On Design of Doubly-Generalized LDPC Codes Based on Multi-Type Information Functions

Enrico Paolini, Marco Chiani and Marc P. C. Fossorier

Abstract—Ensemble design of low-density parity-check (LDPC) codes and their generalizations is usually performed via numerical optimization techniques, such as differential evolution, in which a threshold analysis tool is always necessary. Threshold analysis of unstructured doubly-generalized LDPC (D-GLDPC) code ensembles over the binary erasure channel (BEC) can be performed via extrinsic information transfer (EXIT) chart, exploiting the information functions and split information functions of the check and variable component codes, respectively. In this paper, multi-type information functions of linear block codes are introduced as an extension of the concept of information functions, when the bit positions are assumed to be associated with different types. It is shown how multi-type information functions (together with their split counterparts) can be exploited within an EXIT analysis approach to perform threshold analysis over the BEC of multi-edge type D-GLDPC code ensembles. The proposed technique for threshold analysis captures D-GLDPC codes based on protographs as a special case.

I. INTRODUCTION

Multi-edge type (MET) low-density parity-check (LDPC) codes were introduced in [1] as a universal framework to analyze and design LDPC ensembles beyond the ones captured by a standard approach based on the degree distribution pair. In a MET LDPC ensemble, edges in the Tanner graph are partitioned into disjoint subsets, and each subset is associated with a specific *edge type*. Unstructured regular and irregular ensembles as well as several categories of structured ones represent specific MET ensemble instances. For example, a classical unstructured ensemble may be interpreted as MET ensemble where all edges are of the same type, while a protograph ensemble [2] as a MET ensemble where all edges of the same type correspond to the same edge of the original protograph.

It this paper, we combine the idea of MET Tanner graph with that of doubly-generalized LDPC (D-GLDPC) codes [3]–[5]. For D-GLDPC codes both variable nodes (VNs) and check nodes (CNs) are replaced with generic linear block codes, whereas in generalized LDPC (GLDPC) codes this replacement is allowed for CNs only. We consider the problem of MET D-GLDPC ensemble design, intended as generation of ensembles fulfilling given constraints and having a good decoding threshold. Even if the design may in principle be performed through the same numerical optimization techniques commonly employed for unstructured ensembles, a new tool

E. Paolini and M. Chiani are with DEIS/WiLAB, University of Bologna, via Venezia 52, 47521 Cesena (FC), Cesena, Italy (e-mail: e.paolini@unibo.it, marco.chiani@unibo.it).

M. P. C. Fossorier is with ETIS ENSEA, UCP, CNRS UMR-8051, 6 avenue du Ponceau, 95014 Cergy Pontoise, France (e-mail: mfossorier@ieee.org).

for threshold evaluation is required for the MET D-GLDPC case. In fact, at each step of the optimization process, it is necessary to compare the threshold of newly generated ensembles with those of current ensembles, which are possibly replaced.

A threshold analysis technique of MET D-GLDPC ensembles is developed for the case of transmission over a binary erasure channel (BEC). It is based on the concept of extrinsic information transfer (EXIT) analysis [6], an extension of EXIT chart [7] to the MET case, and relies on an analytical expression for the average extrinsic information outgoing from a VN or CN along its edges of a certain type. It is shown that this extrinsic information depends on what we call the *multi-type information functions* (for a CN) and the *multi-type split information functions* (for a VN). These parameters represent the generalization to the MET setting of the information functions [8] and split information functions [7] used for EXIT chart analysis over the BEC of classical unstructured D-GLDPC ensembles. A technique to calculate the threshold over the BEC for protograph GLDPC codes was developed in [9]. In this work we deal with the more general case of MET D-GLDPC code ensembles.

We finally observe that the weight distribution of MET LDPC and D-GLDPC codes was recently investigated in [10], [11] and [12], respectively, while the weight distribution of GLDPC codes based on protographs was studied in [13].

II. ENSEMBLE DESCRIPTION

A. Edge Types and Node Types

In the Tanner graph of a D-GLDPC code, every VN as well as every CN is a associated with a generic linear block code. We assume that there are u_v variable component code types $I_V = \{1, 2, \ldots u_v\}$ and u_c different check component code types $I_C = \{1, 2, \ldots, u_c\}$. We denote by $q^{(v)}$ and $k^{(v)}$ the length and the dimension of the variable component codes of type v, respectively, and by $s^{(c)}$ and $h^{(c)}$ the length and the dimension of the check component codes of type c, respectively. We also define $m^{(c)} = s^{(c)} - h^{(c)}$. Two variable component codes are of the same type $v \in I_V$ when they have the same dimension, the same length and are represented through the same generator matrix. Two check component codes are of the same type $c \in I_C$ when they have the same dimension, the same length and the same parity-check matrix.

A VN corresponding to a type-v variable component code has $q^{(v)}$ edges incident on it and is associated with $k^{(v)}$ encoded bits of the overall D-GLDPC code. These $k^{(v)}$ bits are interpreted by the VN as its local information bits, while the $q^{(v)}$ sockets of the VN are associated with its local encoded

bits. A CN corresponding to a type-c check component code has $s^{(c)}$ edges incident on it. The $s^{(c)}$ sockets of the CN are associated with its local encoded bits, on which the node imposes $m^{(c)}$ local parity-check constraints.

Within a MET framework, the edges of the Tanner graph are assumed to be of different types $z \in \mathscr{E} = \{1, 2, \ldots, n_e\}$, where n_e is the number of different edge types.

The generic VN is associated with a pair of vectors $(\mathbf{b}, \mathbf{d})$.[1] The length $|\mathbf{d}|$ of $\mathbf{d}$ is equal to the number of edges incident on the VN. The jth element of $\mathbf{d}$, denoted by $d_j \in \mathscr{E}$, specifies the edge type of the jth socket of the VN. The number of sockets of type $z \in \mathscr{E}$ of a VN is denoted by $|\mathbf{d}|_z$. The length $|\mathbf{b}|$ of $\mathbf{b}$ is equal to the number of encoded bits of the overall D-GLDPC code, the VN is associated with. Each element of $\mathbf{b}$ is equal to 1 if the corresponding bit is transmitted, and to 0 if it is punctured. Each VN is then characterized by a triplet $(v; \mathbf{b}, \mathbf{d})$, where $v \in I_V$ identifies the variable component code type so that $|\mathbf{d}| = \sum_{z \in \mathscr{E}} |\mathbf{d}|_z = q^{(v)}$ and $|\mathbf{b}| = k^{(v)}$. The set of $(v; \mathbf{b}, \mathbf{d})$ triplets is denoted by F_V. Two VNs are of the same type when they are associated with the same triplet $(v; \mathbf{b}, \mathbf{d}) \in F_V$.

The generic CN is associated with a vector $\mathbf{d}$ whose definition is analogous to that given for a VN. Again, the jth element of $\mathbf{d}$, denoted by $d_j \in \mathscr{E}$, specifies the edge type of the jth socket of the CN and $|\mathbf{d}|_z$ represents the number of sockets of type $z \in \mathscr{E}$ of the CN. Each CN is then characterized by a pair $(c; \mathbf{d})$, where $c \in I_C$ identifies the check component code type so that $|\mathbf{d}| = \sum_{z \in \mathscr{E}} |\mathbf{d}|_z = s^{(c)}$. The set of $(c; \mathbf{d})$ pairs is denoted by F_C. Two CNs are of the same type when they are associated with the same pair $(c; \mathbf{d}) \in F_C$.

We denote by $\mathscr{E}_{\mathbf{b},\mathbf{d}}^{(v)} \subseteq \mathscr{E}$ and by $\mathscr{E}_{\mathbf{d}}^{(c)} \subseteq \mathscr{E}$ the subsets of edge types connected to VNs of type $(v; \mathbf{b}, \mathbf{d})$ and to CNs of type $(c; \mathbf{d})$, respectively.

B. Ensemble Characterization

The codeword length of the D-GLDPC code (number of transmitted encoded bits) is denoted by N. For each VN type $(v; \mathbf{b}, \mathbf{d}) \in F_V$ and for each CN type $(c; \mathbf{d}) \in F_C$ we introduce the parameters $\nu_{\mathbf{b},\mathbf{d}}^{(v)}$ and $\mu_{\mathbf{d}}^{(c)}$ as follows:

$$\nu_{\mathbf{b},\mathbf{d}}^{(v)} N = \text{number of VNs of type } (v; \mathbf{b}, \mathbf{d})$$
$$\mu_{\mathbf{d}}^{(c)} N = \text{number of CNs of type } (c; \mathbf{d}).$$

The MET D-GLDPC ensemble is defined by the codeword length N and by the set of parameters $\nu_{\mathbf{b},\mathbf{d}}^{(v)}$ and $\mu_{\mathbf{d}}^{(c)}$. A code in the ensemble corresponds to a specific permutation of type-z edges, for all $z \in \mathscr{E}$.

Next, we express the fraction $\lambda_z^{(v; \mathbf{b}, \mathbf{d})}$ of type-z edges connected to VNs of type $(v; \mathbf{b}, \mathbf{d})$. The number of edges of type z connected to the VNs of type $(v; \mathbf{b}, \mathbf{d})$ is given by $N \nu_{\mathbf{b},\mathbf{d}}^{(v)} |\mathbf{d}|_z$. Then we have

$$\lambda_z^{(v; \mathbf{b}, \mathbf{d})} = \frac{\nu_{\mathbf{b},\mathbf{d}}^{(v)} |\mathbf{d}|_z}{\sum_{(v'; \mathbf{b}', \mathbf{d}') \in F_V} \nu_{\mathbf{b}',\mathbf{d}'}^{(v')} |\mathbf{d}'|_z} . \tag{1}$$

[1] While the D-GLDPC MET ensemble description presented in Section II-A and Section II-B is similar to that introduced in [1] for MET LDPC codes, our definition of $\mathbf{d}$ is different as, using generalized VNs and CNs, we need to distinguish the different sockets of a node.

Analogously, the fraction $\rho_z^{(c; \mathbf{d})}$ of type-z edges connected to CNs of type $(c; \mathbf{d})$ is equal to

$$\rho_z^{(c; \mathbf{d})} = \frac{\mu_{\mathbf{d}}^{(c)} |\mathbf{d}|_z}{\sum_{(c'; \mathbf{d}') \in F_C} \mu_{\mathbf{d}'}^{(c')} |\mathbf{d}'|_z} . \tag{2}$$

For each edge type $z \in \mathscr{E}$, the number of sockets of type z emanating from the VN set must be equal to the number of sockets of type z emanating from the CN set. This leads to the set of n_e constraints $\sum_{(v; \mathbf{b}, \mathbf{d}) \in F_V} \nu_{\mathbf{b},\mathbf{d}}^{(v)} |\mathbf{d}|_z = \sum_{(c; \mathbf{d}) \in F_C} \mu_{\mathbf{d}}^{(c)} |\mathbf{d}|_z, \forall z \in \mathscr{E}$. Moreover, the number of unpunctured code bits must sum to N. Denoting by $b = w_H(\mathbf{b})$ the Hamming weight of $\mathbf{b}$, we obtain $\sum_{(v; \mathbf{b}, \mathbf{d}) \in F_V} \nu_{\mathbf{b},\mathbf{d}}^{(v)} b = 1$.

For the generic code in the ensemble, the number K of information bits is not smaller than the difference between the total number of encoded bits (both unpunctured and punctured) and the number of parity constraints. Dividing by N we obtain

$$\frac{K}{N} \geq \sum_{(v; \mathbf{b}, \mathbf{d}) \in F_V} \nu_{\mathbf{b},\mathbf{d}}^{(v)} k^{(v)} - \sum_{(c; \mathbf{d}) \in F_C} \mu_{\mathbf{d}}^{(c)} m^{(c)} \tag{3}$$

where the right-hand side (RHS) of (3) is the design rate of the ensemble, denoted by R.

III. EXIT ANALYSIS OF MET D-GLDPC CODES OVER THE BEC

Every VN receives observables from the communication channel and messages from the CN set along the edges incident on it. Similarly, every CN receives messages from the VN set along the edges incident on it. The channel over which messages are exchanged between VNs and CNs is called the *extrinsic channel* [7]. In a MET framework, we have to distinguish between n_e parallel extrinsic channels, each one associated with a specific edge type $z \in \mathscr{E}$. A VN of type $(v; \mathbf{d}, \mathbf{b})$ receives messages from the extrinsic channels corresponding to the edge types in $\mathscr{E}_{\mathbf{b},\mathbf{d}}^{(v)}$, and a CN of type $(c; \mathbf{d})$ from the extrinsic channels corresponding to the edge types in $\mathscr{E}_{\mathbf{d}}^{(c)}$.

We denote by $I_{E(z)(v; \mathbf{b}, \mathbf{d})}$ and $I_{E(z)(c; \mathbf{d})}$ the average extrinsic information values outgoing from the type-z sockets of a VN of type $(v; \mathbf{b}, \mathbf{d})$ and CN of type $(c; \mathbf{d})$, respectively. These are defined as

$$I_{E(z)(v; \mathbf{b}, \mathbf{d})} = \frac{1}{|\mathbf{d}|_z} \sum_{j : d_j = z} I(V_j; E_j) \tag{4}$$

$$I_{E(z)(c; \mathbf{d})} = \frac{1}{|\mathbf{d}|_z} \sum_{j : d_j = z} I(V_j; E_j) \tag{5}$$

where the random variable (r.v.) V_j represents the bit associated with the jth socket of the VN or CN and the r.v. E_j the extrinsic message outgoing from the node along its jth socket. Similarly, the average a priori information values incoming along the type-z sockets of a VN of type $(v; \mathbf{b}, \mathbf{d})$ and CN of type $(c; \mathbf{d})$ are defined as

$$I_{A(z)(v; \mathbf{b}, \mathbf{d})} = \frac{1}{|\mathbf{d}|_z} \sum_{j : d_j = z} I(V_j; W_j) \tag{6}$$

$$I_{A(z)(c; \mathbf{d})} = \frac{1}{|\mathbf{d}|_z} \sum_{j : d_j = z} I(V_j; W_j) \tag{7}$$

respectively, where the r.v. W_j represents the incoming message along the jth socket.

For each edge type $z \in \mathscr{E}$, the outgoing extrinsic information values along the type-z edges, and averaged over all such edges, are given by

$$I_{E,V(z)} = \sum_{(v;\mathbf{b},\mathbf{d}) \in F_V} \lambda_z^{(v;\mathbf{b},\mathbf{d})} I_{E(z)(v;\mathbf{b},\mathbf{d})} \tag{8}$$

$$I_{E,C(z)} = \sum_{(c;\mathbf{d}) \in F_C} \rho_z^{(c;\mathbf{d})} I_{E(z)(c;\mathbf{d})} \tag{9}$$

for the VN set and the CN set, respectively. Moreover, the corresponding a priori information values are given by

$$I_{A,V(z)} = \sum_{(v;\mathbf{b},\mathbf{d}) \in F_V} \lambda_z^{(v;\mathbf{b},\mathbf{d})} I_{A(z)(v;\mathbf{b},\mathbf{d})} \tag{10}$$

$$I_{A,C(z)} = \sum_{(c;\mathbf{d}) \in F_C} \rho_z^{(c;\mathbf{d})} I_{A(z)(c;\mathbf{d})} . \tag{11}$$

In general, $I_{E,V(z)}$ is a function of the communication channel parameter and of all $I_{A,C(t)}$ values for $t \in \mathscr{E}$. Similarly, $I_{E,C(z)}$ is a function of all $I_{A,V(t)}$ values for $t \in \mathscr{E}$.

Next, let us assume that the communication channel is a BEC with erasure probability ϵ. In this case, the zth extrinsic channel is also a BEC. The average erasure probability of messages incoming towards the VN set along the type-z edges will be denoted by p_z. It is readily shown that for all $z \in \mathscr{E}$, we have $p_z = 1 - I_{A,V(z)}$, where $I_{A,V(z)}$ is given in (10). Analogously, the average erasure probability of messages incoming towards the CN set along the type-z is $p_z = 1 - I_{A,C(z)}$, where $I_{A,c(z)}$ is given in (11). In general, we have $I_{E,V(z)} = I_{E,V(z)}(p_1, p_2, \ldots, p_{n_e}, \epsilon)$ and $I_{E,C(z)} = I_{E,C(z)}(p_1, p_2, \ldots, p_{n_e})$.

EXIT analysis allows to infer if a certain value ϵ of the BEC erasure probability is smaller or larger than the asymptotic threshold ϵ^* for a given ensemble. It consists of updating iteratively the n_e average extrinsic information values $I_{E,V(z)}$ and $I_{E,C(z)}$, until a maximum number $N_{\text{it}}^{\max}$ of iterations has been reached or a stopping criterion has been satisfied. Specifically, at the beginning it sets $I_{E,A(z)} = 0$ (equivalently, $p_z = 1$) for all $z \in \mathscr{E}$. At each iteration, the average extrinsic information $I_{E,V(z)}$ is first calculated for all $z \in \mathscr{E}$ and each such value interpreted as the a-priori value $I_{A,C(z)}$. Then, the average extrinsic information $I_{E,C(z)}$ is calculated for all $z \in \mathscr{E}$ and each such value interpreted as the a-priori value $I_{A,V(z)}$. At the end of the generic iteration, assume that for all $(v; \mathbf{b}, \mathbf{d}) \in F_V$, a VN of type $(v; \mathbf{b}, \mathbf{d})$ has an information set that corresponds to sockets of edge types over which the current value of p_z fulfills $p_z < \bar{p}$, for some (small) $\bar{p}$. Then, the iterative process is stopped and ϵ is declared to be *achievable*, i.e., $\epsilon < \epsilon^*$. Otherwise, a new iteration is run. If the previous condition is not yet fulfilled at the end of the last iteration, ϵ is declared to be *non-achievable*, i.e., $\epsilon > \epsilon^*$. In this way, an upper and a lower bound on ϵ^* converging to each other can be derived and ϵ^* estimated.[2]

From (8) and (9), we see that in order to calculate $I_{E,V(z)}$ and $I_{E,C(z)}$, it is necessary to express the average extrinsic

information outgoing along the type-z edges of a VN of type $(v; \mathbf{b}, \mathbf{d})$ and CN of type $(c; \mathbf{d})$, respectively. This problem is addressed in Section IV and Section V.

IV. MULTI-TYPE INFORMATION FUNCTIONS

In this section we introduce multi-type information functions and their split counterparts. Given a linear block code $\mathcal{C}(n, k)$, define $\mathbf{S}_g$ as a $(k \times g)$ matrix composed of g columns of the generator matrix $\mathbf{G}$ used to represent $\mathcal{C}$ (irrespective of the order of these g columns), let $\sum_{\mathbf{S}_g}$ denote the summation over all possible $\binom{n}{g}$ choices of $\mathbf{S}_g$ and let $\mathrm{R}(\mathbf{S}_g)$ denote the rank of $\mathbf{S}_g$. Then, the gth (unnormalized) information function of $\mathcal{C}$, for $g \in \{0, 1, \ldots, n\}$, is defined as $\tilde{e}_g = \sum_{\mathbf{S}_g} \mathrm{R}(\mathbf{S}_g)$. The information function is independent of the specific choice of $\mathbf{G}$. Similarly, the (g, u)th (unnormalized) split information function of $\mathcal{C}$, for $g \in \{0, 1, \ldots, n\}$ and $u \in \{0, 1, \ldots, k\}$, is defined as $\tilde{e}_{g,u} = \sum_{\mathbf{S}_{g,u}} \mathrm{R}(\mathbf{S}_{g,u})$, where $\mathbf{S}_{g,u}$ is a $(k \times (g+u))$ matrix composed of g columns of $\mathbf{G}$ and u columns of $\mathbf{I}_k$, the identity matrix of order k, and $\sum_{\mathbf{S}_{g,u}}$ denotes the summation over all possible $\binom{n}{g}\binom{k}{h}$ choices of $\mathbf{S}_{g,u}$. As opposed to $\tilde{e}_g$, the split information function $\tilde{e}_{g,u}$ depends on the choice of $\mathbf{G}$, i.e. on the mapping between information words and codewords.

Next, assume that the code bit positions of $\mathcal{C}$ are of different types $e \in \{e_1, e_2, \ldots, e_E\}$. For each $\alpha \in \{1, 2, \ldots, E\}$ let us denote by n_{e_α} the number of bit positions of type e_α, so that $\sum_{\alpha=1}^{E} n_{e_\alpha} = n$. From the generator matrix $\mathbf{G}$ chosen to represent $\mathcal{C}$, let us form E matrices $\mathbf{G}_{e_\alpha}$, where $\mathbf{G}_{e_\alpha}$ is the $(k \times n_{e_\alpha})$ matrix composed of the columns of $\mathbf{G}$ associated with the bit positions of type e_α (irrespective of the order of these columns). Furthermore, let $\mathbf{b}$ be a binary vector of length k and let $b = w_H(\mathbf{b})$.

We define the $(g_{e_1}, \ldots, g_{e_E})$th multi-type information function of $\mathcal{C}$ as

$$\tilde{e}_{g_{e_1}, \ldots, g_{e_E}} = \sum_{\mathbf{S}_{g_{e_1}, \ldots, g_{e_E}}} \mathrm{R}(\mathbf{S}_{g_{e_1}, \ldots, g_{e_E}}) \tag{12}$$

where $\mathbf{S}_{g_{e_1}, \ldots, g_{e_E}}$ is a matrix formed by selecting g_{e_α} columns in $\mathbf{G}_{e_\alpha}$ (for all $\alpha \in \{1, 2, \ldots, E\}$ and irrespective of the positions of these columns) and where $\sum_{\mathbf{S}_{g_{e_1}, \ldots, g_{e_E}}}$ denotes the summation over all $\prod_{\alpha=1}^{E} \binom{n_{e_\alpha}}{g_{e_\alpha}}$ matrices $\mathbf{S}_{g_{e_1}, \ldots, g_{e_E}}$.

Moreover, we define the $(g_{e_1}, \ldots, g_{e_E}, u; \mathbf{b})$th multi-type split information function as

$$\tilde{e}_{g_{e_1}, \ldots, g_{e_E}, u; \mathbf{b}} = \sum_{\mathbf{S}_{g_{e_1}, \ldots, g_{e_E}, u; \mathbf{b}}} \mathrm{R}(\mathbf{S}_{g_{e_1}, \ldots, g_{e_E}, u; \mathbf{b}}) \tag{13}$$

where $\mathbf{S}_{g_{e_1}, \ldots, g_{e_E}, u; \mathbf{b}}$ is a matrix formed by selecting g_{e_α} columns in $\mathbf{G}_{e_\alpha}$ (for all $\alpha \in \{1, 2, \ldots, E\}$ and irrespective of the positions of these columns) and u columns among the b columns of $\mathbf{I}_k$ corresponding to the support of $\mathbf{b}$. In (13), $\sum_{\mathbf{S}_{g_{e_1}, \ldots, g_{e_E}, u; \mathbf{b}}}$ denotes the summation over all $\binom{b}{u} \prod_{\alpha=1}^{E} \binom{n_{e_\alpha}}{g_{e_\alpha}}$ matrices $\mathbf{S}_{g_{e_1}, \ldots, g_{e_E}, u; \mathbf{b}}$.

Again, while the multi-type information functions of $\mathcal{C}$ are independent on the specific choice of the generator matrix $\mathbf{G}$, the multi-type split information functions of $\mathcal{C}$ depend on the code representation. They also depend on the specific choice of the binary vector $\mathbf{b}$.

---

[2] A small $\bar{p}$ and a large $N_{\text{it}}^{\max}$ are desirable to obtain a good estimate of ϵ^*.

$$I_{E(e_\alpha)} = 1 - \frac{1}{q_{e_\alpha}} \sum_{h=0}^{b} \epsilon^h (1-\epsilon)^{b-h} \sum_{t_{e_1}=0}^{q_{e_1}} p_{e_1}^{t_{e_1}} (1-p_{e_1})^{q_{e_1}-t_{e_1}} \cdots \sum_{t_{e_\alpha}=0}^{q_{e_\alpha}-1} p_{e_\alpha}^{t_{e_\alpha}} (1-p_{e_\alpha})^{q_{e_\alpha}-1-t_{e_\alpha}} \cdots \sum_{t_{e_E}=0}^{q_{e_E}} p_{e_E}^{t_{e_E}} (1-p_{e_E})^{q_{e_E}-t_{e_E}}$$
$$\times a_{(\mathbf{t},h;\mathbf{b}),e_\alpha}$$

In the next section we show how the parameters introduced in (12) and (13) may be exploited to calculate the average extrinsic information outgoing from the type-e_α edges of VNs and CNs, in the context of EXIT analysis of MET D-GLDPC codes over the BEC.

V. EXTRINSIC INFORMATION FOR VNS AND CNS

A. Extrinsic Information for the VNs

For a VN of type $(v; \mathbf{b}, \mathbf{d})$, we aim to develop the expression of the average extrinsic information outgoing along the edges of a certain type. Hereafter, for the sake of simplicity we omit the subscript $(v; \mathbf{b}, \mathbf{d})$ in the extrinsic information given by (4), and denote the VN length, dimension and generator matrix simply by q, k and $\mathbf{G}$, respectively. Moreover, we let $\mathscr{E}_{\mathbf{b},\mathbf{d}}^{(v)} = \{e_1, e_2, \ldots, e_E\}$, $q_{e_f} = |\mathbf{d}|_{e_f}$ (so that $\sum_{f=1}^{E} q_{e_f} = q$), $t(j)$ be the edge type of the jth socket of the VN, and $b = w_H(\mathbf{b})$.

Proposition 1: Consider a VN of type $(v; \mathbf{b}, \mathbf{d})$ and assume that the associated component code, of type v, has no idle bits. For all $e_f \in \mathscr{E}_{\mathbf{b},\mathbf{d}}^{(v)}$, let $p_{e_f} = 1 - I_{A,V(e_f)}$ be the average erasure probability of the incoming messages over the edges of type e_f. If maximum *a posteriori* (MAP) decoding is used at the VN, then the average extrinsic information over the edges of type e_α is given by the expression at the top of the page, where $\mathbf{t} = \{t_{e_1}, t_{e_2}, \ldots, t_{e_E}\}$ and where

$$a_{(\mathbf{t},h;\mathbf{b}),e_\alpha} = (q_{e_\alpha} - t_{e_\alpha}) \tilde{e}_{q_{e_1}-t_{e_1},\ldots,q_{e_E}-t_{e_E},b-h;\mathbf{b}}$$
$$- (t_{e_\alpha}+1) \tilde{e}_{q_{e_1}-t_{e_1},\ldots,q_{e_\alpha}-t_{e_\alpha}-1,\ldots,q_{e_E}-t_{e_E},b-h;\mathbf{b}}.$$

Proof: The proof is provided in the case where $\mathscr{E}_{\mathbf{b},\mathbf{d}}^{(v)} = \{e_1, e_2\}$, assuming $e_\alpha = e_1$. The extension to the general case is straightforward. Since the component code has no idle bits

and local MAP decoding is used, from [7, Proposition 1] we may develop $I_{E(e_1)}$ as

$$I_{E(e_1)} = 1 - \frac{1}{q_{e_1}} \sum_{j:t(j)=e_1} H(V_j|\mathbf{Y}, \mathbf{W}_{e_1[j]}, \mathbf{W}_{e_2}) \quad (14)$$

where $\mathbf{Y}$ is the random vector of the k channel observables for the VN, $\mathbf{W}_z$ is the random vector of the q_z messages received by the VN along the edges of type $z \in \{e_1, e_2\}$, and $\mathbf{W}_{z[j]}$ is $\mathbf{W}_z$ except its jth element.

Denote by $\mathcal{T}$ the set of non-erasure positions of a realization $\mathbf{y}$ and by $\mathcal{S}_z$ the set of non-erasure positions of a realization $\mathbf{w}_z$ for $z \in \{e_1, e_2\}$. Note that $|\mathcal{T}| \leq b$ as the punctured bits shall be regarded as erasures. Define as $\mathbf{T}$ the matrix composed of the columns of $\mathbf{I}_k$ corresponding to indexes in $\mathcal{T}$ and as $\mathbf{S}_z$ the matrix composed of the columns of $\mathbf{G}$ corresponding to indexes in $\mathcal{S}_z$. Moreover, define as $\mathbf{j}$ the column of $\mathbf{G}$ corresponding to index j. Assuming independence between the messages received from the extrinsic channel, (14) can be developed as shown at the bottom of the page, where (a) follows from $H(X|Y) = \mathbb{E}_y[H(X|Y = y)]$ and (b) from message independence. Next, concerning equality (b) note that

$$\sum_{j:t(j)=e_1} \sum_{|\mathcal{T}|=u} \sum_{|\mathcal{S}_{e_1}|=i_{e_1}-1,j\notin\mathcal{S}_{e_1}} \sum_{|\mathcal{S}_{e_2}|=i_{e_2}} [R(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2}|\mathbf{j})$$
$$- R(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2})]$$
$$= \sum_{|\mathcal{T}|=u} \sum_{|\mathcal{S}_{e_2}|=i_{e_2}} \Bigg(\sum_{j:t(j)=e_1} \sum_{|\mathcal{S}_{e_1}|=i_{e_1},j\in\mathcal{S}_{e_1}} R(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2})$$
$$- \sum_{j:t(j)=e_1} \sum_{|\mathcal{S}_{e_1}|=i_{e_1}-1,j\notin\mathcal{S}_{e_1}} R(\mathbf{T}|\mathbf{S}_{e_2}|\mathbf{S}_{e_2}) \Bigg)$$
$$= \sum_{|\mathcal{T}|=u} \sum_{|\mathcal{S}_{e_2}|=i_{e_2}} \Bigg(i_{e_1} \sum_{|\mathcal{S}_{e_1}|=i_{e_1}} R(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2})$$

$$I_{E(e_1)} \stackrel{(a)}{=} 1 - \frac{1}{q_{e_1}} \sum_{j:t(j)=e_1} \sum_{\mathcal{T}} \sum_{\mathcal{S}_{e_1}:j\notin\mathcal{S}_{e_1}} \sum_{\mathcal{S}_{e_2}} \Pr(\mathcal{T}) \Pr(\mathcal{S}_{e_1}) \Pr(\mathcal{S}_{e_2}) [R(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2}|\mathbf{j}) - R(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2})]$$

$$= 1 - \frac{1}{q_{e_1}} \sum_{j:t(j)=e_1} \sum_{u=0}^{b} \sum_{|\mathcal{T}|=u} \Pr(|\mathcal{T}|=u) \sum_{i_{e_1}=1}^{q_{e_1}} \sum_{|\mathcal{S}_{e_1}|=i_{e_1}-1,j\notin\mathcal{S}_{e_1}} \Pr(|\mathcal{S}_1|=i_{e_1}-1)$$

$$\times \sum_{i_{e_2}=0}^{q_{e_2}} \sum_{|\mathcal{S}_{e_2}|=i_{e_2}} \Pr(|\mathcal{S}_{e_2}|=i_{e_2}) [R(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2}|\mathbf{j}) - R(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2})]$$

$$\stackrel{(b)}{=} 1 - \frac{1}{q_{e_1}} \sum_{u=0}^{b} \epsilon^{b-u}(1-\epsilon)^u \sum_{i_{e_1}=1}^{q_{e_1}} p_{e_1}^{q_{e_1}-i_{e_1}}(1-p_{e_1})^{i_{e_1}-1} \sum_{i_{e_2}=0}^{q_{e_2}} p_{e_2}^{q_{e_2}-i_{e_2}}(1-p_{e_2})^{i_{e_2}}$$

$$\times \sum_{j:t(j)=e_1} \sum_{|\mathcal{T}|=u} \sum_{|\mathcal{S}_{e_1}|=i_{e_1}-1,j\notin\mathcal{S}_{e_1}} \sum_{|\mathcal{S}_{e_2}|=i_{e_2}} [R(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2}|\mathbf{j}) - R(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2})].$$

$$I_{E(e_\alpha)} = 1 - \frac{1}{s_{e_\alpha}} \sum_{t_{e_1}=0}^{s_{e_1}} p_{e_1}^{t_{e_1}} (1-p_{e_1})^{s_{e_1}-t_{e_1}} \cdots \sum_{t_{e_\alpha}=0}^{s_{e_\alpha}-1} p_{e_\alpha}^{t_{e_\alpha}} (1-p_{e_\alpha})^{s_{e_\alpha}-1-t_{e_\alpha}} \cdots \sum_{t_{e_E}=0}^{s_{e_E}} p_{e_E}^{t_{e_E}} (1-p_{e_E})^{s_{e_E}-t_{e_E}} \, a_{\mathbf{t},e_\alpha}$$
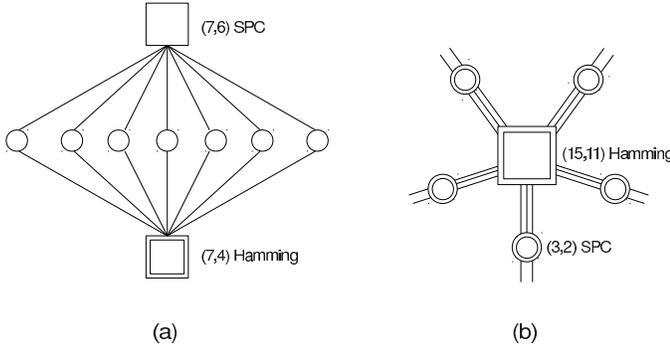

Fig. 1. (a): $R = 3/7$ GLDPC protograph. All VNs are repetition-2 codes, one CN is a $(7,4)$ Hamming code and the other a $(7,6)$ SPC code. (b): $R = 3/5$ D-GLDPC protograph. All VNs are $(3,2)$ SPC codes and the CN is a $(15,11)$ Hamming code.

$$- (q_{e_1} - i_{e_1} + 1) \sum_{|\mathcal{S}_{e_1}|=i_{e_1}-1} \mathsf{R}\left(\mathbf{T}|\mathbf{S}_{e_1}|\mathbf{S}_{e_2}\right) \Big)$$

$$= i_{e_1} \tilde{e}_{i_{e_1},i_{e_2},u;\mathbf{b}} - (q_{e_1} - i_{e_1} + 1) \tilde{e}_{i_{e_1}-1,i_{e_2},u;\mathbf{b}}$$

where, $\tilde{e}_{i_{e_1},\ldots,i_{e_E},u;\mathbf{b}}$ is the $(i_{e_1},\ldots,i_{e_E},u;\mathbf{b})$th MET split information function introduced in Section IV. The proof is completed by $t_{e_f} = q_{e_f} - i_{e_f}$, $f \in \{1,2\}$, and $h = b - u$. ∎

As expected, for a degree-1 VN the developed formula yields $I_E = 1 - \epsilon$ (unpunctured) and $I_E = 1$ (punctured).

B. Extrinsic Information for the CNs

Consider now a CN of type $(c;\mathbf{d})$, and let $\mathcal{E}_{\mathbf{d}}^{(c)} = \{e_1, e_2, \ldots, e_E\}$. The average extrinsic information outgoing along type-e_α edges may be easily obtained from that developed for a VN in the limit case $\epsilon \to 1$. Denoting the CN length simply by s, and letting $s_{e_f} = |\mathbf{d}|_{e_f}$ (so that $\sum_{f=1}^{E} s_{e_f} = s$), we obtain the expression at the top of the page, where

$$a_{\mathbf{t},e_\alpha} = (s_{e_\alpha} - t_{e_\alpha}) \tilde{e}_{s_{e_1}-t_{e_1},\ldots,s_{e_E}-t_{e_E}}$$
$$- (t_{e_\alpha} + 1) \tilde{e}_{s_{e_1}-t_{e_1},\ldots,s_{e_\alpha}-t_{e_\alpha}-1,\ldots,s_{e_E}-t_{e_E}} \quad (15)$$

and $\tilde{e}_{i_{e_1},\ldots,i_{e_E}}$ is the $(i_{e_1},\ldots,i_{e_E})$th MET information function introduced in Section IV.

VI. EXAMPLES

In this section, the thresholds over the BEC for some example MET ensembles are analyzed using the developed tool. All thresholds have been calculated setting $\bar{p} = 10^{-7}$.

Example 1 (GLDPC ensemble): Consider a 2-edge type ensemble where the VN set is composed of length-2 repetition codes and the CN set of a mixture of length-7 SPC codes and $(7,4)$ Hamming codes, such that every VN is checked by exactly one SPC CN and one Hamming CN. We have $\mathcal{E} = \{1,2\}$ (where any edge incident on a Hamming CN is of type 1, and is of type 2 otherwise), $I_V = \{1\}$ and $I_C = \{1,2\}$

(where 1 is associated with Hamming codes and 2 with SPC codes). All VNs are of the same type $(v;\mathbf{b},\mathbf{d}) = (1;[1],[1,2])$. Letting $\mathbf{d}_H = [1,1,1,1,1,1,1]$ and $\mathbf{d}_{SPC} = [2,2,2,2,2,2,2]$, we have $\mu_{\mathbf{d}_H}^{(1)} = 1/7$, $\mu_{\mathbf{d}_{SPC}}^{(2)} = 1/7$, $\nu_{[1],[1,2]}^{(1)} = 1$ and $R = 3/7$ by (3). For this ensemble we have $\epsilon^* = 0.4944$ for $N_{it}^{max} = 100$ and $\epsilon^* = 0.4960$ for $N_{it}^{max} = 1000$.

Interestingly, these threshold values are quite better than $\epsilon^* = 0.3293$, the threshold obtained for an unstructured ensemble with the same component codes in the same proportion. This suggests that by imposing the constraint that each VN is checked by a Hamming CN and by a SPC CN, the unstructured ensemble is expurgated from bad codes (at least in the sense of threshold). We also analyzed the protograph GLDPC ensemble whose protograph is depicted in Fig. 1(a), which may be seen as a further expurgation of the 2-edge type ensemble. For this protograph ensemble we obtained slightly better thresholds, $\epsilon^* = 0.4959$ for $N_{it}^{max} = 100$ and $\epsilon^* = 0.4974$ for $N_{it}^{max} = 1000$. In this case, however, 14 extrinsic information values have to be tracked instead of 2.

We also observe that an $R = 1/2$ ensemble may be obtained by puncturing any of the VNs in the protograph. For example, puncturing the VN associated with the column $[1000]^T$ of the Hamming CN generator matrix (in systematic form), the proposed EXIT analysis tool returned $\epsilon^* = 0.4244$ for $N_{it}^{max} = 1000$.

Example 2 (D-GLDPC ensemble): We applied the proposed analysis technique to the protograph D-GLDPC ensemble whose protograph is depicted in Fig. 1(b). All VNs are $(3,2)$ SPC codes with generator matrix $\mathbf{G} = [101,011]$, while the CN is a $(15,11)$ Hamming code. The ensemble design rate is $R = 3/5$. For this protograph ensemble, the proposed EXIT analysis (which in this case requires tracking 15 extrinsic information values) returned $\epsilon^* = 0.3524$ for $N_{it}^{max} = 100$ and $\epsilon^* = 0.3528$ for $N_{it}^{max} = 1000$. Interestingly, we found that these threshold values can be improved by replacing one of the SPC VNs in the protograph with a $(3,2)$ VN with local minimum distance 1. For example, if the VN connected to the sockets of the Hamming CN corresponding to the columns $[00000000010]^T, [00000000001]^T, [10111001011]^T$ of its generator matrix $\mathbf{G}_H$ (in systematic form) is replaced with a $(3,2)$ code represented by the matrix $\mathbf{G} = [111,100]$, then we obtain $\epsilon^* = 0.3603$ for $N_{it}^{max} = 1000$.

A rate $R = 2/3$ ensemble can be obtained by puncturing one bit in the previous ensemble. For example, puncturing the bit associated with the column $[10]^T$ of $\mathbf{I}_2$ in the VN connected to the sockets of $\mathbf{G}_H$ corresponding to the columns $[11011010101]^T, [11101100110]^T, [11110111000]^T$, we obtain $\epsilon^* = 0.2849$ for $N_{it}^{max} = 1000$. Again, this threshold can be improved to $\epsilon^* = 0.2913$ by replacing the above mentioned $(3,2)$ SPC VN with the same VN with minimum distance 1.

VII. Conclusion

In this paper, a technique for threshold evaluation of MET D-GLDPC codes over the BEC has been proposed and its effectiveness illustrated through examples. The technique is based on the concept of EXIT analysis and exploits multi-type information functions and split information functions of linear block codes. It is expected to be adopted to perform the analysis step within optimization algorithms for MET D-GLDPC code ensemble design.

Appendix: Variable and Check Nodes with Minimum Distance 1

As illustrated in Example 2 in Section VI, VNs with minimum distance 1 may be beneficial in terms of threshold (in a similar way as degree-1 VNs for LDPC codes). Although these VNs cannot be included in standard EXIT chart analysis of unstructured D-GLDPC ensembles, they can be captured by EXIT analysis in a MET context, provided they are judiciously connected to the rest of the graph. Next, a unified treatment is provided for distance-1 VNs and CNs.

Consider a VN of type $(v; \mathbf{b}, \mathbf{d})$, length q, dimension k and assume it has minimum distance 1. Again, let $\mathscr{E}_{\mathbf{b},\mathbf{d}}^{(v)} = \{e_1, e_2, \ldots, e_E\}$, $q_{e_f} = |\mathbf{d}|_{e_f}$ and $b = w_H(\mathbf{b})$. Let $\mathbf{G}$ be the $(k \times q)$ generator matrix chosen to represent the VN and $\mathbf{J}$ be a $(k \times b)$ matrix composed of the columns of $\mathbf{I}_k$ that correspond to the unpunctured bits of the VN. Denote by $\mathbf{S}_{q-1}^{(j)}$ a submatrix formed by $q-1$ columns of $\mathbf{G}$ except the jth column, by $\mathbf{J}_z$ a submatrix formed by z columns of $\mathbf{J}$, and let $\mathbf{S}_{q-1,z}^{(j)} = [\mathbf{S}_{q-1}^{(j)}|\mathbf{J}_z]$. Note that $\mathsf{R}(\mathbf{S}_{q-1}^{(j)}) = k - 1$ when the jth column of $\mathbf{G}$ corresponds to the support of a weight-1 codeword of the VN and $\mathsf{R}(\mathbf{S}_{q-1}^{(j)}) = k$ otherwise [3, Proposition 2].

Any weight-1 codeword of the VN whose support corresponds to a socket of type e_f will be referred to as a weight-1 codeword of type e_f. We denote by A_{1,e_f} the number of weight-1 codewords of type e_f. Any weight-1 codeword, such that there exists at least one column in $\mathbf{J}$ that is linearly independent of the $q-1$ columns of $\mathbf{G}$ not associated with the support of the codeword, is said to be a *split-dependent* weight-1 codeword, and a *split-independent* weight-1 codeword otherwise. We denote by A'_{1,e_f} the number of split-dependent weight-1 codewords of type e_f and and by A''_{1,e_f} the number of split-independent weight-1 codewords of type e_f, so that $A_{1,e_f} = A'_{1,e_f} + A''_{1,e_f}$.

Next, for $j \in \{1, 2, \ldots, q\}$ let $\omega(j) = 1$ if the jth column of $\mathbf{G}$ corresponds to the support of a split-dependent weight-1 codeword, and $\omega(j) = 0$ otherwise. Moreover, consider the indicator function $\chi(\mathbf{S}_{q-1,z}^{(j)}) = k - \mathsf{R}(\mathbf{S}_{q-1,z}^{(j)})$.

Proposition 2: The average extrinsic information outgoing from the type-e_α edges of a VN fulfills

$$I_{E(e_\alpha)} \to 1 - \frac{g_{1,e_\alpha}(\epsilon) + A''_{1,e_\alpha}}{q_{e_\alpha}} \qquad (16)$$

as $p_{e_f} \to 0$ for all $f \in \{1, 2, \ldots, E\}\setminus\{\alpha\}$, where

$$g_{1,e_\alpha}(\epsilon) = \sum_{h=1}^{b} \epsilon^h (1-\epsilon)^{b-h} \sum_{j:\{t(j)=e_\alpha, \omega(j)=1\}} \sum_{\mathbf{S}_{q-1,z}^{(j)}} \chi(\mathbf{S}_{q-1,z}^{(j)}).$$

The proof of Proposition 2 is not included in the paper due to space constraints. Note that only the split-dependent codewords of weight-1 contribute to $g'_{1,e_\alpha}(\epsilon)$ in (16), while only the split-independent ones contribute to A''_{e_α}. For $\epsilon \to 1$ a similar result may be developed for a CN with minimum distance 1. In this case, (16) is replaced by

$$I_{E(e_\alpha)} \to 1 - \frac{A_{1,e_\alpha}}{q_{e_\alpha}}. \qquad (17)$$

As evident from Proposition 2 (and from the CN counterpart (17)), the average extrinsic information outgoing from a $d_{\min} = 1$ VN or CN along type-z sockets including the support of a weight-1 codeword is always bounded away from 1. As a consequence, the overall average extrinsic information values over type-z edges, expressed by (8) and (9), cannot converge to 1. This may in some cases jeopardize fulfilling of the stopping criterion described in Section III for any value of ϵ. For this reason, the use of $d_{\min} = 1$ VNs (and CNs) into MET D-GLDPC ensembles must be carefully controlled.

Acknowledgment

References

[1] T. Richardson and R. Urbanke, "Multi-edge type LDPC codes," LTHC-Report-2004-001, 2004.

[2] J. Thorpe, "Low density parity check (LDPC) codes constructed from protographs," JPL INP Progress Report 42-154, 2003.

[3] E. Paolini, M. Fossorier, and M. Chiani, "Doubly-generalized LDPC codes: Stability bound over the BEC," *IEEE Trans. Inf. Theory*, vol. 55, no. 3, pp. 1027–1046, Mar. 2009.

[4] Y. Wang and M. Fossorier, "Doubly generalized LDPC codes over the AWGN channel," *IEEE Trans. Commun.*, vol. 57, no. 5, pp. 1312–1319, May 2009.

[5] E. Paolini, M. Fossorier, and M. Chiani, "Generalized and doubly generalized LDPC codes with random component codes for the binary erasure channel," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1651–1672, Apr. 2010.

[6] G. Liva and M. Chiani, "Protograph LDPC codes design based on EXIT analysis," in *in Proc. of 2007 IEEE Global Telecommunications Conf.*, Washington, DC, USA, Nov. 2007, pp. 3250–3254.

[7] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: Model and erasure channel properties," *IEEE Trans. Inf. Theory*, vol. 50, no. 11, pp. 2657–2673, Nov. 2004.

[8] T. Helleseth, T. Kløve, and V. I. Levenshtein, "On the information function of an error-correcting code," *IEEE Trans. Inf. Theory*, vol. 43, pp. 549–557, Mar. 1997.

[9] M. Lentmaier, M. Tavares, and G. Fettweis, "Exact erasure channel density evolution for protograph-based generalized LDPC codes," in *Proc. of 2009 IEEE Int. Symp. on Information Theory*, Seoul, Korea, Jul. 2009, pp. 566–570.

[10] T. Awano, K. Kasai, T. Shibuya, and K. Sakaniwa, "Three-edge type LDPC code ensembles with exponentially few codewords with linear small weight," in *Proc. of 2008 Int. Symp. on Information Theory and its Applications*, Auckland, New Zealand, Dec. 2008, pp. 1–6.

[11] K. Kasai, C. Poulliat, K. Sakaniwa, T. Awano, and D. Declercq, "Weight distributions of multi-edge type LDPC codes," in *Proc. of 2009 IEEE Int. Symp. on Information Theory*, Seoul, Korea, Jul. 2009, pp. 60–64.

[12] C.-L. Wang, S. Lin, and M. Fossorier, "On asymptotic ensemble weight enumerators of multi-edge type codes," in *Proc. of 2009 IEEE Global Telecommunications Conf.*, Honolulu, HI, USA, Nov.Dec. 2009, pp. 1–6.

[13] S. Abu-Surra, W. Ryan, and D. Divsalar, "Ensemble enumerators for protograph-based generalized LDPC codes," in *Proc. of 2007 IEEE Global Telecommunications Conf.*, Washington, DC, USA, Nov. 2007, pp. 1492–1497.