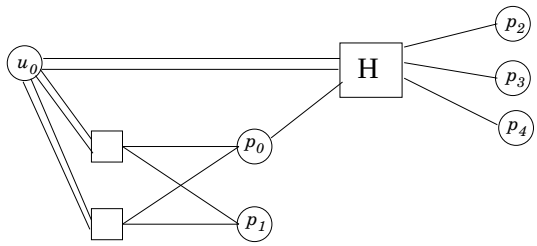

Fig. 3. Rate-1/6 HD-LDPC protograph.

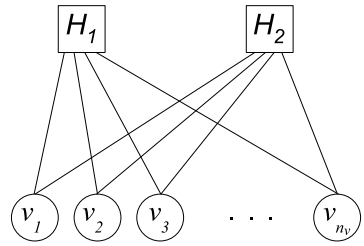

Fig. 4. Rate-1/7, $n_v = 7$, (or, rate-7/15, $n_v = 15$) HD-LDPC protograph.

the relative proportion of occurrences of each codeword in constituent CN code $\mathcal{C}$. In other words, $P_{\boldsymbol{\omega}} = [p_1, p_2, \ldots, p_K]$ 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 kth codeword. Then the type class of $P_{\boldsymbol{\omega}}$, $T(P_{\boldsymbol{\omega}})$, is the set of all length-N sequences of codewords in $\mathcal{C}$, each containing n_k occurrences of the kth codeword in $\mathcal{C}$, for $k = 1, 2, ..., K$. Observe that $|T(P_{\boldsymbol{\omega}})| = C(N; n_1, n_2, \ldots, n_K)$. From [23, Thm.12.1.3] $|T(P_{\boldsymbol{\omega}})| \to e^{N \cdot H(P_{\boldsymbol{\omega}})}$, as $N \to \infty$, where $H(P_{\boldsymbol{\omega}}) = -\sum_{k=1}^{K} p_k \ln p_k$. Consequently, as $N \to \infty$ we rewrite (7) as

$$\begin{aligned} A_{\mathbf{w}}^{\mathcal{C}} &= \sum_{\{\mathbf{n}\}} C(N; n_1, n_2, \ldots, n_K) \\ &= \sum_{\{P_{\boldsymbol{\omega}}\}} |T(P_{\boldsymbol{\omega}})| \\ &\to \sum_{\{P_{\boldsymbol{\omega}}\}} e^{N \cdot H(P_{\boldsymbol{\omega}})} . \end{aligned} \tag{14}$$

It follows from (13) and (14) that

$$a^{\mathcal{C}}(\boldsymbol{\omega}) = \max_{\{P_{\boldsymbol{\omega}}\}} \{H(P_{\boldsymbol{\omega}})\}, \tag{15}$$

under the constraint that $\{P_{\boldsymbol{\omega}}\}$ is the set of solutions to $\boldsymbol{\omega} = P_{\boldsymbol{\omega}} \cdot \mathbf{M}^{\mathcal{C}}$, $p_1, p_2, \ldots, p_K \geq 0$ and $\sum_{k=1}^{K} p_k = 1$. These are the asymptotic equivalents of the constraints mentioned below (7).

Let d_{min} be the minimum distance of a linear block code. In [1] it was shown that

$$\Pr\{d_{min} \leq n\delta\} \leq \sum_{l=1}^{n\delta} \binom{n}{l} P(l) = \sum_{l=1}^{n\delta} A_l, \tag{16}$$

where $P(l)$ is the probability that a particular sequence of weight l is a codeword in the ensemble. Let δ_{min} be the second zero crossing of $r(\delta)$ (the first crossing is $r(0) = 0$). Assuming δ_{min} exists, it is called the typical minimum distance if $r(\delta) < 0$ for all $0 < \delta < \delta_{min}$ [9]. If $\sum_{l=1}^{n\delta_{min}} A_l \to A_{n\delta_{min}}$ as n becomes large, then $\Pr\{d_{min} \leq n\delta_{min} - 1\} \leq \sum_{l=1}^{n\delta_{min}-1} A_l \to 0$, so that with probability near one, $d_{min} = n\delta_{min}$. That is, the minimum distance of virtually all of the members of the code ensemble increases linearly with n.

Example 3: We evaluate the asymptotic weight enumerator for the rate-1/6 HD-LDPC code presented in [17]–[20], which is based on the protograph in Fig. 3 (HD = "Hamming-doped"). The constraint node marked by H is a (6, 3) shortened Hamming code with parity check matrix

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}. \tag{17}$$

Associate the vector of normalized weights $[\tilde{\delta}_0,\ \tilde{\delta}_1,\ \tilde{\delta}_2, \tilde{\delta}_3,\ \tilde{\delta}_4,\ \tilde{\delta}_5]$ with the vector of variable nodes $[u_0, p_0, p_1, p_2, p_3, p_4]$, and label the columns of $\mathbf{H}$ by $[u_0, u_0, p_0, p_2, p_3, p_4]$, in accordance with Fig. 3. Note that the two SPC nodes have the same neighborhood, and so they have the same vector weight numerator. Under this setup, because $|S_t| = 6$, the asymptotic weight enumerator is given by $r(\delta) = \tilde{r}(6\delta)/6$, where

$$\begin{aligned} \tilde{r}(\tilde{\delta}) = \max_{\tilde{\delta}_0, \ldots, \tilde{\delta}_5} \Big\{ & 2a^{SPC}(\tilde{\boldsymbol{\delta}}_1) + a^{H}(\tilde{\boldsymbol{\delta}}_2) \\ & -5H(\tilde{\delta}_0) - 2H(\tilde{\delta}_1) - H(\tilde{\delta}_2) \Big\}. \end{aligned} \tag{18}$$

such that $\sum_{i=0}^{5} \tilde{\delta}_i = \tilde{\delta}$, where $\tilde{\boldsymbol{\delta}}_1 = [\tilde{\delta}_0, \tilde{\delta}_0, \tilde{\delta}_1, \tilde{\delta}_2]$ and $\tilde{\boldsymbol{\delta}}_2 = [\tilde{\delta}_0, \tilde{\delta}_0, \tilde{\delta}_1, \tilde{\delta}_3, \tilde{\delta}_4, \tilde{\delta}_5]$. The asymptotic weight enumerator is shown in Fig. 5. The figure shows that there is no typical minimum distance δ_{min} for this code (or $\delta_{min} = 0$) so that d_{min} for this code ensemble does not grow linearly with n. We also evaluated the asymptotic weight enumerator for the case when the variable node u_0 is punctured. Equation (18) still holds but the constraint becomes $\sum_{i=1}^{5} \tilde{\delta}_i = \tilde{\delta}$. The result is represented by the rate-1/5 HD-LDPC curve in Fig. 5, where again we see that d_{min} for the ensemble does not grow linearly with n. (In spite of this, we have found via simulation that short codes designed according to this protograph provide good performance in both the waterfall and the floor region [17]–[20].) □

Example 4: Consider the protograph in Fig. 4, where the parity-check matrix $\mathbf{H}_1$ corresponds to the (7, 4) Hamming code generated by $\mathbf{G}$ of Example 2. The parity-check matrix $\mathbf{H}_2$ corresponds to the following column-permutation of $\mathbf{H}_1$: $(6, 7, 1, 2, 3, 4, 5)$. That is, the 1st column of $\mathbf{H}_2$ is the 6th column of $\mathbf{H}_1$, ..., and the 7th column of $\mathbf{H}_2$ is the 5th column of $\mathbf{H}_1$. A code constructed per this protograph has rate 1/7 since each CN represents 3 redundant bits. The variable nodes v_1, v_2, …,v_7 have the normalized weights $\tilde{\delta}_1, \tilde{\delta}_2, \ldots, \tilde{\delta}_7$. The asymptotic weight enumerator is $r(\delta) = \tilde{r}(7\delta)/7$, where

$$\tilde{r}(\tilde{\delta}) = \max_{\tilde{\delta}_1, \ldots, \tilde{\delta}_7} \left\{ a^{H_1}(\tilde{\boldsymbol{\delta}}_1) + a^{H_2}(\tilde{\boldsymbol{\delta}}_2) - \sum_{j=1}^{7} H(\tilde{\delta}_j) \right\}. \tag{19}$$

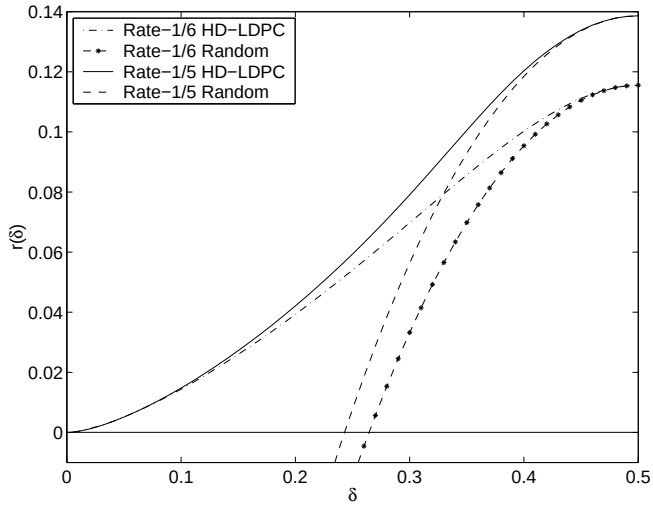

Fig. 5. Asymptotic weight enumerator for the Rate-1/6 (1/5) HD-LDPC code.

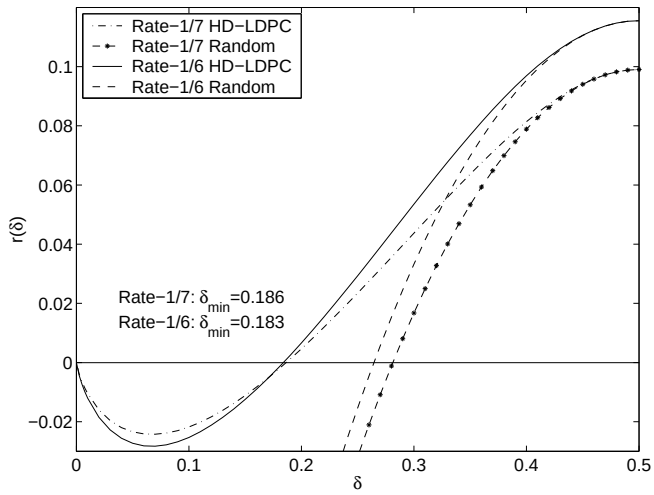

Fig. 6. Asymptotic weight enumerator for the Rate-1/7 (1/6) HD-LDPC code.

such that $\sum_{i=1}^{7} \tilde{\delta}_i = \tilde{\delta}$, where $\tilde{\boldsymbol{\delta}}_1 = [\tilde{\delta}_1, \tilde{\delta}_2, \ldots, \tilde{\delta}_7]$, and $\tilde{\boldsymbol{\delta}}_2 = [\tilde{\delta}_6, \tilde{\delta}_7, \tilde{\delta}_1, \tilde{\delta}_2, \tilde{\delta}_3, \tilde{\delta}_4, \tilde{\delta}_5]$. We also evaluated the asymptotic weight enumerator for the rate-1/6 HD-LDPC code which results from puncturing one of the VNs. The result is presented in Fig. 6. Note that this ensemble has a relatively large δ_{min}. Consequently, a code based on this protograph has, with probability near one, a large minimum distance. □

IV. On asymptotic ensemble weight enumerators

In this section we discuss the following: First, we discuss the difficulty in evaluating the asymptotic weight enumerators of arbitrary protograph-based G-LDPC code via the techniques of the previous sections. This motivates a conjecture which, if provable, can vastly simplify computations. Second, we show how to utilize the asymptotic weight enumerators in the design of protograph-based G-LDPC codes. Specifically, we give an example which demonstrates the impact of column permutations of the parity-check matrices of CNs (first pointed out by Tanner [11]).

The drawback of our method in evaluating the asymptotic weight enumerators can be seen from (15): the number of the maximization arguments equals the number of the codewords, K, in the constituent code $\mathcal{C}$. In the previous examples we considered constituent codes with K small, but this is not the case in general. Our future research will include finding a general method to alleviate this issue of having to maximize over large sets. However, we have made the following observations when performing Example 4 and examples of other protographs with similar characteristics (not included here):

1) In maximizing over the $\tilde{\delta}_i$'s in (12), the optimal point occurs when all of the $\tilde{\delta}_i$ are equal.
2) In the maximization in (15), the optimal point occurs when codewords of equal weight have the same proportion of occurrence in the constituent CN code.

These observations have led us to make the following conjecture: *Whenever all of the constraints nodes in the G-LDPC protograph have the same weight enumerators, and all of the variable nodes in the protograph have the same degree, the above observations will hold.* We apply this conjecture in the next example.

Example 5: Consider again the protograph in Fig. 4, but with (15, 11) Hamming codes for the constraints $\mathbf{H}_1$ and $\mathbf{H}_2$ [17]–[19], where

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

Note that there are $K = 2048$ codewords in each constituent code, so it would be difficult to evaluate (15) for this code. Because the conditions of the conjecture (same CN weight enumerators, same VN degree) hold, we will adopt the two observations above as assumptions. Let us define $p^{(\rho)}$ as the proportion of occurrence of a codeword of weight ρ in the constituent CN code, and let $\bar{P} = [p^{(0)}, p^{(3)}, p^{(4)}, p^{(5)}, p^{(6)}, p^{(7)}, p^{(8)}, p^{(9)}, p^{(10)}, p^{(11)}, p^{(12)}, p^{(15)}]$. By adopting the first observation above as an assumption, all elements $\tilde{\delta}_i$ in $\tilde{\boldsymbol{\delta}}$ will have the same value, say δ'. Also, because $\sum_{i=1}^{15} \tilde{\delta}_i = \tilde{\delta}$, it follows that $\delta' = \tilde{\delta}/15 = \delta$. Consequently, $a^{H_j}(\tilde{\boldsymbol{\delta}})$, $j = ,1,2$, is a function of δ, and so we write

$$a^{H_j}(\tilde{\boldsymbol{\delta}}) = a^{H_j}(\delta) = \max_{\{P(\tilde{\boldsymbol{\delta}})\}} \left\{ H(P(\tilde{\boldsymbol{\delta}})) \right\}, \qquad (20)$$

under the constraint $\{P(\tilde{\boldsymbol{\delta}})\} = \{[p_1, p_2, \ldots, p_K]\}$ is the set of solutions to $\tilde{\boldsymbol{\delta}} = [\delta, \delta, \ldots, \delta] = P(\tilde{\boldsymbol{\delta}}) \cdot \mathbf{M}^{\mathcal{C}}$, $p_1, p_2, \ldots, p_K \geq 0$ and $\sum_{k=1}^{K} p_k = 1$. Now, with the second observation above as an assumption (codewords of equal weight had the same

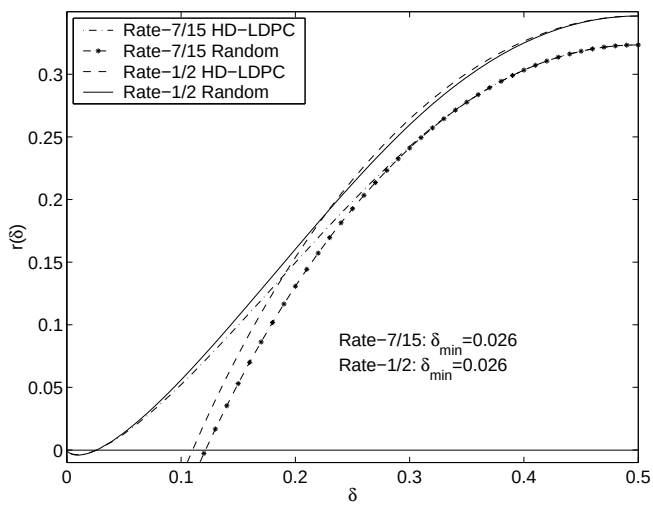

Fig. 7. Asymptotic weight enumerator for the Rate-7/15 (1/2) HD-LDPC code.

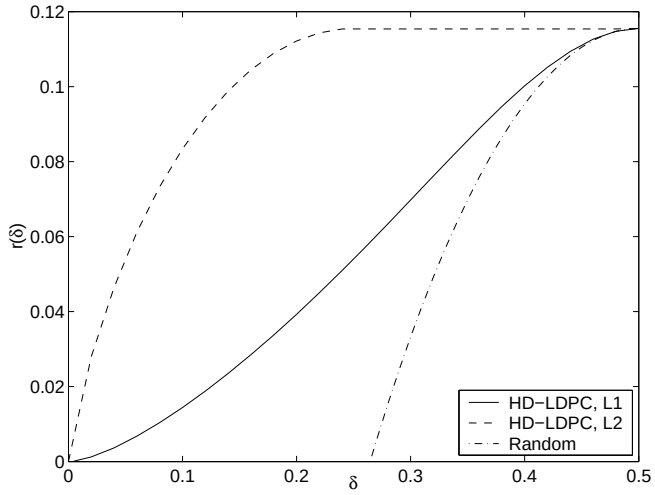

Fig. 8. Asymptotic weight enumerator for the Rate-1/6 HD-LDPC code with two different edge-labeling at the Hamming constraint.

proportion of occurrence), $H(P(\tilde{\boldsymbol{\delta}}))$ simplifies as

$$\begin{aligned} H(P(\tilde{\boldsymbol{\delta}})) &= -\sum_{k=1}^{K} p_k \ln p_k \\ &= -p^{(0)}\ln p^{(0)} - 35p^{(3)}\ln p^{(3)} - 105p^{(4)}\ln p^{(4)} \\ &\quad - 168p^{(5)}\ln p^{(5)} - 280p^{(6)}\ln p^{(6)} - 435p^{(7)}\ln p^{(7)} \\ &\quad -435p^{(8)}\ln p^{(8)} - 280p^{(9)}\ln p^{(9)} - 168p^{(10)}\ln p^{(10)} \\ &\quad - 105p^{(11)}\ln p^{(11)} - 35p^{(12)}\ln p^{(12)} - p^{(15)}\ln p^{(15)} \\ &\triangleq H(\bar{P}) \end{aligned}$$

where we have utilized weight enumerator information for the (15, 11) Hamming code (e.g., $A_0 = 1$, $A_3 = 35$). The constraint $[\delta, \delta, \ldots, \delta] = P(\tilde{\boldsymbol{\delta}}) \cdot \mathbf{M}^{\mathcal{C}}$ yields the two equivalent constraints $p^{(0)} = 1 - 5\delta + 35p^{(4)} + 112p^{(5)} + 280p^{(6)} + 580p^{(7)} + 725p^{(8)} + 560p^{(9)} + 392p^{(10)} + 280p^{(11)} + 105p^{(12)} + 4p^{(15)}$ and $p^{(3)} = 1/7\delta - 4p^{(4)} - 8p^{(5)} - 16p^{(6)} - 29p^{(7)} - (232/7)\,p^{(8)} - 24p^{(9)} - 16p^{(10)} - 11p^{(11)} - 4p^{(12)} - (1/7)\,p^{(15)}$. Thus, (20) can be replaced by

$$a^{H_j}(\delta) = \max_{\{\bar{P}\}} \left\{ H(\bar{P}) \right\},$$

under the two constraint equations above. Finally, the asymptotic weight enumerator is given by $r(\delta) = \tilde{r}(15\delta)/15$, where $\tilde{r}(\tilde{\delta}) = 2a^{H_1}(\tilde{\delta}/15) - 15H(\tilde{\delta}/15) = 2a^{H_1}(\delta) - 15H(\delta)$. Clearly, under these assumptions, the computation of $a^{H_j}(\delta)$, hence $r(\delta)$, is vastly simplified. The $r(\delta)$ result appears in Fig. 7. Also included in the figure is $r(\delta)$ for a rate-1/2 code obtained by puncturing one bit in the protograph. Note that, for both cases, d_{min} for the ensemble increases linearly with n. □

We now give an example which demonstrates the impact of column permutations of the parity-check matrices of CNs, thus demonstrating how asymptotic weight enumerators may be utilized in the design of protograph-based G-LDPC codes.

Example 6: Tanner [11] pointed out that permuting the CN labeling (equivalently, permuting the columns of the constituent codes' parity-check matrices) of G-LDPC codes produces different codes with different properties. In view of this, we show in this example how permuting the labeling on the CNs in the protograph of Example 3 greatly changes the asymptotic weight enumerator for the corresponding ensemble of codes. The results are shown in Fig. 8, where in the first labeling, L_1, the columns of $\mathbf{H}$ are labeled by $u_0, u_0, p_0, p_2, p_3, p_4$ and, in the second labeling, L_2, the columns of $\mathbf{H}$ are labeled by $u_0, u_0, p_2, p_0, p_3, p_4$. That is, referring to Fig. 3 and equation (17), for labeling L_1, node p_2 represents the (6,3) Hamming parity bit whereas for labeling L_2, node p_0 represents the (6,3) Hamming parity bit (while it is simultaneously an accumulator output). We see in Fig. 8 that labeling L_2 leads to a particularly poor ensemble: in addition to $\delta_{min} = 0$ (true also for labeling L_1), the multiplicities A_d are very large as evidenced by the large values of $r(\delta)$. □

V. Conclusion

We extended the ensemble weight enumerator technique of [9] to protograph-based G-LDPC codes. The method is relatively simple for CN's with small dimensionality k and it allowed us to show which G-LDPC ensembles under consideration had the property that d_{min} grows with code length. Unfortunately, the computational complexity for finding the vector weight enumerator grows as 2^k. In fact, we relied on a conjecture to find the ensemble weight enumerator of a G-LDPC code with (15,11) Hamming CNs.

Further work will include a proof of this conjecture and will utilize these results along with other tools (e.g., EXIT charts) in the design of G-LDPC codes with good properties (large d_{min} and good iterative decoding threshold). We also know how to derive stopping set enumerators for protograph-based G-LDPC codes and this is the topic of a future paper.

VI. Acknowledgments

This research was supported in part by grant NNX06AC17G from NASA Goddard Space Flight Center. This research in part was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with NASA.

References

[1] R. G. Gallager, *Low-density parity-check codes.* Cambridge, MA: MIT Press, 1963.

[2] S. Litsyn and V. Shevelev, "On ensembles of low-density parity-check codes: Asymptotic distance distributions," *IEEE Trans. on Inform. Theory*, vol. 48, pp. 887–908, April 2002.

[3] S. Litsyn and V. Shevelev, "Distance distributions in ensembles of irregular low-density parity-check codes," *IEEE Trans. on Inform. Theory*, vol. 49, pp. 3140–3159, December 2003.

[4] R. Ikegaya, K. Kasai, T. Shibuya, and K. Sakaniwa, "Asymptotic weight and stopping set distributions for detailedly represented irregular LDPC code ensembles," *IEEE Int. Symp. on Inform. Theory*, p. 208, July 2004.

[5] D. Burshtein and G. Miller, "Asymptotic enumeration methods for analyzing LDPC codes," *IEEE Trans. on Inform. Theory*, vol. 50, pp. 1115–1131, June 2004.

[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," *Proc. of 36th Allerton Conf.*, September 1998.

[8] S. L. Fogal, R. McEliece, and J. Thorpe, "Enumerators for protograph ensembles of LDPC codes," *IEEE Int. Symp. on Inform. Theory*, 2005.

[9] D. Divsalar, "Ensemble weight enumerators for protograph LDPC codes," *IEEE Int. Symp. on Inform. Theory*, pp. 1554–1558, July 2006.

[10] J. Thorpe, "Low-density parity-check (LDPC) codes constructed from protographs," Tech. Rep. 42-154, IPN Progress Report, August 2003.

[11] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. on Inform. Theory*, vol. 27, pp. 533–547, September 1981.

[12] J. Boutros, O. Pothier, and G. Zemor, "Generalized low density (Tanner) codes," *IEEE Int. Conf. on Commun., ICC '99*, pp. 441–445, June 1999.

[13] M. Lentmaier and K. S. Zigangirov, "Iterative decoding of generalized low-density parity-check codes," *IEEE Int. Symp. on Inform. Theory*, p. 149, August 1998.

[14] N. Miladinovic and M. Fossorier, "Generalized LDPC codes with Reed-Solomon and BCH codes as component codes for binary channels," *IEEE Global Telecommunications Conf., GLOBECOM '05*, November 2005.

[15] S. Vialle and J. Boutros, "A Gallager-Tanner construction based on convolutional codes," *Proceedings of Int. Workshop on Coding and Cryptography, WCC'99*, pp. 393–404, January 1999.

[16] G. Liva and W. E. Ryan, "Short low-error-floor Tanner codes with Hamming nodes," *IEEE Military Commun. Conf., MILCOM '05*, 2005.

[17] G. Liva, W. E. Ryan, and M. Chiani, "Design of quasi-cyclic Tanner codes with low error floors," *4th Int. Symp. on Turbo Codes, ISTC-2006*, April 2006.

[18] S. Abu-Surra, G. Liva, and W. E. Ryan, "Design and performance of selected classes of Tanner codes," *UCSD Workshop on Information Theory and Its Applications*, February 2006. http://ita.ucsd.edu/workshop/06/talks/papers/129.pdf.

[19] S. Abu-Surra, G. Liva, and W. E. Ryan, "Low-floor Tanner codes via Hamming-node or RSCC-node doping," *Lecture Notes in Computer Science*, vol. 3857, pp. 245–254, January 2006.

[20] S. Abu-Surra, G. Liva, and W. E. Ryan, "Design of generalized LDPC codes and their decoders," submitted to *IEEE Int. Conf. on Commun., ICC '07*, 2007.

[21] G. Liva, W. E. Ryan, and M. Chiani, "Quasi-cyclic generalized LDPC codes with low floors," accepted pending revision, *IEEE Trans. on Commun.*

[22] S. Lin, J. Xu, I. Djurdjevic, and H. Tang, "Hybrid construction of LDPC codes," *Proc. of the 40th Annual Allerton Conf. on Commun., Control, and Computing, Illinois*, October 2002.

[23] T. M. Cover, *Elements of information theory*. New York: Wiley, 1991.