Doubly-Generalized LDPC Codes: Stability Bound Over the BEC

Enrico Paolini, *Member, IEEE*, Marc P. C. Fossorier, *Fellow, IEEE*, and Marco Chiani, *Senior Member, IEEE*

***Abstract*—The iterative decoding threshold of low-density parity-check (LDPC) codes over the binary erasure channel (BEC) fulfills an upper bound depending only on the variable and check nodes with minimum distance 2. This bound is a consequence of the stability condition, and is here referred to as stability bound. In this paper, a stability bound over the BEC is developed for doubly-generalized LDPC codes, where variable and check nodes can be generic linear block codes, assuming maximum *a posteriori* erasure correction at each node. It is proved that also in this generalized context the bound depends only on the variable and check component codes with minimum distance 2. A condition is also developed, namely, the derivative matching condition, under which the bound is achieved with equality. The stability bound leads to consider single parity-check codes used as variable nodes as an appealing option to overcome common problems created by generalized check nodes.**

***Index Terms*—Binary erasure channel (BEC), error-correcting codes, extrinsic information transfer (EXIT) chart, information function, low-density parity-check (LDPC) codes, stability condition.**

I. Introduction

LOW-density parity-check (LDPC) codes [1] have been intensively studied in the last decade due to their capability to approach the Shannon limit under iterative belief-propagation decoding. An LDPC code of length N and dimension K can be graphically represented as a bipartite graph, known also as Tanner graph, with N variable nodes (VNs) and $M \geq N-K$ check nodes (CNs) [2]. The degree of either a VN or CN is defined as the number of edges connected to it. A degree-n VN of an LDPC code can be interpreted as a length-n repetition code, i.e., as an $(n,1)$ linear block code repeating n times its single information bit towards the CN set. A degree-n CN of an LDPC code can be interpreted as a length-n single parity-check (SPC) code, i.e., as a $(n,n-1)$ linear block code associated with one parity equation.

Manuscript received February 06, 2008; revised August 25, 2008. Current version published February 25, 2009. This work was supported in part by the European Community under Seventh Framework Program grant agreement ICT OPTIMIX n.INFSO-ICT-214625. The material in this paper was presented in part at the IEEE International Symposium on Information Theory (ISIT), Nice, France, June 2007.

E. Paolini and M. Chiani are with DEIS/WiLAB, University of Bologna, 47023 Cesena (FC), Italy (e-mail: e.paolini@unibo.it; marco.chiani@unibo.it).

M. P. C. Fossorier is with ETIS ENSEA/UCP/CNRS UMR-80516, Cergy, Pontoise, France (e-mail: mfossorier@ieee.org).

Communicated by I. Sason, Associate Editor for Coding Theory.

Color versions of Figures 6 and 7 in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2008.2011446

An extension of the concept of LDPC code is represented by doubly-generalized LDPC (D-GLDPC) codes [3], where the VNs and the CNs are allowed to be generic (n,k) linear block codes instead of repetition and SPC codes, respectively. If only the CN set is generalized while all the VN are repetition codes, then the code is called a generalized LDPC (GLDPC) code [2].

In a D-GLDPC code, the codes used as VNs and CNs are called *component codes*. In this work, each component code is assumed to be a linear block code having a minimum distance $d_{\min} \geq 2$. The VNs and the CNs that are not repetition or SPC codes, respectively, are referred to as *generalized nodes*. The corresponding code structure is depicted in Fig. 1. An (n_p,k_p) generalized VN is characterized by n_p connections towards the CN set; moreover, k_p of the N D-GLDPC encoded bits are associated with it, and interpreted as its k_p information bits. A repetition VN is a particular case with $k_p=1$. An (n_l,k_l) generalized CN is characterized by n_l connections towards the VN set, and represents n_l-k_l independent parity-check equations of the D-GLDPC code. An SPC CN is a particular case with $n_l-k_l=1$.

Given a sequence of N bits, the generic (n,k) VN encodes a subset of k such bits to generate n bits, each one associated with a specific bipartite graph edge connected to some CN. The sequence of N bits is a codeword for the D-GLDPC code if and only if the produced bits at each CN are a valid codeword for that CN.

In his pioneering work [2], Tanner introduced GLDPC codes with a uniform CN set composed of identical CNs, and a uniform VN set composed of repetition codes, all with the same length. Such GLDPC codes are also known as Tanner codes. In [2], encoding and decoding algorithms were proposed and lower bounds on the overall GLDPC code minimum distance were developed. These bounds involve the variable component codes length, the check component codes minimum distance, and the bipartite graph girth.

GLDPC codes with uniform CN set have been investigated, for example, in [4]–[8]. In [4], the growth rate of the weight distribution is calculated for Tanner codes with Bose–Chaudhuri–Hocquenghem (BCH) check component codes and length-2 repetition VNs, leading to an asymptotic lower bound on the minimum distance. The same lower bound is developed in [5] assuming Hamming check component codes and again length-2 repetition VNs. Both works extend the approach developed by Gallager in [1, Ch. 2], showing that for large enough N, we can expect good minimum distance properties for randomly generated Tanner codes with uniform VN set and uniform CN set, where the CN set is composed of linear block codes with minimum distance $d_{\min} \geq 3$. More specifically,

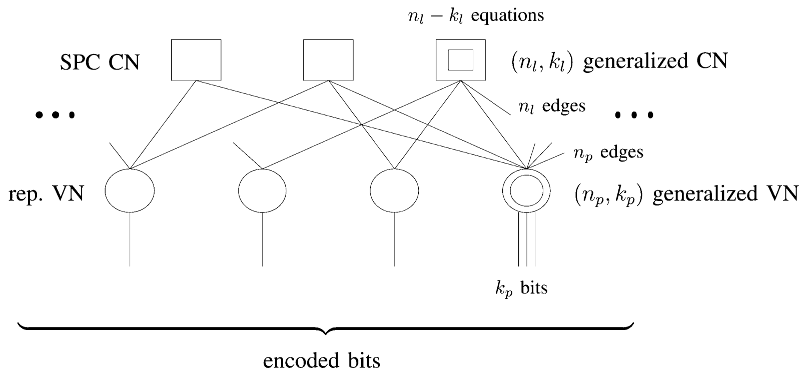

Fig. 1. Structure of a D-GLDPC code.

for large enough N, the minimum distance is a linear function of N.

The growth rate of the number of small-weight codewords for GLDPC codes with uniform CN set and an irregular VN set, composed of repetition codes with different lengths, has been investigated in [6]. It is shown that for large enough N, a minimum distance increasing linearly with N is expected when either the uniform CN set is composed of linear block codes with $d_{\min} \geq 3$, or the minimum length of the repetition VNs is 3. On the other hand, if the CN minimum distance and the minimum length of the repetition VNs are both equal to 2, then for a randomly selected GLDPC code in the ensemble, we expect a minimum distance growing as a linear or sublinear function of N (for large N), depending on the sign of the first-order coefficient in the Taylor series of the growth rate.

In [7], capacity-approaching GLDPC codes with irregular VN set and uniform CN set, where the CNs are represented by Hadamard constraints, were designed for the additive white Gaussian noise (AWGN) channel. This construction is shown to be effective in the low-rate regime. Generalized LDPC codes with uniform VN set composed of length-2 repetition VNs and uniform CN set composed of BCH or Reed–Solomon codes, over both the binary erasure channel (BEC) and the binary symmetric channel (BSC), are considered in [8], assuming bounded distance decoding at the CNs. A threshold analysis over the BEC based on density evolution is developed, together with finite-length analysis based on the concept of generalized stopping set. Furthermore, the analysis over the BEC is exploited to obtain tight bounds on the code performance over the BSC.

Instances of GLDPC codes characterized by a hybrid CN set, composed of a mixture of linear block codes of different types, are investigated in [9]–[11]. The family of GLDPC codes studied in [9] (and there called *hybrid codes*) is characterized by a uniform VN set with length-2 repetition VNs and a hybrid CN set composed of a mixture of Hamming and SPC CNs. According to this construction, each VN is connected to one Hamming CN and to one SPC CN. The effectiveness of this code construction over the Gilbert–Elliott channel is demonstrated. Moreover, the growth rate of the number of small-weight codewords is investigated, showing that the overall minimum distance is expected to grow linearly with the codeword length N.

Generalized LDPC codes for the AWGN channel are designed in [10]. Their CN set is composed of a mixture of SPC and Hamming codes, and a quasi-cyclic structure guaranteeing efficient encoding is used. The work [11] extends [6] in that it evaluates the growth rate of the number of small-weight codewords for GLDPC codes with an irregular VN set and a hybrid CN set. It is shown that a GLDPC code minimum distance growing linearly with N is expected when either the *smallest* CN minimum distance is 3 or the minimum length of the repetition VN is 3. Moreover, if the minimum repetition VN length and the smallest CN minimum distance are both 2, then the expected behavior of the GLDPC code minimum distance (i.e., linear or sublinear with N) only depends on those check and variable component codes in the Tanner graph with $d_{\min} = 2$.

The class of D-GLDPC codes was introduced in [3], where the iterative decoding algorithm for such codes over the AWGN channel is described (for a description of the iterative decoder over the BEC we refer to [12]). Further works concerning D-GLDPC codes are [13]–[15]. In [13], extrinsic information transfer (EXIT) chart analysis [16] for D-GLDPC codes over the AWGN channel is introduced. In [14], a technique based on EXIT chart for threshold analysis of D-GLDPC codes (and GLDPC codes as a particular case) over the BEC is proposed, assuming that each check and variable component code is a random linear block code with minimum distance at least 2. In [15], EXIT chart analysis for the AWGN channel is exploited to construct finite-length D-GLDPC codes whose performance is compared with the performance of LDPC and GLDPC counterparts over the AWGN channel, under a common set of constraints. The simulation results show how the D-GLDPC code is capable of outperforming the LDPC and GLDPC counterparts in both the waterfall and error floor regions.

This paper investigates the stability bound of D-GLDPC codes transmitted over the BEC. The stability bound is an upper bound on the iterative decoding threshold, well known in the context of LDPC codes [17]–[19]. Within the framework of LDPC codes, this bound is of great importance as the reciprocal of the parameter upper bounding the threshold plays a key role in terms of expected minimum distance behavior, encoding complexity, and capability of the ensemble to approach the Shannon capacity [20]. We show that for D-GLDPC codes only the variable and check component codes with minimum distance $d_{\min} = 2$, including length-2 repetition VNs and SPC CNs, contribute to the stability bound and that the bound depends on the generator matrix chosen to represent the VNs with minimum distance 2. Moreover, we show that for D-GLDPC codes achieving the bound with equality, the asymptotic threshold over the BEC can be expressed by a simple, although in general not closed-form, formula. D-GLDPC exploiting SPC VNs are also considered, with the purpose of highlighting their potential capability to overcome common problems affecting GLDPC codes, in particular rate reduction.

The paper is organized as follows. Some preliminary material is presented in Section II. More precisely, some definitions and notations are first introduced. Then the stability bound of LDPC codes over the BEC is reviewed. The concepts of EXIT functions of D-GLDPC codes EXIT charts, and of linear block codes information functions and split information functions are also introduced in this section. Furthermore, the expressions of the EXIT functions for the VN set and CN set of D-GLDPC codes are presented and the approach to obtain the stability bound over the BEC for D-GLDPC codes is described. The stability bound of D-GLDPC codes over the BEC is presented and discussed in Section III, while in Section IV, a particular case is studied, namely, D-GLDPC codes with SPC variable component codes. Concluding remarks are given in Section V.

II. Preliminaries

A. *Definitions and Notations*

We denote by $\mathcal{B}_V$ and $\mathcal{B}_C$ the sets of VN and CN types, respectively. Each VN type and each CN type is identified with an index. We denote by $\mathcal{I}_V$ and $\mathcal{I}_C$ the sets of indices for the VN and CN types, respectively. We let $\mathcal{R}_V$ be the subset of $\mathcal{I}_V$ corresponding to the repetition VNs and $\mathcal{G}_V$ be the subset of $\mathcal{I}_V$ corresponding to the generalized VNs, such that $\mathcal{R}_V \cup \mathcal{G}_V = \mathcal{I}_V$. Analogously, we let $\mathcal{S}_C$ be the subset of $\mathcal{I}_C$ corresponding to the SPC CNs and $\mathcal{G}_C$ be the subset of $\mathcal{I}_C$ corresponding to the generalized CNs, such that $\mathcal{S}_C \cup \mathcal{G}_C = \mathcal{I}_C$. We also denote by $\mathcal{I}_{V,2}$ and $\mathcal{I}_{C,2}$ the subsets of $\mathcal{I}_V$ and $\mathcal{I}_C$ corresponding to the VN and CN types with minimum distance 2, respectively. Similarly, we denote by $\mathcal{G}_{V,2}$ and $\mathcal{G}_{C,2}$ the subsets $\mathcal{G}_V$ and $\mathcal{G}_C$ corresponding to the generalized VN and CN types with minimum distance 2, respectively.

For each $i \in \mathcal{I}_V$ and for each $i \in \mathcal{I}_C$, we let n_i and k_i be the length and the dimension of each type-i component code, and $r_i = k_i/n_i$ be its code rate. For each $i \in \mathcal{I}_C$, we denote by $A_w^{(i)}$ the number of Hamming weight w codewords of a type-i CN. For each $i \in \mathcal{I}_V$, we denote by $B_{w,u}^{(i)}$ the number of Hamming weight w codewords generated by Hamming weight u information words of a type-i VN. We define the VN set and CN set degree distributions from an edge perspective as

$$\lambda(x) \triangleq \sum_{i \in I_V} \lambda_i x^{n_i - 1} \tag{1}$$

and

$$\rho(x) \triangleq \sum_{i \in I_C} \rho_i x^{n_i - 1} \tag{2}$$

respectively, where λ_i and ρ_i are the fractions of edges connected to the VNs and CNs of type i, respectively. We use the convention that for the repetition VN types in the right-hand side (RHS) of (1), any index $i \in \mathcal{R}_V$ is equal to the length of the corresponding repetition VNs. Similarly, we assume that for the SPC CN types in the RHS of (2), any index $i \in \mathcal{S}_C$ is equal to the length of the corresponding SPC CNs. This assumption allows to write $\lambda(x) = \lambda_{\rm rep}(x) + \sum_{i \in \mathcal{G}_V} \lambda_i x^{n_i - 1}$ with

$$\lambda_{\rm rep}(x) \triangleq \sum_{i \in \mathcal{R}_V} \lambda_i \, x^{i-1}. \tag{3}$$

Similarly, we write $\rho(x) = \rho_{\rm SPC}(x) + \sum_{i \in \mathcal{G}_C} \rho_i x^{n_i - 1}$ with

$$\rho_{\rm SPC}(x) \triangleq \sum_{i \in \mathcal{S}_C} \rho_i \, x^{i-1}. \tag{4}$$

Letting N_V and N_C be the number of VNs and CNs, respectively, and letting $\int \lambda = \int_0^1 \lambda(x) \, \mathrm{d}\, x$ and $\int \rho = \int_0^1 \rho(x) \, \mathrm{d}\, x$ the number $N_{V,i}$ of type-i VNs and the number $N_{C,i}$ of type-i CNs are given by $N_{V,i} = N_V \frac{\lambda_i}{n_i \int \lambda}$ and $N_{C,i} = N_C \frac{\rho_i}{n_i \int \rho}$, respectively. Moreover, the D-GLDPC codeword length and number of (not necessarily independent) parity-check equations are given by

$$N = \frac{N_V}{\int \lambda} \sum_{i \in \mathcal{I}_V} \frac{\lambda_i k_i}{n_i}$$

and

$$M = \frac{N_C}{\int \rho} \sum_{i \in \mathcal{I}_C} \frac{\rho_i (n_i - k_i)}{n_i}$$

respectively, so that the overall design code rate is

$$R = 1 - \frac{\sum_{i \in \mathcal{I}_C} \rho_i \, (1 - r_i)}{\sum_{i \in \mathcal{I}_V} \lambda_i \, r_i}. \tag{5}$$

B. *Stability Bound of LDPC Codes Over the BEC*

For LDPC code ensembles an important role is played by a theorem known as the *stability condition*, representing a necessary condition for successful decoding. The stability condition can be formulated for LDPC ensembles transmitted over a generic memoryless binary-input output-symmetric (MBIOS) channel, and its most important consequence is the possibility to bound the asymptotic iterative decoding threshold. If the communication channel is a BEC with erasure probability q, we have the following result [17]–[19].

Theorem 1 (Stability Bound of LDPC Codes Over the BEC): The asymptotic iterative decoding threshold q^* of an LDPC ensemble over the BEC fulfills

$$q^* \leq [\lambda'(0) \, \rho'(1)]^{-1}. \tag{6}$$

The inequality (6) is referred to as *stability bound* in this paper. (It is sometimes referred to as the stability condition. For more details we refer to [19, Theorem 3.66] and the related discussion.) In (6), $\lambda'(0) = \lambda_2$ is the fraction of edges connected to the length-2 repetition VNs, while $\rho'(1)$ is the derivative (computed at $x = 1$) of the LDPC CN degree distribution $\rho(x) = \sum_{j \geq 2} \rho_j x^{j-1}$, where ρ_j is the fraction of edges connected to SPC CNs of length j. The bound (6) was first developed from density evolution.

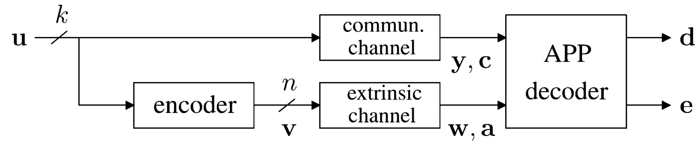

Fig. 2. Block diagram that can be used as a model for either a VN or a CN in the context of D-GLDPC code belief-propagation decoding.

C. EXIT Functions and EXIT Charts for D-GLDPC Codes

In this subsection, the threshold analysis of D-GLDPC codes over the BEC through EXIT chart is introduced. We use here upper-case letters to refer to random variables and lower-case letters for their realizations.

The EXIT chart approach for D-GLDPC codes can be conveniently introduced looking at the communication model depicted in Fig. 2, which is a particular case of [21, Fig. 3]. A k-bit word $\boldsymbol{u} = [u_0, \ldots, u_{k-1}]$ (where the U_i's are independent and identically distributed (i.i.d.) Bernoulli random variables with equiprobable values) is transmitted over a *communication channel* resulting in the word $\boldsymbol{y} = [y_0, \ldots, y_{k-1}]$ or, equivalently, in the vector of k log-likelihood ratios $\boldsymbol{c} = [c_0, \ldots, c_{k-1}]$, where

$$c_i = \log \frac{P(y_i | U_i = 1)}{P(y_i | U_i = 0)}.$$

In parallel, the word $\boldsymbol{u}$ is encoded by an (n,k) linear block encoder with no idle bits, producing the n-bit codeword $\boldsymbol{v} = [v_0, \ldots, v_{n-1}]$ (thus, each V_i is a Bernoulli random variable with equiprobable values). This codeword is then transmitted over an *extrinsic channel* resulting in the received word $\boldsymbol{w} = [w_0, \ldots, w_{n-1}]$ or, equivalently, in the vector of n log-likelihood ratios $\boldsymbol{a} = [a_0, \ldots, a_{n-1}]$, where

$$a_i = \log \frac{P(w_i | V_i = 1)}{P(w_i | V_i = 0)}.$$

An *a posteriori probability* (APP) decoder processes the output of both the communication and the extrinsic channels to produce the n *a posteriori* log-likelihood ratios $\boldsymbol{d} = [d_0, \ldots, d_{n-1}]$ and the n *extrinsic* log-likelihood ratios $\boldsymbol{e} = [e_0, \ldots, e_{n-1}]$. We have

$$d_i = \log \frac{\Pr(V_i = 1 | \boldsymbol{y}, \boldsymbol{w})}{\Pr(V_i = 0 | \boldsymbol{y}, \boldsymbol{w})}$$

and

$$e_i = \log \frac{\Pr(V_i = 1 | \boldsymbol{y}, \boldsymbol{w}_{[i]})}{\Pr(V_i = 0 | \boldsymbol{y}, \boldsymbol{w}_{[i]})}$$

where the notation $\boldsymbol{r}_{[i]}$ is used to denote a vector $\boldsymbol{r}$ except its ith element.

Note that this communication model can be effectively applied to a VN (either repetition or generalized) in the context of D-GLDPC codes iterative decoding. Here, the communication channel of Fig. 2 is the communication channel over which the whole D-GLDPC code is transmitted. In fact, as explained in Section I, any VN interprets its associated D-GLDPC encoded bits as its local information bits. The n encoded bits of an (n,k) VN are checked by the CN set. At each iteration of a log-domain belief-propagation decoder, the VN receives n messages, one along each edge, and interprets the ith message as the *a priori* log-likelihood ratio for its ith encoded bit, namely, a_i. Therefore, the extrinsic channel of Fig. 2 models the channel over which the messages coming from the CN set are received by the VN during each decoding iteration. Assuming that the communication channel of Fig. 2 is the useless channel, i.e., removing the upper branch in the block diagram, Fig. 2 can be also used as a model for a CN, either SPC or generalized. In fact, no communication channel is present in this case. From a CN perspective, the extrinsic channel of Fig. 2 models the channel over which the messages coming from the VN set are received by the CN during each decoding iteration.

For either a VN or a CN, the average *a priori* information is defined as the average mutual information between each encoded bit and the corresponding *a priori* log-likelihood ratio incoming from the extrinsic channel at the generic iteration of the iterative decoding process, namely

$$I_A = \frac{1}{n} \sum_{i=1}^{n} I(V_i; A_i). \tag{7}$$

Similarly, for either a VN or a CN, the average extrinsic information is defined as the average mutual information between each encoded bit and the corresponding outcoming extrinsic log-likelihood ratio at the generic iteration of the iterative decoding process, namely

$$I_E = \frac{1}{n} \sum_{i=1}^{n} I(V_i; E_i). \tag{8}$$

Exploiting [21, Proposition 1], for a VN we have $I(V_i; E_i) = I(V_i; Y_i, \boldsymbol{A}_{[i]})$ for $i = 1, \ldots, n$, which allows I_E to be expressed as a function of I_A and of the communication channel parameters. Similarly, for a CN we have $I(V_i; E_i) = I(V_i; \boldsymbol{A}_{[i]})$ for $i = 1, \ldots, n$, which allows I_E to be expressed as a function of I_A. Assume the communication channel is parameterized by a real-valued parameter α, such that if $\alpha' < \alpha''$ then the channel with parameter α'' is a degraded version of the channel with parameter α'. We have the following definitions.

Definition 1: The *EXIT function* of a VN expresses the average extrinsic information (8) of the VN as a function of its average *a priori* information (7) and of the communications channel parameters, i.e.,

$$I_E = I_E(I_A, \alpha). \tag{9}$$

Definition 2: The *EXIT function* of a CN expresses the average extrinsic information (8) of the CN as a function of its average *a priori* information (7), i.e.,

$$I_E = I_E(I_A). \tag{10}$$

All the VNs of the same type have the same EXIT function. Considering this fact and regarding the overall VN set as a unique VN composed of a mixture of codes, it is readily shown that the EXIT function of the overall VN set is given by the weighted summation of the $|\mathcal{I}_V|$ EXIT functions associated with the VN types, the weights being the coefficients of $\lambda(x)$ defined in (1). Taking the same approach, the EXIT function of the overall CN set is shown to be given by the weighted summation of the $|\mathcal{I}_C|$ EXIT functions associated with the CN types, the weights being the coefficients of $\rho(x)$ defined in (2) (an example is provided in [21, Section III-C]).

Definition 3: The *EXIT chart* of a D-GLDPC code is a plot of the VN set EXIT function and of the *inverse* CN set EXIT function as functions of I_A.

At each iteration of the iterative decoder, the average extrinsic information coming out from the VN set is equal to the CN set *a priori* average information, and *vice versa*. In the limit where the codeword length tends to infinity, the iterative decoding process can be graphically represented as zig-zag pattern (sometimes referred to as the "decoding trajectory") between the VN set EXIT function and the inverse CN set EXIT function. Let us denote by $I_{E,V}(I_A,\alpha)$ and by $I_{E,C}(I_A)$ the VN set EXIT function and the CN set EXIT function, respectively. A necessary and sufficient condition for a successful decoding taking place over a channel with parameter α is the presence of a *decoding tunnel* in the EXIT chart, meaning that $I_{E,V}(I_A,\alpha) > I_{E,C}^{-1}(I_A)\ \forall I_A \in [0,1)$, where $I_{E,C}^{-1}(I_A)$ denotes the inverse CN set EXIT function. The value $\alpha = \alpha^*$ corresponding to the first occurrence of a tangency point between $I_{E,V}(I_A,\alpha)$ and $I_{E,C}^{-1}(I_A)$ is the *iterative decoding threshold* as evaluated by EXIT chart analysis.

D. Information Functions and Split Information Functions

If the communication channel is a BEC, then the EXIT function of a VN is closely related to code parameters known as (un-normalized) *split information functions* [21], while the EXIT function of a CN is closely related to code parameters known as (un-normalized) *information functions* [22]. These parameters are defined next. The given definitions are equivalent to those provided in [21, p. 2663] and [22, p. 549], respectively.

Definition 4: Let $\boldsymbol{G}$ be a generator matrix for an (n,k) linear block code $\mathcal{C}$ and $\boldsymbol{I}_k$ be the $(k \times k)$ identity matrix. The (g,h)th un-normalized split information function of $\mathcal{C}$ under the representation $\boldsymbol{G}$, denoted by $\tilde{e}_{g,h}$, is defined as the summation of the ranks over all the possible submatrices obtained selecting g columns (with $0 \leq g \leq n$) out of $\boldsymbol{G}$ and h columns (with $0 \leq h \leq k$) out of $\boldsymbol{I}_k$.

Definition 5: Let $\boldsymbol{G}$ be a generator matrix for an (n,k) linear block code $\mathcal{C}$. The gth un-normalized information function of $\mathcal{C}$, denoted by $\tilde{e}_g$, is defined as the summation of the ranks over all the possible submatrices obtained selecting g columns (with $0 \leq g \leq n$) out of $\boldsymbol{G}$.

The information function of a linear block code is independent of the code representation. This follows from the fact that any generator matrix of a linear block code can be obtained from any other generator matrix through row summations only. These row summations cannot modify the rank of any submatrix composed of a subset of columns. On the other hand, the split information functions of a linear block code depend on its chosen generator matrix representation, due to the concatenation of $\boldsymbol{G}$ with $\boldsymbol{I}_k$ [14]. In this case, row summations performed on $\boldsymbol{G}$ but not on $\boldsymbol{I}_k$ lead to different ranks for the submatrices composed of a subset of $\boldsymbol{G}$ columns and a subset of $\boldsymbol{I}_k$ columns. Repetition VNs make a trivial exception to the rule, as they admit one code representation only.

E. EXIT Functions for D-GLDPC Codes Over the BEC

Let us assume as transmission channel a BEC with erasure probability q. For a bipartite graph with random connections, the extrinsic channel, that is, the channel over which the messages are exchanged between the VN set and the CN set during the iterative decoding process, is modeled as a second BEC with erasure probability p depending on the decoding iteration [21]. By applying the definition of I_A given in (7), it is readily shown that in this case we have $I_A = 1 - p$ for both a VN and a CN. Hereafter, the EXIT function of a VN is expressed as a function of p and q, and the EXIT function of a CN as a function of p. To obtain the EXIT functions as functions of I_A (and q for the VNs), just replace p with $1 - I_A$.

The EXIT function of an (n,k) VN over the BEC can be readily obtained by applying [21, Theorem 2] to the context of D-GLDPC codes.

Lemma 1: The EXIT function of an (n,k) VN over the BEC, when *maximum a posteriori* (MAP) erasure correction is performed at the VN, is given by

$$I_E(p,q) = 1 - \frac{1}{n}\sum_{t=0}^{n-1}\sum_{z=0}^{k} a_{t,z}\, p^t(1-p)^{n-t-1}\, q^z\,(1-q)^{k-z} \tag{11}$$

where

$$a_{t,z} = (n-t)\,\tilde{e}_{n-t,k-z} - (t+1)\,\tilde{e}_{n-t-1,k-z}. \tag{12}$$

It is readily shown that (11) leads to

$$I_E(p,q) = 1 - q\,p^{n-1} \tag{13}$$

for an $(n,1)$ repetition VN [21, Example 4].

The EXIT function of an (n,k) CN over the BEC can be obtained by letting $q \to 1$ in (11) (absence of communication channel [21, Example 15]).

Lemma 2: The EXIT function of an (n,k) CN over the BEC, when MAP decoding is performed at the CN, is given by

$$I_E(p) = 1 - \frac{1}{n}\sum_{t=0}^{n-1} a_t\, p^t(1-p)^{n-t-1} \tag{14}$$

where

$$a_t = (n-t)\,\tilde{e}_{n-t} - (t+1)\tilde{e}_{n-t-1}.$$

For an $(n,n-1)$ SPC CN, (14) yields [21, Example 5]

$$I_E(p) = (1-p)^{n-1}. \tag{15}$$

Under the hypothesis of a random bipartite graph, VN set and CN set EXIT functions can be expressed as weighted summations of the VN and CN type EXIT functions, namely

$$I_{E,V}(p,q) = \sum_{i \in \mathcal{I}_V} \lambda_i \, I_E^{(i)}(p,q) \tag{16}$$

and

$$I_{E,C}(p) = \sum_{i \in \mathcal{I}_C} \rho_i \, I_E^{(i)}(p) \tag{17}$$

where $I_E^{(i)}(p,q)$ and $I_E^{(i)}(p)$ are the EXIT functions for the ith VN type and for the ith CN type, respectively, and λ_i and ρ_i are the fractions of edges towards the VNs of type i and the CNs of type i, respectively.

For the sake of clarity, it is useful to isolate the contribution of the repetition component codes in (16) and the contribution of the SPC component codes in (17). We have

$$\begin{aligned} I_{E,V}(p,q) &= \sum_{i \in \mathcal{R}_V} \lambda_i \, (1 - q\, p^{\, i-1}) + \sum_{i \in \mathcal{G}_V} \lambda_i \, I_E^{(i)}(p,q) \\ &= \sum_{i \in \mathcal{R}_V} \lambda_i - q\, \lambda_{\mathrm{rep}}(p) + \sum_{i \in \mathcal{G}_V} \lambda_i \, I_E^{(i)}(p,q) \end{aligned} \tag{18}$$

$$\begin{aligned} I_{E,C}(p) &= \sum_{i \in \mathcal{S}_C} \rho_i \, (1-p)^{i-1} + \sum_{i \in \mathcal{G}_C} \rho_i \, I_E^{(i)}(p) \\ &= \rho_{\mathrm{SPC}}(1-p) + \sum_{i \in \mathcal{G}_C} \rho_i \, I_E^{(i)}(p). \end{aligned} \tag{19}$$

To obtain (18) we make use of (13) and (3). Similarly, to obtain (19) we make use of (15) and (4).

We recall that the split information function of a generalized VN, and therefore its EXIT function (11), depends on the chosen generator matrix representation. Hence, the performance of the overall D-GLDPC code depends on the code representation of its variable component codes. For this reason, two generalized VNs associated with the same code, but with different generator matrices (i.e., different mappings between information words and codewords) must be regarded as VNs of different types in (16). On the other hand, since the information function of each CN is independent of the specific code representation, different check component code representations are associated with the same EXIT function for the generalized CNs. The performance of a GLDPC or D-GLDPC code is then independent of the specific representation of its generalized check component codes.

F. Stability Bound Over the BEC From Exit Chart Analysis and Derivative Matching Condition

Next we propose a simple graphical interpretation of the stability bound (6) of LDPC codes over the BEC, using an EXIT chart.

Proposition 1 (Second Form of the Stability Bound of LDPC Codes Over the BEC): The stability bound of LDPC codes over the BEC is equivalent to the following condition for $q = q^*$

$$\left. \frac{\partial I_{E,V}(p,q^*)}{\partial p} \right|_{p=0} \geq \left. \frac{\mathrm{d} I_{E,C}^{-1}(p)}{\mathrm{d}p} \right|_{p=0}. \tag{20}$$

Proof: For LDPC codes, (6) implies (20) and *vice versa*. □

The approach followed in this paper to obtain a generalization of the stability bound for D-GLDPC codes over the BEC consists of developing (20) using the general expressions of the VN set and CN set EXIT functions (16) and (17) together with the expressions of the EXIT functions for a specific VN or a specific CN reviewed in (11) and (14), respectively. When extending the approach to D-GLDPC codes, it is necessary to take care of the value assumed by the VN set and inverse CN set EXIT functions at $p = 0$ (equivalently, at $I_A = 1$). For binary LDPC code ensembles characterized by VNs and CNs with degree at least 2, for $p = 0$ we always have $I_{E,V}(p,q) = 1 \forall q$ and $I_{E,C}^{-1}(p) = 1$. As discussed in [14], this holds also for binary D-GLDPC codes with all variable and check component codes having a minimum distance $d_{\min} \geq 2$, which is an assumption of this paper.

There exist LDPC degree distributions achieving the stability bound (6) with equality, so that their threshold over the BEC assumes the simple closed form $q^* = [\lambda'(0)\rho'(1)]^{-1}$ (for LDPC codes this is referred to as the *flatness condition* [23]). For such LDPC distributions, the first occurrence of a tangency point between the VN set EXIT function $I_{E,V}(p,q)$ and the inverse CN set EXIT function $I_{E,C}^{-1}(p)$ appears at $p = 0$ (equivalently, $I_A = 1$), i.e.,

$$\left. \frac{\partial I_{E,V}(p,\, q^*)}{\partial p} \right|_{p=0} = \left. \frac{\mathrm{d} I_{E,C}^{-1}(p)}{\mathrm{d}p} \right|_{p=0}. \tag{21}$$

Achieving (20) with equality is also possible for D-GLDPC code ensembles. This condition is referred to as the *derivative matching condition* in the sequel.

III. Stability Bound of D-GLDPC Codes Over the BEC

A. Main Result

In this section, the stability bound for D-GLDPC code ensembles transmitted over the BEC is claimed and discussed.

Theorem 2 (Stability Bound of D-GLDPC Codes Over the BEC): Let us consider a D-GLDPC code ensemble with a nonzero fraction of VNs with minimum distance 2 and a nonzero fraction of CNs with minimum distance 2. Let $\mathcal{I}_{V,2}$ and $\mathcal{I}_{C,2}$ be the set of indices for the minimum distance 2 VN and CN types, respectively. Let $A_2^{(i)}$ and $B_{2,u}^{(i)}$ denote the number of Hamming weight–2 codewords for a CN of type $i \in \mathcal{I}_{C,2}$ and the number of Hamming weight–2 codewords generated by a Hamming weight u information word for a VN of type $i \in \mathcal{I}_{V,2}$, respectively. For a type-i VN or CN let n_i be the VN or CN length. For a type-i VN, let k_i be the VN dimension. Furthermore, let λ_i and ρ_i denote the fraction of edges connected to the type-i VNs and to the type-i CNs, respectively. Define

$$P(x) \triangleq \sum_{i \in \mathcal{I}_{V,2}} \lambda_i \, P_i(x) \tag{22}$$

where, for each $i \in \mathcal{I}_{V,2}$

$$P_i(x) = \sum_{u=1}^{k_i} \frac{2\, B_{2,u}^{(i)}}{n_i} \, x^u. \tag{23}$$

Furthermore, define

$$C \triangleq \sum_{i \in \mathcal{I}_{C,2}} \rho_i \, C_i \tag{24}$$

where, for each $i \in \mathcal{I}_{C,2}$

$$C_i = \frac{2\, A_2^{(i)}}{n_i}. \tag{25}$$

Then, the asymptotic iterative decoding threshold q^* of the D-GLDPC code ensemble over the BEC, assuming MAP erasure correction at each component code, fulfills

$$q^* \le P^{-1}\left(\frac{1}{C}\right) \tag{26}$$

where $P^{-1}(\,\cdot\,)$ is the inverse function of $P(\,\cdot\,)$.

Proof: See Appendix I. □

As pointed out in Section I, GLDPC codes represent a particular case of D-GLDPC codes in that all the VNs are repetition codes. For this particular case, Theorem 2 is specialized as follows.

Corollary 1 (Stability Bound of GLDPC Codes Over the BEC): The asymptotic iterative decoding threshold q^* of a GLDPC code ensemble over the BEC, assuming MAP erasure correction at each component code, fulfills

$$q^* \le [\lambda'(0)\, C]^{-1} \tag{27}$$

where C is defined in (24).

Proof: For a GLDPC ensemble, the only $(d_{\min} = 2)$ VNs are the length-2 repetition VN, for which $P_i(x) = x$. Moreover, we have $\lambda(x) = \lambda_{\text{rep}}(x)$, where $\lambda_{\text{rep}}(x)$ is defined in (3) with the assumption that each index $i \in \mathcal{R}_V$ is equal to the length of the corresponding repetition VNs. We then obtain $P(x) = \lambda'(0)\, x$ and therefore $P^{-1}(x) = x/\lambda'(0)$ which, substituted in (26), leads to (27). □

Furthermore, for LDPC codes all the CNs are SPC codes, so that $\rho(x) = \rho_{\text{SPC}}(x)$ with $\rho_{\text{SPC}}(x)$ defined in (4), with the assumption that each index $i \in \mathcal{S}_C$ is equal to the length of the corresponding SPC CN. For a length-i SPC CN, i.e., for a traditional degree-i LDPC CN, C_i defined in (25) is given by $C_i = i - 1$. Hence, for LDPC ensembles we then obtain $C = \rho'(1)$ and the stability bound (6) follows. Note that for both GLDPC and LDPC ensembles the stability bound does not depend on the VN representations as for the length-2 repetition VNs, only one representation is possible.

B. Discussion

1) Dependence of the Bound on the VNs and CNs With Minimum Distance 2: The stability bound of LDPC codes over the BEC depends solely on the variable and check component codes with minimum distance 2, i.e., on the length-2 repetition VNs and on the SPC CNs (of any length). This property is preserved when the bound is extended to the more general class of D-GLDPC codes. In fact, the polynomial $P(x)$ defined in (22) only depends on the $(d_{\min} = 2)$ VN types, while the constant C defined in (24) only depends on the $(d_{\min} = 2)$ CN types.

Note that these VNs and CNs are the weakest nodes in the Tanner graph from an erasure recovery capability viewpoint. Furthermore, note that the bound (26) is obtained by developing (20) within the framework of D-GLDPC codes, and then by analyzing the EXIT chart for values of p close to zero. Since p represents the erasure probability of the extrinsic channel, we can interpret (26) as an upper bound on the iterative decoding threshold obtained assuming the residual fraction of erasure messages is close to zero. This fact suggests that, when the iterative decoder is about to (asymptotically) succeed, the residual erasure messages are associated with the weakest nodes, i.e., the $(d_{\min} = 2)$ VNs and CNs.

2) Dependence of the Bound on the Generalized VN Representations: It follows from (22) that $P(x)$, and then the bound (26), depends on the input–output weight spectrum of the variable component codes with minimum distance 2. More precisely, $P(x)$ depends on the coefficients $B_{2,u}^{(i)}$ with $i \in \mathcal{I}_{V,2}$, and therefore it depends on the generator matrix chosen for each $i \in \mathcal{I}_{V,2}$ (through the single enumerator $B_{2,u}^{(i)}$). On the other hand, the parameter C only depends on the distance spectrum of the check component codes with minimum distance 2 so that it is independent of the specific generator matrix chosen for each $i \in \mathcal{I}_{C,2}$. This is coherent with the fact that the EXIT function of a generalized VN depends on its representation while the EXIT function of a generalized CN does not depend on its representation, as pointed out in Section II-E.

We observe that for each $i \in \mathcal{I}_{V,2}$, at $x = 1$ the polynomial $P_i(x)$ specified in (23) does not depend on the representation chosen for the type-i VNs. In fact, at $x = 1$ we have

$$P_i(1) = \frac{2\sum_{u=1}^{k_i} B_{2,u}^{(i)}}{n_i} = \frac{2\, B_2^{(i)}}{n_i}$$

where $B_2^{(i)}$, the number of Hamming weight–2 codewords of a type-i VN, is independent of its representation.

We note that the bound (26) is useful when $P^{-1}(1/C) < 1 - R$ (i.e., the BEC capacity) and useless when $P^{-1}(1/C) > 1 - R$, where R is the design rate given by (5) that is independent of the VNs representation. As a particular case, if the smallest minimum distance among either the variable component code types or the check component code types is at least 3, (26) becomes useless.

For a given value of C and a given design rate R, consider two instances of $P(x)$, namely, $P_{(1)}(x)$ and $P_{(2)}(x)$, corresponding to different representations of the generalized VNs. Let us assume $P_{(1)}(x) > P_{(2)}(x)$ for all $0 < x < 1$. Furthermore, let us assume that (26) is useful for both $P_{(1)}(x)$ and $P_{(2)}(x)$, which implies that $P_{(1)}(1 - R) > 1/C$ and $P_{(2)}(1 - R) > 1/C$ (this is depicted in Fig. 3). By (26) we must have $P(q^*) \le 1/C$: Therefore, $P_{(2)}(x)$ allows a potentially larger threshold than $P_{(1)}(x)$. In general, for a given set of CN types, a given distribution $\rho(x)$, and a given design rate R, assuming the condition $P(1 - R) > 1/C$ is fulfilled (useful bound), from a stability bound viewpoint polynomials $P(x)$ as small as possible in the interval $(0, 1)$ are preferable. In terms of generalized VNs representation, provided the overall $P(x)$ fulfills $P(1 - R) > 1/C$, representations of $(d_{\min} = 2)$-VNs associated with a smaller

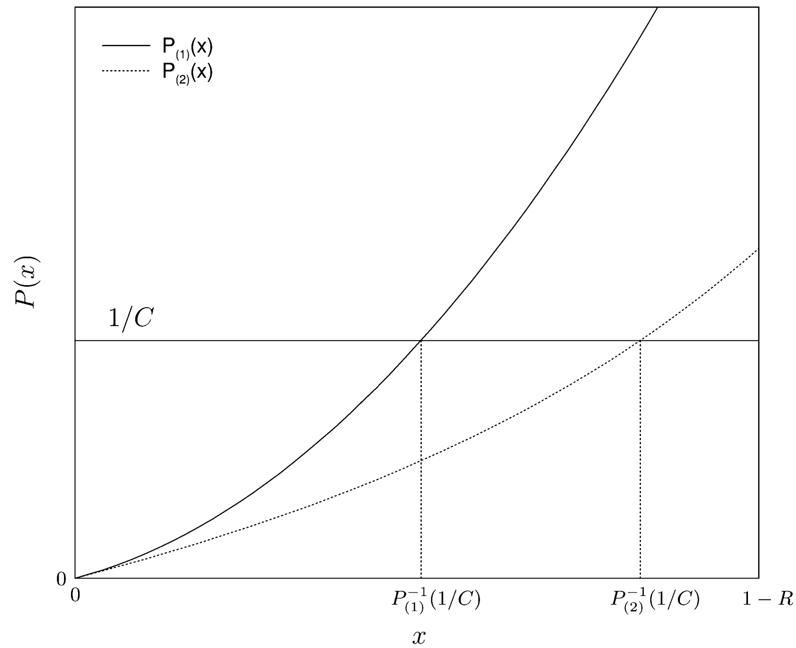

Fig. 3. Comparison between two different polynomials $P(x)$, denoted by $P_{(1)}(x)$ (solid curve) and $P_{(2)}(x)$ (dotted curve). For the same value of C such that $P_{(1)}(1-R) > 1/C$ and $P_{(2)}(1-R) > 1/C$, if $P_{(1)}(x) > P_{(2)}(x)$ for $0 < x < 1$, then $P_2(x)$ allows an upper bound on the threshold q^* closer to $1-R$.

$P_i(x)$ for $0 < x < 1$ are preferable from a stability bound viewpoint. In general, a small $P_i(x)$ for $0 < x < 1$ is obtained for generator matrices for which Hamming weight–2 codewords are obtained from high Hamming weight information words.

3) Closed-Form Bounds: The polynomial $P(x)$ may be of any order so that it is not possible to obtain a general closed-form expression for the RHS of (26). It is however possible to develop weaker bounds admitting a closed form by noting that, since for any $i \in \mathcal{I}_{V,2}$ each term of $P_i(x)$ is positive, we may remove any term from $P_i(x)$ and obtain a lower bound on $P_i(x)$. Next, for any $i \in \mathcal{I}_{V,2}$, we remove all terms from $P_i(x)$ except the lowest degree terms.

For each $i \in \mathcal{I}_{V,2}$, let us define

$$P_i^{[j]}(x) = \sum_{u=1}^{j} \frac{2B_{2,u}^{(i)}}{n_i} x^u$$

and

$$P^{[j]}(x) = \sum_{i \in \mathcal{I}_{V,2}} \lambda_i P_i^{[j]}(x).$$

Since all terms of $P(x)$ are positive, we have $P^{[j]}(x) \leq P(x)$ for all $x \geq 0$ and any j. Then for a given D-GLDPC ensemble, we have $P^{[j]}(q^*) \leq P(q^*) \leq 1/C$, so that we obtain the weaker bound

$$q^* \leq P^{-[j]}\left(\frac{1}{C}\right) \tag{28}$$

where $P^{-[j]}(x)$ denotes the inverse function of $P^{[j]}(x)$. For $j = 1$, (28) assumes the form

$$q^* \leq (\beta C)^{-1}$$

where

$$\beta = \sum_{i \in I_{V,2}} \lambda_i \frac{2B_{2,1}^{(i)}}{n_i}.$$

For $j = 2$ it is readily shown that (28) specializes to

$$q^* \leq \frac{1}{2\alpha}\sqrt{\beta^2 + \frac{4\alpha}{C}} - \frac{\beta}{2\alpha}$$

where

$$\alpha = \sum_{i \in \mathcal{I}_{V,2}} \lambda_i \frac{2B_{2,2}^{(i)}}{n_i}.$$

There are also special cases in which the RHS of (26) admits a closed form. For instance, this is possible when the only ($d_{\min} = 2$) variable component codes are length-2 repetition codes and length-n SPC codes in systematic form. Let λ be the fraction of edges connected to the length-2 repetition VNs and μ the fraction of edges connected to the length-n SPC VNs (so $\lambda + \mu$ is the total fraction of edges connected to $d_{\min} = 2$ VNs). Specializing the expression of $P_i(x)$ given in (23) for an SPC VN represented in systematic form (more details are provided in Section IV-B) we have in this case

$$P(x) = \lambda\, x + \mu\, \frac{2(n-1)}{n}\, x \left(\frac{n-2}{2}\, x + 1\right).$$

By solving for positive y the equation $P(y) = x$, we obtain

$$P^{-1}(x) = \frac{-n\,\lambda + 2\,(1-n)\,\mu}{2\,(n-2)\,(n-1)\,\mu} + \frac{\sqrt{[n\,\lambda + 2\,(n-1)\,\mu]^2 + 4\,(n-2)\,(n-1)\,n\,\mu\,x}}{2\,(n-2)\,(n-1)\,\mu}. \tag{29}$$

We plot (29) in Fig. 4, assuming $\lambda + \mu = 0.654$ and SPC VNs of length $n = 15$, for $\mu = 0$ (dashed curve, corresponding to the presence of only length-2 repetition VNs) and for $\mu = 0.1, 0.522, 0.654$ (solid curves). Note that the curve associated

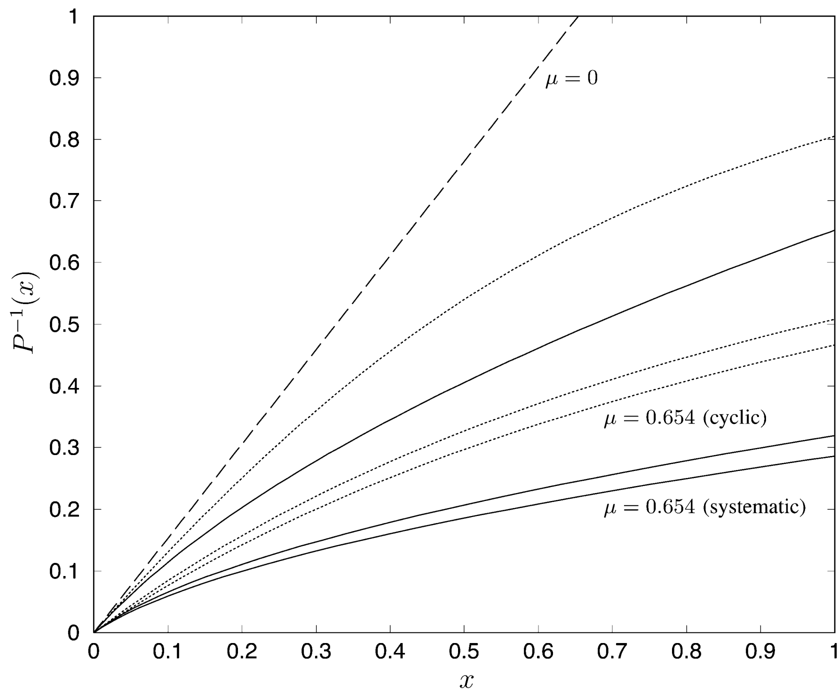

Fig. 4. Plot of $P^{-1}(\cdot)$ for a D-GLDPC ensemble where the only $d_{\min} = 2$ VNs are length-2 repetition VNs and length-15 SPC VNs. The total fraction of edges connected to $d_{\min} = 2$ VNs is $\lambda + \mu = 0.654$, where λ is the fraction of edges connected to the length-2 repetition VNs and μ is the fraction of edges connected to the length-15 SPC VNs. The dashed line corresponds to $\mu = 0$, i.e., to the presence of length-2 repetition VNs only. The solid curves correspond to the systematic representation of the SPC VNs (for $\mu = 0.1, 0.522, 0.654$) while the dotted curves correspond to the cyclic representation of the SPC VNs (for $\mu = 0.1, 0.522, 0.654$). For both representations, $\mu = 0.654$ corresponds to the presence of SPC VNs only.

with $\mu = 0.654$ corresponds to the presence of only length-15 SPC VNs in systematic form.

4) Derivative Matching Condition: For a D-GLDPC ensemble satisfying the derivative matching condition, the iterative decoding threshold over the BEC assumes the form

$$q^* = P^{-1}\left(\frac{1}{C}\right). \tag{30}$$

In this subsection, we highlight how in some cases (30) cannot be achieved within the framework of GLDPC codes while, on the other hand, it is in principle possible to satisfy (30) by properly generalizing the VN set.

We observe that any length-n_i and Hamming weight–2 binary sequence is a codeword for a length-n_i SPC CN. Thus, for any length-n_i CN with minimum distance 2, C_i given in (25) fulfills

$$C_i \le n_i - 1$$

with equality if and only if the CN is a length-n_i SPC code. We also note that any generalized CN with minimum distance larger than 2 does not contribute to the constant C defined in (24). We may formalize this as follows.

Fact 1 (Parameter C Loss): The generalized CNs introduce a loss in terms of the parameter C with respect to an LDPC ensemble characterized by the same CN degrees and $\rho(x)$.

Next, we observe that the overall design rate of a D-GLDPC ensemble, given by (5), is monotonically increasing with respect to any r_i, for both $i \in \mathcal{I}_V$ and $i \in \mathcal{I}_C$, and that a generalized CN of length n_i has a code rate smaller than that of a length-n_i SPC CN. Hence, we have the following.

Fact 2 (Rate Loss): The generalized CNs introduce a loss in terms of the ensemble design rate R with respect to the design rate of an LDPC ensemble characterized by the same CN degrees and $\rho(x)$.[1]

Given an LDPC code ensemble with some C, some R, and achieving (30) for some q^*, the replacement of a fraction of length-n_i SPC CNs with length-n_i generalized CNs leads to lower values of both C and R, which are denoted next by $\overline{C}$ and $\overline{R}$. Note that the reduction of C induces a derivative mismatch at $p = 0$ (equivalently, at $I_A = 1$) between the VN set EXIT function and the inverse CN set EXIT function, so that (30) is no longer fulfilled. In fact, as stated by Lemma 3 in Appendix I, we have $\mathrm{d}I_{E,C}^{-1}(p)/\mathrm{d}p\,|_{p=0} = -1/C$ which, reformulated in terms of $I_A = 1 - p$, can be written as

$$\left.\frac{\mathrm{d}I_{E,C}^{-1}(I_A)}{\mathrm{d}I_A}\right|_{I_A=1} = 1/C$$

showing that a loss in terms of C induces an increase in the derivative of the inverse CN set EXIT function at $I_A = 1$.

Assume we wish to modify $\lambda(x)$, within the framework of GLDPC codes, to increase the design rate from $\overline{R}$ back to R and to have (30) fulfilled again for the same q^*. A necessary (in general, not sufficient) modification to the VN set consists of increasing the fraction of edges connected to length-2 repetition VNs, namely, $\lambda'(0)$. In fact, for GLDPC ensembles (30) assumes the form

$$q^* = [\lambda'(0)C]^{-1} \tag{31}$$

[1]This rate loss was already observed in [8].

so that increasing $\lambda'(0)$ is mandatory to compensate for a loss in terms of C, assuming q^* is constant. Moreover, length-2 repetition VNs are generally the highest rate VNs in GLDPC ensembles of interest, so that they provide the best compensation for the rate loss. Suppose, however, that the value $\overline{C}$ fulfills $\overline{C} < 1/(1-R)$, where R is the design rate to be achieved. From (31) we obtain

$$\lambda'(0) = \frac{1}{q^*\overline{C}} > \frac{1-R}{q^*} > 1$$

as over the BEC we must have $q^* < 1 - R$. Hence, the derivative matching condition cannot be achieved by any GLDPC ensemble with design rate R and $C < 1/(1-R)$.

Suppose now we wish to solve the same problem within the framework of D-GLDPC codes. Let ν be the fraction of edges connected to the length-2 repetition VNs in the initial ensemble, and denote by $\overline{P}(x)$ the polynomial $P(x)$ of the final VN set. In order to achieve (30) for the same q^* and compensate for the rate loss, it is necessary to design the VN set so that

$$\overline{P}^{-1}(1/\overline{C}) = [\nu\, C]^{-1} \tag{32}$$

and so that the final rate is R. Note that, as opposed to the GLDPC case, increasing the fraction of edges connected to the length-2 repetition VNs is not a mandatory way to satisfy (32) within the framework of D-GLDPC codes. To better clarify this point, suppose for instance that for some $i \in \mathcal{G}_{V,2}$ we have $B_{2,1}^{(i)} \geq n_i/2$. It is readily shown that under this condition $P_i(x)$ specified in (23) fulfills $P_i(x) > x$ for all $x > 0$, so that $P_i^{-1}(x) < x$ for all $x > 0$. This simple example illustrates how certain classes of $(d_{\min} = 2)$ generalized VNs have the capability to *flatten* the function $P^{-1}(x)$ with respect to the linear behavior obtained with GLDPC codes. An example of this phenomenon can be observed in Fig. 4, where the $(d_{\min} = 2)$ VNs are a mixture of length-2 repetition VNs and length-15 SPC VNs. In principle, a sublinear $P^{-1}(x)$ allows (32) to be satisfied even in the presence of values of $\overline{C}$ for which $\overline{C} < 1/(1-R)$, i.e., for which the derivative mismatch cannot be compensated with GLDPC codes. Also note that introducing generalized VNs of rate $r_i > 1/2$ is beneficial in terms of compensation for the rate loss $R - \overline{R}$. Under this perspective, $(d_{\min} = 2)$ generalized VNs with code rate larger than $1/2$ are an appealing option to compensate for both the parameter C loss and the rate loss due to the use of generalized CNs. A subclass of such VNs is considered in the next section.

IV. D-GLDPC Codes With SPC Variable Nodes

A. Motivation

GLDPC codes employing generalized CNs such as Hamming or BCH CNs are known to represent a possible solution for obtaining a good compromise between waterfall performance and error floor. Examples of such GLDPC code constructions are described in [4], [5], [7], [9], [10]. In general, increasing the fraction of edges connected to the generalized CNs may be favorable from the viewpoint of the overall code minimum distance and then of the error floor, but presents drawbacks within the framework of GLDPC codes, as highlighted next.

Note that, as the loss in terms of the parameter C claimed in Fact 1 is associated with an increase of the derivative of the inverse CN set EXIT function at $I_A = 1$, it tends to induce an area gap between the EXIT curves for values of I_A close to 1.[2] This phenomenon is even more pronounced when generalized CNs with $d_{\min} \geq 3$ are introduced. In fact, the EXIT function over the BEC of any such CN is characterized by $\mathrm{d}I_E(I_A)/\mathrm{d}I_A|_{I_A=1} = 0$ (see the Proof of Lemma 3 in Appendix II). Then, its inverse EXIT function presents a derivative diverging to $+\infty$ as $I_A \to 1$. The inverse EXIT function of any generalized CNs with $d_{\min} \geq 3$ has a typical shape. An example is presented in Fig. 5, where the inverse EXIT function of a $(31, 21)$ BCH CN is depicted (dot-dashed curve).

As explained in the previous section, the only possible way to reduce the area gap between the EXIT curves introduced by the generalized CNs, while compensating for the rate loss, consists of using sufficiently large fractions of edges connected to length-2 repetition VNs. Indeed, this is the reason for which numerical tools for threshold optimization of GLDPC code ensembles such as differential evolution (DE) [24], [25] typically return distributions with large fractions of edges connected to the length-2 repetition VNs when large fractions of edges connected to the generalized CNs are imposed, especially if these CNs are characterized by $d_{\min} \geq 3$ and if the target code rate is not too small (for instance, a design rate $R = 1/2$ is considered in a numerical example presented in Section IV-D).

The EXIT function of a length-2 repetition VN is given by $I_E(I_A, q) = 1 - q(1 - I_A)$ which is linear with respect to I_A. When a GLDPC ensemble presents a large fraction of edges connected to generalized CNs with $d_{\min} \geq 3$ and a large fraction of edges connected to length-2 repetition VNs, the shape of the overall inverse CN set EXIT function (17), and that of the overall VN set EXIT function (16) are heavily influenced by these CNs and VNs, respectively. The corresponding EXIT curves present quite a different shape and usually do not allow to obtain a satisfying curve matching. Consequently, the threshold values achieved by such GLDPC code ensembles are usually quite poor. Several examples of such poor thresholds can be found in [8, Table II] for GLDPC codes where all the CNs have minimum distance at least 3.

Allowing the use of generalized VNs provides an increased flexibility in the variable component codes selection that may be exploited to overcome these issues. In this context, an interesting class of generalized VNs is represented by $(n, n-1)$ SPC VNs, each one having n edges towards the CN set and associated with $n - 1$ encoded bits. SPC VNs exhibit three appealing features. First, as for a given n and $d_{\min} \geq 2$ they are the VNs with the highest code rate $r = (n-1)/n$, they are able to provide an excellent compensation for the rate loss claimed in Fact 2. Second, they have minimum distance 2. This feature makes them candidates to compensate for the area gap. Finally, they are as easy to decode as repetition codes.

We focus next on D-GLDPC codes employing SPC VNs. Through a numerical case study which uses the DE algorithm to optimize the degree profile of D-GLDPC ensemble under a

[2] See [21] for the implications of the area gap on the threshold. In general, the larger this area gap, the weaker the threshold becomes.

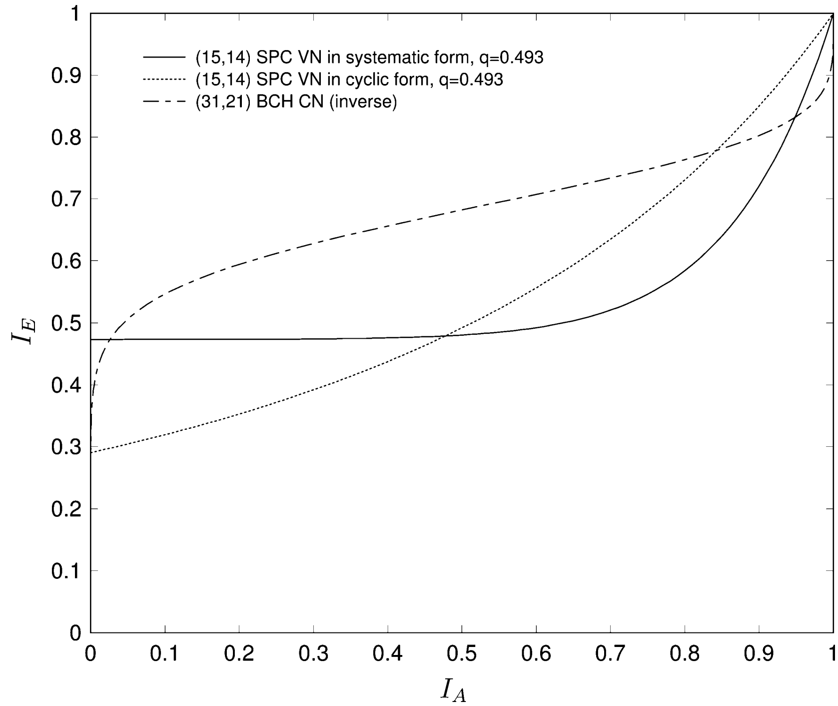

Fig. 5. Inverse EXIT function over the BEC of a $(31, 21)$ BCH CN (dot-dashed curve) and EXIT functions over the BEC of a $(15, 14)$ SPC VN represented in systematic form (solid curve) and in cyclic form (dotted curve) for a channel erasure probability $q = 0.493$.

given set of constraints, the effect of the SPC VNs is investigated from both an EXIT chart viewpoint and from a stability bound viewpoint. We consider two possible representations of the SPC CNs, namely, the systematic and the cyclic representations.

B. SPC Variable Nodes in Systematic Form

Let us suppose that the VNs of type-i are length-n_i SPC codes in systematic form, i.e., represented by the $((n_i - 1) \times n_i)$ generator matrix

$$\boldsymbol{G}_i = \begin{bmatrix} 1 & 0 & \cdots & 0 & 1 \\ 0 & 1 & \cdots & 0 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 1 \end{bmatrix}.$$

Each of these VNs has $\binom{n_i}{2}$ weight-2 codewords. Specifically, there are $n_i - 1$ weight-2 codewords generated by weight-1 information words, and $\binom{n_i-1}{2} = \frac{(n_i-1)(n_i-2)}{2}$ weight-2 codewords generated by weight-2 information words. No other weight-2 codewords are generated by information words of weight larger than 2. Then

$$B_{2,u}^{(i)} = \begin{cases} n_i - 1, & \text{if } u = 1 \\ (n_i - 1)(n_i - 2)/2, & \text{if } u = 2 \\ 0, & \text{if } u = 3, \dots, n_i - 1 \end{cases}$$

so that $P_i(x)$ given in (23) becomes

$$P_i(x) = \frac{2(n_i - 1)}{n_i}\, x \left(1 + \frac{n_i - 2}{2}\, x\right). \tag{33}$$

C. SPC Variable Nodes in Cyclic Form

Let the VNs of type-i be $(n_i, n_i - 1)$ SPC codes in cyclic form, i.e., generated by

$$\boldsymbol{G}_i = \begin{bmatrix} 1 & 1 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 1 \end{bmatrix}.$$

In this case, we obtain an expression of $P_i(x)$ different from (33). In fact, it is readily shown that in an SPC code represented in cyclic form, an information word of weight u generates a weight-2 codeword if and only if all its “1” positions are contiguous. Then, for all $u = 1, \dots, n_i - 1$ we have $B_{2,u}^{(i)} = n_i - u$, from which $P_i(x)$ given in (23) becomes

$$P_i(x) = \sum_{u=1}^{n_i-1} \frac{2\,(n_i - u)}{n_i}\, x^u = \frac{2\,x\,[x^{n_i} - n_i\,(x-1) - 1]}{n_i\,(x-1)^2}. \tag{34}$$

If $n_i = 2$ or $n_i = 3$, then (33) coincides with (34) as expected. Specifically, from both (33) and (34) we obtain $P_i(x) = x$ and $P_i(x) = \frac{2}{3}\,x^2 + \frac{4}{3}\,x$ for $n_i = 2$ and $n_i = 3$, respectively.

For $n_i > 3$, let us denote by $P_i^{(s)}(x)$ and by $P_i^{(c)}(x)$ the polynomial $P_i(x)$ of a length-n_i SPC VN in systematic and cyclic form, respectively. We show next that if $n_i > 3$

$$\begin{aligned} P_i^{(s)}(x) &> P_i^{(c)}(x), && \text{if } 0 < x < 1 \\ P_i^{(s)}(x) &= P_i^{(c)}(x), && \text{if } x = 1 \\ P_i^{(s)}(x) &< P_i^{(c)}(x), && \text{if } x > 1. \end{aligned}$$

TABLE I
OPTIMUM D-GLDPC ENSEMBLES RETURNED BY THE DIFFERENTIAL EVOLUTION ALGORITHM

	D-GLDPC S_1	D-GLDPC C_1	D-GLDPC S_2	D-GLDPC C_2
Variable Nodes				
SPC 15		0.086822		0.521581
rep 2	0.348281	0.096639	1.000000	0.132836
rep 3	0.076273	0.443419		
rep 4	0.408832	0.183604		
rep 14				0.145293
rep 15	0.166614	0.189517		0.200291
Check Nodes				
BCH	0.149520	0.139048	0.700000	0.721799
SPC 5				0.278201
SPC 7	0.106288	0.860952		
SPC 8	0.744192			
SPC 12			0.174190	
SPC 13			0.125810	
q^*	0.491036	0.491875	0.291516	0.478585
$P^{-1}(1/C)$	0.491057	0.491894	0.291902	0.478585

We have

$$\begin{aligned} P_i^{(s)}(x) - P_i^{(c)}(x) &= \frac{2}{n_i}\left[(n_i-1)\,x + \frac{(n_i-1)(n_i-2)}{2}\,x^2\right] \\ &\quad - \frac{2}{n_i}\sum_{u=1}^{n_i-1}(n_i-u)\,x^u \\ &= \frac{2\,x^2}{n_i}\left[\frac{(n_i-2)(n_i-3)}{2} - \sum_{u=3}^{n_i-1}(n_i-u)\,x^{u-2}\right]. \end{aligned} \quad (35)$$

Since

$$\sum_{u=3}^{n_i-1}(n_i-u) = \frac{(n_i-2)(n_i-3)}{2}$$

we have $P_i^{(s)}(1) = P_i^{(c)}(1)$, a result already highlighted in the previous section. For $0 < x < 1$ we have

$$\sum_{u=3}^{n_i-1}(n_i-u)\,x^{u-2} < \frac{(n_i-2)(n_i-3)}{2}$$

which leads to $P_i^{(s)}(x) - P_i^{(c)}(x) > 0$; analogously, for $x > 1$ we have

$$\sum_{u=3}^{n_i-1}(n_i-u)\,x^{u-2} > \frac{(n_i-2)(n_i-3)}{2}$$

which leads to $P_i^{(s)}(x) - P_i^{(c)}(x) < 0$. This result for $0 < x < 1$ indicates that the cyclic representation has a better potential than the systematic representation to provide a higher threshold q^* (see Fig. 3).

D. Distribution Optimization Example

Let $f(\mathcal{B}_V, \mathcal{B}_C, \lambda(x), \rho(x))$ be a function returning the threshold for the D-GLDPC ensemble characterized by VN type set $\mathcal{B}_V$, CN type set $\mathcal{B}_C$, and variable and check node degree distributions $\lambda(x)$ and $\rho(x)$, respectively. We pose the optimization problem as follows.

- *Given* that both communication channel and the extrinsic channel are a BEC
- *Maximize* $f(\mathcal{B}_V, \mathcal{B}_C, \lambda(x), \rho(x))$ with respect to $\lambda(x)$ and $\rho(x)$
- *Subject to* $\mathcal{B}_V$, $\mathcal{B}_C$, R, and $\mathcal{A}$.

Here, R is the design code rate expressed by (5) and $\mathcal{A}$ is a set of possible additional constraints. The ensemble optimization is performed using the DE algorithm.

We consider the optimization of D-GLDPC ensembles with design rate $R = 1/2$. We let the set $\mathcal{B}_V$ include repetition VNs with degree ranging between 2 and 15 and length-15 SPC VNs either in cyclic or systematic form. Furthermore, we let the set $\mathcal{B}_C$ include SPC CNs with degree ranging between 5 and 15 and $(31, 21)$ BCH CNs (having minimum distance 5). First we assume $\mathcal{A}$ is the empty set. Let us consider the two columns of Table I labeled as "D-GLDPC S_1" and "D-GLDPC C_1" which correspond to the optimization of the D-GLDPC ensembles employing the SPC VNs in systematic and in cyclic form, respectively. We observe that in the first case, the optimal distribution returned by DE is a GLDPC distribution, i.e., there is a zero fraction of edges connected to the SPC VNs in systematic form. On the contrary, in the second case we obtain a D-GLDPC ensemble where the fraction of edges connected to the $(d_{\min} = 2)$ VNs is partitioned almost equally between the length-2 repetition VNs and the SPC VNs. This ensemble exhibits the best threshold among the obtained distributions. Both distributions exhibit an excellent derivative matching as q^* is very close to $P^{-1}(1/C)$.

Let us now consider the two columns of Table I labeled as "D-GLDPC S_2" and "D-GLDPC C_2," corresponding to the optimization of the D-GLDPC ensembles employing the SPC VNs in systematic and in cyclic form, respectively, subject to the further constraint ($\mathcal{A}$ in the optimization problem formalization) that the fraction of edges connected to the BCH CNs is at least 0.70. Such a constraint is usually intended to lower the error floor. In this case, we also observe that the optimal distribution where SPC VNs in systematic form are allowed in $\mathcal{B}_V$ is actually a GLDPC distribution. This distribution is characterized by a large threshold loss compared with that of the D-GLDPC S_1 distribution (that is compared with the optimal distribution when removing the additional constraint $\mathcal{A}$). On the other hand, the D-GLDPC C_2 distribution makes heavy use of SPC VNs in cyclic form and, despite the large fraction of edges connected to the BCH CNs, the overall threshold loss is much smaller compared with that observed when allowing SPC VNs is systematic form. Again both distributions exhibit an excellent derivative matching.

E. Discussion

1) Dependence of the Optimal Distribution on the Generalized VNs Representation: From Table I, for the same set of variable component codes and the same set of optimization constraints, we notice a pronounced dependence on the representation of the generalized VNs for both the optimal degree profile and the optimal threshold. In the case where $\mathcal{A}$ is assumed to be the empty set as well as in the case where a lower bound is imposed on the fraction of edges connected to the generalized CNs, DE converges to a GLDPC distribution if the systematic form of the SPC VNs is adopted. DE converges to a different distribution, namely, a D-GLDPC distribution, if the cyclic form is adopted.

To further elaborate this point, we depict in Fig. 5 the EXIT functions over the BEC of a $(15, 14)$ SPC VN represented in systematic form (solid curve) and in cyclic form (dotted curve), assuming a BEC erasure probability $q = 0.493$ (a value close to the thresholds of the D-GLDPC S_1 and D-GLDPC C_1 distributions). We notice how different representations of the same VN may be associated with EXIT functions with quite a different behavior. In particular, we observe that the EXIT function associated with the systematic form is flat for sufficiently small values of I_A. This feature makes it problematic to obtain a good curve fitting in the EXIT chart, at small values of I_A, exploiting the SPC VNs in systematic form. Coherently, DE returns a zero fraction of edges connected to these VNs, only exploiting length-2 repetition codes as minimum distance 2 VNs. On the other hand, the same feature is not shared by the EXIT function of the cyclic representation. The SPC VNs in cyclic form are effectively combined by DE with the other VNs to reduce the area gap between the curves.

2) Robustness to Generalized CNs: The distributions D-GLDPC S_1 and D-GLDPC S_2 in Table I may be also interpreted as the optimal GLDPC distributions with design rate $R = 1/2$ returned by DE for the same sets $\mathcal{B}_C$ and $\mathcal{A}$, and for $\mathcal{B}_V$ including no generalized VN types. Therefore, a comparison of the D-GLDPC S_2 and D-GLDPC C_2 distributions allows us to discuss the role played by the SPC VNs is cyclic form, in the presence of a large fraction of edges connected to the BCH CNs.

As expected, the optimal GLDPC distribution is characterized by a large fraction of length–2 repetition VNs. In this specific example, in the optimal distribution returned by DE, all the VNs are repetition-2 codes. They provide the necessary rate compensation to obtain an overall rate $1/2$ and allow a very good derivative matching at $I_A = 1$ to be achieved (as the stability bound is close to the threshold). On the other hand, the optimal D-GLDPC ensemble with SPC VNs in cyclic form makes a moderate use of the length–2 repetition VNs as most of the rate compensation is performed by the high rate $r = 14/15$ SPC VNs. This fact is very beneficial as the rate compensation is achieved with a smaller fraction of edges connected to the VNs with minimum distance 2 (0.654417 versus 1.0), allowing DE to exploit also low-rate VN types for the purpose of curve fitting, specifically the repetition VNs of length 14 and 15.

In Figs. 6 and 7, we illustrate the EXIT charts of the D-GLDPC S_2 and D-GLDPC C_2 distributions, respectively, at $q = q^*$. In the first case, all the VNs are length 2 repetition codes, so that the VN set EXIT function is linear in I_A. The optimal distribution returned by DE achieves a good derivative matching a $I_A = 1$, but is affected by a large unfilled gap between the EXIT curves at lower values of I_A. Consequently, its threshold is quite poor. In the second case, we observe a much better curve fitting obtained by combining high rate $d_{\min} = 2$ VNs, helpful in filling the area gap at values of I_A close to 1, and very low rate VNs, helpful in tightly fitting the inverse CN set EXIT curve at smaller values of I_A. Overall, the D-GLDPC code ensemble exhibits a robustness to the introduction of a large fraction of edges connected to the BCH CNs as the threshold loss with respect to the unconstrained optimal distribution is much smaller than in the GLDPC case.

3) SPC VNs and Stability Bound: Let us consider again Fig. 4. Since for the D-GLDPC C_2 distribution, the total fraction of edges connected to the VNs with minimum distance 2 is about 0.654 and the fraction of edges connected to the SPC VNs is about 0.522, the dotted curve associated with $\mu = 0.522$ represents the function $P^{-1}(x)$ for the D-GLDPC C_2 distribution. This curve allows us to appreciate the useful role played by the SPC VNs to compensate for the loss in the parameter C discussed in Section III-B. The D-GLDPC C_1 distribution is characterized by $C = 5.165712$. In that case a large fraction of edges (0.860952) are connected to SPC CNs contributing to C. On the other hand, the D-GLDPC C_2 distribution is characterized by $C = 1.112804$, i.e., by quite a smaller value of C. This is due to the imposition of a large fraction of edges connected to the BCH CNs that do not contribute to C since their minimum distance is larger than 2. We observe that for $C = 1.112804$ and $R = 1/2$ we have $C < 1/(1 - R)$, implying that such a value of C is not compatible with the derivative matching condition in the context of GLDPC codes. However, when passing from the D-GLDPC C_1 distribution to the D-GLDPC C_2 distribution, we observe a significant increase in the fraction of edges connected to the SPC VNs. The effect is to obtain a

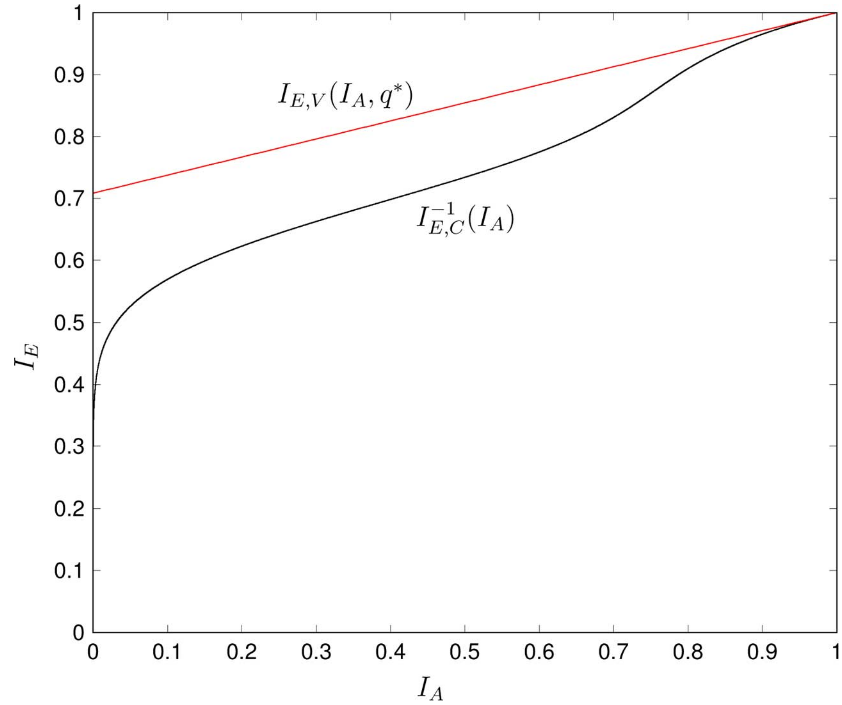

Fig. 6. EXIT chart for the D-GLDPC S_2 distribution in Table I.

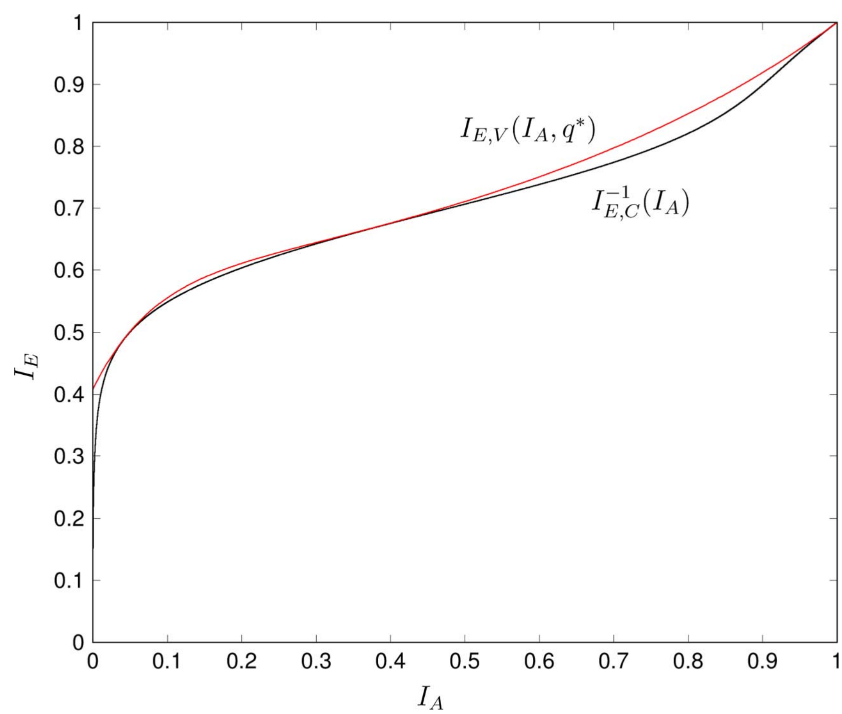

Fig. 7. EXIT chart for the D-GLDPC C_2 distribution in Table I.

function $P^{-1}(x)$ achieving the stability bound with equality, with a value of threshold still close to $1-R$.

Referring to Fig. 4, we observe that the dotted curve for $\mu = 0.522$ takes a value very close to 0.5 for $x = 1/1.112804 = 0.898631$ while the dashed here takes a value larger than 1, as expected from this discussion. Observing the solid curve associated with $\mu = 0.522$, we also note that for the same variable and check component code types and distributions $\lambda(x)$ and $\rho(x)$ (and then for the same R), a much smaller threshold could be potentially achieved using the systematic representation, instead of the cyclic one. This was expected from the discussion in Section III-B2 and with the comparison between the two representations, in terms of $P_i(x)$, provided in Section IV-C.

V. Conclusion

In this paper, a stability bound over the BEC has been developed for D-GLDPC codes. It generalizes the analogous bound valid for LDPC code ensembles. For D-GLDPC codes, as for

LDPC codes, the only variable and check component codes contributing to the bound are those having minimum distance 2. A derivative matching condition sufficient to achieve the bound with equality has also been defined. If this condition is fulfilled, then the decoding threshold over the BEC for D-GLDPC codes is expressed by a simple formula, although in general not in closed form. For GLDPC codes, this formula always leads to a closed-form threshold expression. The stability bound analysis has suggested SPC codes used as variable nodes as an appealing option to overcome common problems created by generalized CNs. The effectiveness of this approach has been shown through a numerical example.

APPENDIX I
PROOF OF THEOREM 2

Let us first focus on the derivative of the CN set EXIT function at $p = 0$. From (19) we have

$$\left.\frac{\mathrm{d}I_{E,C}(p)}{\mathrm{d}p}\right|_{p=0} = -\rho'_{\mathrm{SPC}}(1) + \sum_{i\in\mathcal{G}_C} \rho_i \left.\frac{\mathrm{d}I_E^{(i)}(p)}{\mathrm{d}p}\right|_{p=0}. \quad (36)$$

In order to develop (36), it is necessary to explicit the derivative at $p = 0$ of each generalized CN type EXIT function. The following lemma shows that the RHS of (36) only depends on the CN types with minimum distance $d_{\min} = 2$. It relates the RHS of (36) to the multiplicity of the codewords with Hamming weight 2 of such CN types.

Lemma 3: The first derivative at $p = 0$ of the CN set EXIT function for a D-GLDPC code ensemble over the BEC, assuming MAP erasure correction at each check component code, satisfies

$$\left.\frac{\mathrm{d}I_{E,C}(p)}{\mathrm{d}p}\right|_{p=0} = -\sum_{i\in\mathcal{I}_{C,2}} \rho_i \frac{2A_2^{(i)}}{n_i} = -C.$$

Proof: See Appendix II. □

The derivative at $p = 0$ of the inverse CN set EXIT function $I_{E,C}^{-1}(p)$ is then given by

$$\left.\frac{\mathrm{d}I_{E,C}^{-1}(p)}{\mathrm{d}p}\right|_{p=0} = -\frac{1}{C}. \quad (37)$$

Next, we develop the partial derivative of the VN set EXIT function, with respect to p and evaluated at $p = 0$. It follows from (18) that

$$\left.\frac{\partial I_{E,V}(p,q)}{\partial p}\right|_{p=0} = -q\,\lambda'_{(\mathrm{rep})}(0) + \sum_{i\in\mathcal{G}_V} \lambda_i \left.\frac{\partial I_E^{(i)}(p,q)}{\partial p}\right|_{p=0}. \quad (38)$$

In order to develop the summation over the generalized VN types in the RHS of (38), we have to explicit the partial derivative with respect to p of each generalized VN type EXIT function, evaluated at $p = 0$. The following lemma provides a simple expression for (38), revealing how its RHS only depends on the VN types with minimum distance $d_{\min} = 2$.

Lemma 4: The partial derivative with respect to p and evaluated at $p = 0$, of the VN set EXIT function for a D-GLDPC code ensemble over the BEC, assuming MAP erasure correction at each variable component code, satisfies

$$\left.\frac{\partial I_{E,V}(p,q)}{\partial p}\right|_{p=0} = -\sum_{i\in\mathcal{I}_{V,2}} \lambda_i \sum_{u=1}^{k_i} \frac{2\,B_{2,u}^{(i)}}{n_i}\, q^u = -P(q). \quad (39)$$

Proof: See Appendix III. □

Using (37) and (39) into (20) we obtain the following expression for the stability bound of D-GLDPC code ensembles over the BEC:

$$P(q^*) \le \frac{1}{C}. \quad (40)$$

The left-hand side (LHS) of (40) is a real polynomial $P(\cdot)$ in the variable q^*. This polynomial can be written as $P(q^*) = \sum_{i\in\mathcal{I}_{V,2}} \lambda_i\, P_i(q^*)$, where $P_i(x)$, defined in (23), is a degree-k_i real polynomial associated with the ($d_{\min} = 2$) type-i VNs. Each $P_i(\cdot)$ is a monotonically increasing function (since all its coefficients are positive). Consequently, $P(\cdot)$ is a monotonically increasing function and its inverse $P^{-1}(\cdot)$ exists. By applying $P^{-1}(\cdot)$ to both terms of (40) we finally obtain (26).

APPENDIX II
PROOF OF LEMMA 3

Let us consider an (n,k) generalized CN with EXIT function $I_E(p)$ in the form (14). Then

$$\left.\frac{\mathrm{d}I_E(p)}{\mathrm{d}p}\right|_{p=0} = \frac{(n-1)a_0 - a_1}{n}.$$

The generator matrix of the check component code is full rank by definition, so $\tilde{e}_n = k$. Because of Proposition 3 in Appendix IV, removing any single column from a linear block code generator matrix does not reduce its rank if and only if $d_{\min} \ge 2$, in which case we obtain $\tilde{e}_{n-1} = n\,k$, so that $a_0 = n\,\tilde{e}_n - \tilde{e}_{n-1} = n\,k - n\,k = 0$. As noted in Section I, the hypothesis $d_{\min} \ge 2$ is always assumed in this paper. Then, we can assume $a_0 = 0$.

If $d_{\min} \ge 2$ for the CN we obtain

$$\left.\frac{\mathrm{d}I_E(p)}{\mathrm{d}p}\right|_{p=0} = -\frac{a_1}{n}$$

with $a_1 = (n-1)\tilde{e}_{n-1} - 2\,\tilde{e}_{n-2} = k\,n\,(n-1) - 2\,\tilde{e}_{n-2}$. By applying Proposition 3, we obtain

$$a_1 = 0, \quad \text{if } d_{\min} \ge 3 \quad (41)$$

and

$$a_1 > 0, \quad \text{if } d_{\min} = 2. \quad (42)$$

In fact, if the CN exhibits a minimum distance $d_{\min} \ge 3$, then removing any pair of columns from the generator matrix does not affect the rank. In this case $2\,\tilde{e}_{n-2} = 2\,k\binom{n}{2} = k\,n\,(n-1)$, hence $a_1 = 0$.

According to these results, the only generalized CNs that contribute to the summation in the RHS of (36) are those characterized by $d_{\min} = 2$. By recalling that all the SPC codes have minimum distance 2, we conclude that (36) only depends on the check component codes with $d_{\min} = 2$. Therefore, the derivative at $p = 0$ of the CN set EXIT function can be then expressed as

$$\left.\frac{\mathrm{d}I_{E,C}(p)}{\mathrm{d}p}\right|_{p=0} = -\rho'_{\mathrm{SPC}}(1) - \sum_{i\in\mathcal{G}_{C,2}} \rho_i \frac{k_i n_i(n_i-1) - 2\,\tilde{e}^{(i)}_{n-2}}{n_i} = -\rho'_{\mathrm{SPC}}(1) - \sum_{i\in\mathcal{G}_{C,2}} \rho_i \frac{2\Delta^{(i)}_{n-2}}{n_i} \tag{43}$$

where

$$\Delta^{(i)}_{n-2} \triangleq \frac{k_i n_i(n_i-1)}{2} - \tilde{e}^{(i)}_{n-2} \tag{44}$$

and where the superscript (i) in both $\tilde{e}^{(i)}_{n-2}$ and $\Delta^{(i)}_{n-2}$ is used to define the dependence of these parameters on the CN type explicitly. The parameter $\Delta^{(i)}_{n-2}$ does not depend on the chosen representation for the ith generalized CN type.

Let us consider an (n,k) generalized check component code with $d_{\min} = 2$, and let $\mathcal{S}_{n-2}$ be the generic $(k \times (n-2))$ matrix obtained by removing two columns from (any representation of) its generator matrix. Because of Proposition 2 in Appendix IV, it follows that removing any single column cannot reduce the rank, and therefore the rank of $\mathcal{S}_{n-2}$ is either k or $k-1$. Denoting by $\sum_{\mathcal{S}_{n-2}}$ the summation over all the possible $\binom{n}{2}$ submatrices $\mathcal{S}_{n-2}$, we have

$$\begin{aligned}\Delta_{n-2} &= \frac{kn(n-1)}{2} - \tilde{e}_{n-2}\\ &= \sum_{\mathcal{S}_{n-2}} k - \sum_{\mathcal{S}_{n-2}} \mathrm{rank}(\mathcal{S}_{n-2})\\ &= \sum_{\mathcal{S}_{n-2}} (k - \mathrm{rank}(\mathcal{S}_{n-2}))\end{aligned}$$

where each term in the summation is either equal to 0 or to 1. Denoting by $\overline{\mathcal{S}}_{n-2}$ the submatrix composed of the two removed columns, by Proposition 2 in Appendix IV any such term is equal to 1 if and only if $\overline{\mathcal{S}}_{n-2}$ covers a (necessarily weight-2) codeword. Since each $\overline{\mathcal{S}}_{n-2}$ can cover at most one weight-2 codeword, and since there are no two different $\overline{\mathcal{S}}_{n-2}$ covering the same weight-2 codeword, we have $\Delta_{n-2} = A_2$. Therefore, we can write (43) as

$$\left.\frac{\mathrm{d}I_{E,C}(p)}{\mathrm{d}p}\right|_{p=0} = -\rho'_{\mathrm{SPC}}(1) - \sum_{i\in\mathcal{G}_{C,2}} \rho_i \frac{2\,A^{(i)}_2}{n_i} \tag{45}$$

We can further simplify (45) by noting that

$$\rho'_{\mathrm{SPC}}(1) = \sum_{i\in\mathcal{S}_C} \rho_i\,(i-1) = \sum_{i\in\mathcal{S}_C} \rho_i \frac{2\,A^{(i)}_2}{n_i}$$

as for a SPC CN $n_i = i$ and $A^{(i)}_2 = \binom{i}{2} = i\,(i-1)/2$. Equation (37) follows.

Appendix III
Proof of Lemma 4

Let us consider an (n,k) generalized VN whose EXIT function is given by (11), and let us define

$$f_z(p) \triangleq \sum_{t=0}^{n-1} a_{t,z}\, p^{\,t}\,(1-p)^{n-1-t},$$

where $a_{t,z}$ is defined in (12). Then

$$\left.\frac{\mathrm{d}f_z(p)}{\mathrm{d}p}\right|_{p=0} = -(n-1)a_{0,z} + a_{1,z}$$

from which we have

$$\left.\frac{\partial I_E(p,q)}{\partial p}\right|_{p=0} = -\frac{1}{n}\sum_{z=0}^{k} \left.\frac{\mathrm{d}f_z(p)}{\mathrm{d}p}\right|_{p=0} q^z\,(1-q)^{k-z} = \sum_{z=0}^{k} \frac{(n-1)a_{0,z} - a_{1,z}}{n}\, q^z\,(1-q)^{k-z}. \tag{46}$$

The expression (46) can be further simplified by invoking Proposition 3 in Appendix IV. Since any variable component code has minimum distance $d_{\min} \geq 2$ by hypothesis, removing any single column from the generator matrix $\boldsymbol{G}$ of the variable component code cannot reduce the rank of $\boldsymbol{G}$. It follows that

$$\begin{aligned}a_{0,z} &= n\,\tilde{e}_{n,k-z} - \tilde{e}_{n-1,k-z}\\ &= k\,n\binom{k}{k-z} - k\,n\binom{k}{k-z}\\ &= 0\end{aligned}$$

thus leading to

$$\left.\frac{\partial I_E(p,q)}{\partial p}\right|_{p=0} = -\sum_{z=0}^{k} \frac{a_{1,z}}{n}\, q^z\,(1-q)^{k-z}.$$

Proposition 3 in Appendix IV can be invoked again in order to show that

$$a_{1,z} = 0 \quad \forall z, \qquad \text{if } d_{\min} \geq 3 \tag{47}$$

and

$$a_{1,z} = kn(n-1)\binom{k}{k-z} - 2\tilde{e}_{n-2,k-z} > 0 \quad \forall z, \quad \text{if } d_{\min} = 2 \tag{48}$$

where $d_{\min}$ is the variable component code minimum distance. In fact, under the hypothesis $d_{\min} \geq 3$, removing any single column or any pair of columns from (any representation of) $\boldsymbol{G}$ cannot reduce its rank. Under this hypothesis

$$\begin{aligned}a_{1,z} &= (n-1)\,\tilde{e}_{n-1,k-z} - 2\,\tilde{e}_{n-2,k-z}\\ &= k\,n\,(n-1)\binom{k}{k-z} - 2\,k\binom{n}{n-2}\binom{k}{k-z}\\ &= 0.\end{aligned}$$

Hence, the only generalized variable component codes contributing to the RHS of (38) are those with minimum distance $d_{\min}=2$. Introducing $a_{0,z}=0$ and (48) in (46), and then using (46) in (38), we can write this latter equation as

$$\left.\frac{\partial I_{E,V}(p,q)}{\partial p}\right|_{p=0} = -q\,\lambda'_{(\text{rep})}(0) - \sum_{i\in\mathcal{G}_{V,2}} \lambda_i \times \sum_{z=0}^{k_i} \frac{k_i\, n_i(n_i-1)\binom{k_i}{k_i-z} - 2\,\tilde{e}^{(i)}_{n-2,k-z}}{n_i}\, q^z\,(1-q)^{k_i-z} = -q\,\lambda'_{(\text{rep})}(0) - \sum_{i\in\mathcal{G}_{V,2}} \lambda_i \sum_{z=0}^{k_i} \frac{2\Delta^{(i)}_{n-2,k-z}}{n_i} q^z\,(1-q)^{k_i-z} \tag{49}$$

where

$$\Delta^{(i)}_{n-2,k-z} \triangleq \frac{k_i n_i(n_i-1)}{2}\binom{k_i}{k_i-z} - \tilde{e}^{(i)}_{n-z,k-z} \tag{50}$$

and where the superscript (i) to both $\tilde{e}^{(i)}_{n-z,k-z}$ and $\Delta^{(i)}_{n-2,k-z}$ is used to define the dependence on the VN type explicitly. As opposed to $\Delta^{(i)}_{n-2}$ in (44), $\Delta^{(i)}_{n-2,k-z}$ in (50) depends on the component code representation.

Next we claim the for an (n,k) generalized variable component code with $d_{\min}=2$ we have

$$\sum_{z=0}^{k} \frac{2\,\Delta_{n-2,k-z}}{n} q^z\,(1-q)^{k-z} = \sum_{u=1}^{k} \frac{2\,B_{2,u}}{n}\, q^u. \tag{51}$$

Equation (51) is proved at the end of this appendix. It allows us to write

$$\left.\frac{\partial I_{E,V}(p,q)}{\partial p}\right|_{p=0} = -q\,\lambda'_{(\text{rep})}(0) - \sum_{i\in\mathcal{G}_{V,2}} \lambda_i \sum_{u=1}^{k_i} \frac{2\,B^{(i)}_{2,u}}{n_i}\, q^u. \tag{52}$$

This expression can be further simplified by noting that the only weight-2 codeword of a length-2 repetition VN is $\boldsymbol{c}=[1,1]$, which is generated by a weight-1 information word. Then, for a length-2 repetition VN we have

$$\lambda'_{(\text{rep})} \sum_{u=1}^{k} \frac{2\,B_{2,u}}{n} q^u = \lambda'_{(\text{rep})}\; q.$$

Hence, (52) can be put into the more compact form (39).

Proof of (51): Let $\mathcal{C}$ be a $(d_{\min}=2)(n,k)$ variable component code and let $\boldsymbol{G}$ be the chosen generator matrix for $\mathcal{C}$. We denote by $\mathcal{C}_2$ the subset of $\mathcal{C}$ composed of the weight-2 codewords $\boldsymbol{c}\in\mathcal{C}$. Furthermore, we denote by $\mathcal{C}'$ the $(n+k,k)$ linear block code generated by the concatenated matrix $[\boldsymbol{G}|\boldsymbol{I}_k]$. The generic codeword of $\mathcal{C}$ is denoted by $\boldsymbol{c}$, while the generic codeword of $\mathcal{C}'$ by $\boldsymbol{c}'$. We have $\boldsymbol{c}'=[\boldsymbol{c}|\boldsymbol{u_c}]$, where $\boldsymbol{c}$ and $\boldsymbol{u_c}$ must satisfy $\boldsymbol{c}=\boldsymbol{u_c}\boldsymbol{G}$. The code $\mathcal{C}'$ then depends on the chosen generator matrix representation for $\mathcal{C}$. It is readily shown that $d'_{\min}\geq d_{\min}+1$, where $d_{\min}$ and $d'_{\min}$ are the minimum distances of $\mathcal{C}$ and $\mathcal{C}'$, respectively.

Let $\mathcal{S}_{n-2,k-z}$ be a generic $(k\times(n-2+k-z))$ matrix obtained by selecting $n-2$ columns in $\boldsymbol{G}$ and $k-z$ columns in $\boldsymbol{I}_k$. Let $\mathcal{S}^{\boldsymbol{G}}_{n-2}$ be a generic $(k\times(n-2))$ matrix obtained by selecting $n-2$ columns in $\boldsymbol{G}$ and $\mathcal{S}^{\boldsymbol{I}}_{k-z}$ be a generic $(k\times(k-z))$ matrix obtained by selecting $k-z$ columns in $\boldsymbol{I}_k$ (we also use the notation $\mathcal{S}_{n-2,k-z}=[\mathcal{S}^{\boldsymbol{G}}_{n-2}|\mathcal{S}^{\boldsymbol{I}}_{k-z}]$). Let $\overline{\mathcal{S}}^{\boldsymbol{G}}_{n-2}$ be the submatrix of $\boldsymbol{G}$ composed of the two unselected columns, and $\overline{\mathcal{S}}_{n-2,k-z}$ the submatrix composed of the two unselected columns of $\boldsymbol{G}$ and of the z unselected columns of $\boldsymbol{I}_k$. Moreover, let $\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}$ be a generic $\mathcal{S}^{\boldsymbol{G}}_{n-2}$ such that $\overline{\mathcal{S}}^{\boldsymbol{G}}_{n-2}$ covers a weight-2 codeword of $\mathcal{C}$. There is a one-to-one correspondence between weight-2 codewords $\boldsymbol{c}\in\mathcal{C}$ and matrices $\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}$, so that we can denote by $\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}(\boldsymbol{c})$ the matrix $\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}$ such that $\boldsymbol{c}$ is covered by the two removed columns.

Let us apply Proposition 2 in Appendix IV to the code $\mathcal{C}'$. Each codeword $\boldsymbol{c}'\in\mathcal{C}'$ is composed of the concatenation of a codeword $\boldsymbol{c}\in\mathcal{C}$ with one of the possible 2^k sequences of k bits (where by the linearity of $\mathcal{C}$ the all-zero length-k sequence is always concatenated with the all-zero codeword of $\mathcal{C}$). Combining this observation with Proposition 2 in Appendix IV we observe that a necessary (though not sufficient) condition for having $\text{rank}(\mathcal{S}_{n-2,k-z})<k$ is that $\overline{\mathcal{S}}^{\boldsymbol{G}}_{n-2}$ covers a weight-2 codeword of $\mathcal{C}$. In fact, if no weight-2 codeword of $\mathcal{C}$ is covered by $\overline{\mathcal{S}}^{\boldsymbol{G}}_{n-2}$, by Proposition 2 we have $\text{rank}(\mathcal{S}^{\boldsymbol{G}}_{n-2})=k$, and hence $\text{rank}(\mathcal{S}_{n-2,k-z})=k$, regardless of the value assumed by $\text{rank}(\mathcal{S}^{\boldsymbol{I}}_{k-z})<k$. Then, we can write $\Delta_{n-2,k-z}$ as

$$\begin{aligned}\Delta_{n-2,k-z} &\overset{(a)}{=} \frac{kn(n-1)}{2}\binom{k}{k-z} - \tilde{e}_{n-2,k-z}\\ &\overset{(b)}{=} \sum_{\mathcal{S}_{n-2,k-z}} k - \sum_{\mathcal{S}_{n-2,k-z}} \text{rank}(\mathcal{S}_{n-2,k-z})\\ &= \sum_{\mathcal{S}_{n-2,k-z}} (k-\text{rank}(\mathcal{S}_{n-2,k-z}))\\ &\overset{(c)}{=} \sum_{\mathcal{S}^{\boldsymbol{G}}_{n-2}} \sum_{\mathcal{S}^{\boldsymbol{I}}_{k-z}} \left(k-\text{rank}\left(\left[\mathcal{S}^{\boldsymbol{G}}_{n-2}\middle|\,\mathcal{S}^{\boldsymbol{I}}_{k-z}\right]\right)\right)\\ &\overset{(d)}{=} \sum_{\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}} \sum_{\mathcal{S}^{\boldsymbol{I}}_{k-z}} \left(k-\text{rank}\left(\left[\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}\middle|\,\mathcal{S}^{\boldsymbol{I}}_{k-z}\right]\right)\right)\end{aligned} \tag{53}$$

In the previous equation list: (a) is the definition of $\Delta_{n-2,k-z}$; (b) follows from the number of choices of $\mathcal{S}_{n-2,k-z}$ and from the definition of $\tilde{e}_{n-2,k-z}$; (c) follows from $\mathcal{S}_{n-2,k-z}=[\mathcal{S}^{\boldsymbol{G}}_{n-2}|\mathcal{S}^{\boldsymbol{I}}_{k-z}]$; (d) follows from Proposition 2 applied to $\mathcal{C}'$. Using (53) we have

$$\begin{aligned}&\sum_{z=0}^{k} \frac{2\,\Delta_{n-2,k-z}}{n} q^z\,(1-q)^{k-z}\\ &= \frac{2}{n}\sum_{z=0}^{k} \sum_{\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}} \sum_{\mathcal{S}^{\boldsymbol{I}}_{k-z}} \left(k-\text{rank}\left(\left[\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}\middle|\,\mathcal{S}^{\boldsymbol{I}}_{k-z}\right]\right)\right) q^z(1-q)^{k-z}\\ &= \frac{2}{n}\sum_{\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}} \sum_{z=0}^{k}\\ &\quad\times \sum_{\mathcal{S}^{\boldsymbol{I}}_{k-z}} \left(k-\text{rank}\left(\left[\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}\middle|\,\mathcal{S}^{\boldsymbol{I}}_{k-z}\right]\right)\right) q^z\,(1-q)^{k-z}\end{aligned} \tag{54}$$

where the last equality follows from a change in the order of summations.

By hypothesis there are no VNs with minimum distance 1. Then, for a given weight-2 codeword $\boldsymbol{c} \in \mathcal{C}_2$, any submatrix $\mathcal{S}_{n-2,k-z}$ is such that $\overline{\mathcal{S}}_{n-2,k-z}$ can cover at most one codeword of $\mathcal{C}'$, i.e., the codeword $[\boldsymbol{c}|\boldsymbol{u_c}]$ subject to $\boldsymbol{c} = \boldsymbol{u_c}\boldsymbol{G}$. If we denote by $w_H(\boldsymbol{u_c})$ the Hamming weight of $\boldsymbol{u_c}$, for each weight-2 codeword $\boldsymbol{c} \in \mathcal{C}_2$ the summation over z in (54) can always start from $w_H(\boldsymbol{u_c})$. In fact, for $z = 0, \ldots, w_H(\boldsymbol{u_c}) - 1$ it is not possible for $\overline{\mathcal{S}}_{n-2,k-z}$ to cover the codeword $[\boldsymbol{c}|\boldsymbol{u_c}]$, hence, $k - \mathrm{rank}([\mathcal{S}^{\boldsymbol{G}}_{n-2}|\mathcal{S}^{\boldsymbol{I}}_{k-z}]) = 0$ due to Proposition 2 in Appendix IV. That allows writing the RHS of (54) as

$$\frac{2}{n}\sum_{\boldsymbol{c}\in\mathcal{C}_2}\sum_{z=w_H(\boldsymbol{u_c})}^{k} \times \sum_{\mathcal{S}^{\boldsymbol{I}}_{k-z}}\left(k - \mathrm{rank}\left(\left[\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}(\boldsymbol{c})\middle|\,\mathcal{S}^{\boldsymbol{I}}_{k-z}\right]\right)\right) q^z\,(1-q)^{k-z}. \tag{55}$$

For a given $z \geq w_H(\boldsymbol{u_c})$, the codeword $[\boldsymbol{c}|\boldsymbol{u_c}]$ is covered by exactly $\binom{k-w_H(\boldsymbol{u_c})}{z-w_H(\boldsymbol{u_c})}$ matrices $\overline{\mathcal{S}}_{n-2,k-z}$. In fact, out of the z columns of $\overline{\mathcal{S}}_{n-2,k-z}$ corresponding to $\boldsymbol{I}_k$, $w_H(\boldsymbol{u_c})$ columns must be kept in fixed positions (to cover $\boldsymbol{u_c}$), while the remaining $z - w_H(\boldsymbol{u_c})$ columns can be chosen out of the $k - w_H(\boldsymbol{u_c})$ available positions. Hence, by Proposition 2 in Appendix IV, there are exactly $\binom{k-w_H(\boldsymbol{u_c})}{z-w_H(\boldsymbol{u_c})}$ non-null terms in

$$\left(k - \mathrm{rank}\left(\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}(\boldsymbol{c})\middle|\,\mathcal{S}^{\boldsymbol{I}}_{k-z}\right)\right) q^z\,(1-q)^{k-z}.$$

Deleting from $\boldsymbol{G}$ two columns corresponding to a weight-2 codeword of $\boldsymbol{c}$ reduces the rank of this matrix by one, leading to a rank $k-1$. In fact, considering the VN minimum distance $d_{\min} = 2$, removing the first column cannot reduce the rank (see Proposition 3 in Appendix IV) and removing the second column reduces the rank (see Proposition 2 in Appendix IV) necessarily by one. We can then conclude that each of the $\binom{k-w_H(\boldsymbol{u_c})}{z-w_H(\boldsymbol{u_c})}$ non-null terms in the summation

$$\sum_{\mathcal{S}^{\boldsymbol{I}}_{k-z}}\left(k - \mathrm{rank}([\hat{\mathcal{S}}^{\boldsymbol{G}}_{n-2}(\boldsymbol{c})|\mathcal{S}^{\boldsymbol{I}}_{k-z}])\right)$$

is equal to one, independently of z. Then we can further develop (55) as

$$\frac{2}{n}\sum_{\boldsymbol{c}\in\mathcal{C}_2}\sum_{z=w_H(\boldsymbol{u_c})}^{k}\binom{k-w_H(\boldsymbol{u_c})}{z-w_H(\boldsymbol{u_c})} q^z\,(1-q)^{k-z}. \tag{56}$$

Next we observe that those weight-2 codewords $\boldsymbol{c}$ associated with the same $w_H(\boldsymbol{u_c})$ (i.e., generated by information words having the same weight) produce the same contribution in (56), since only the Hamming weight of the information words $\boldsymbol{u_c}$ matters. This observation allows us to write (56) as

$$\sum_{u=1}^{k}\frac{2\,B_{2,u}}{n}\sum_{z=u}^{k}\binom{k-u}{z-u} q^z\,(1-q)^{k-z} \tag{57}$$

where $B_{2,u}$ is the number of weight-2 codewords $\boldsymbol{c}$ such that $w_H(\boldsymbol{u_c}) = u$. In general, $B_{2,u}$ depends on the variable component code representation. By noting that

$$\sum_{z=u}^{k}\binom{k-u}{z-u} q^z\,(1-q)^{k-z} = q^u$$

we finally obtain (51).

Appendix IV
Reducing A Generator Matrix Rank by Column Elimination

For a given (n,k) linear block code $\mathcal{C}$ and for a given representation $\boldsymbol{G}$ of its generator matrix, we denote by $\mathcal{S}_t$ a generic submatrix obtained by selecting t columns out of $\boldsymbol{G}$, and by $\overline{\mathcal{S}}_t$ the submatrix composed of the $n-t$ unselected columns.

Definition 6: We say that $\overline{\mathcal{S}}_t$ covers a non-null codeword $\boldsymbol{c} \in \mathcal{C}$ when there are no "1" positions of $\boldsymbol{c}$ corresponding to columns belonging to $\mathcal{S}_t$.

Example 1: Let us consider a $(7,3)$ simplex code with generator matrix

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

and let us denote by $\mathcal{S}_2$ the submatrix composed of the last two columns of $\boldsymbol{G}$. Then, the only non-null codeword covered by $\overline{\mathcal{S}}_2$ is $[0,1,1,1,1,0,0]$.

The following result states that in order to reduce the rank of a given generator matrix by column elimination, it is necessary and sufficient that the removed pattern of columns covers at least one non-null codeword.

Proposition 2: For any generator matrix representation of an (n,k) linear block code $\mathcal{C}$, $\mathrm{rank}(\mathcal{S}_t) < k$ if and only if $\overline{\mathcal{S}}_t$ covers at least one non-null codeword.

Proof:

[*Sufficiency*] Suppose that $\overline{\mathcal{S}}_t$ covers a nonzero codeword $\hat{\boldsymbol{c}}$, and consider a representation $\hat{\boldsymbol{G}}$ of the generator matrix where $\hat{\boldsymbol{c}}$ is one of the rows. It follows that removing from $\hat{\boldsymbol{G}}$ the $n-t$ columns associated with $\overline{\mathcal{S}}_t$ reduces the rank because at least one of the rows becomes an all-zero row, so that $\mathrm{rank}(\mathcal{S}_t) < k$. Since any representation of the generator matrix can be obtained from any other representation by row additions only, and since row additions cannot modify the rank of submatrices composed of generator matrix columns, we have $\mathrm{rank}(\mathcal{S}_t) < k$ also for any representation other than $\hat{\boldsymbol{G}}$.

[*Necessity*] Conversely, let us suppose that $\mathrm{rank}(\mathcal{S}_t) < k$ for a given generator matrix representation. Using the same argument as for the sufficiency, we observe that this inequality must be satisfied also for any other representation of the generator matrix. As removing $\overline{\mathcal{S}}_t$ from any generator matrix leads to a $(k \times t)$ matrix with reduced rank, it must be possible to obtain (from any generator matrix representation) a generator matrix where one or more rows have only "0" in those positions corresponding to $\mathcal{S}_t$. All these rows correspond to non-null codewords of $\mathcal{C}$ covered by $\overline{\mathcal{S}}_t$. □

The following result represents a particular case of Proposition 2.

Proposition 3: We have $\mathrm{rank}(\mathcal{S}_t) = k$ for all $\mathcal{S}_t$ if and only if $n - t < d_{\min}$.

Proof:

[*Sufficiency*] Let us suppose that $\mathrm{rank}(\mathcal{S}_t) = k$ for all $\mathcal{S}_t$. By applying Proposition 2 it follows that no submatrix $\overline{\mathcal{S}}_t$ (composed of $n-t$ columns) can cover any codeword. Then $n-t < d_{\min}$.

[*Necessity*] Conversely, let us suppose that $n - t < d_{\min}$. Then, no submatrix $\overline{\mathcal{S}}_t$ (composed of $n-t$ columns) can cover any codeword. By applying Proposition 2 we conclude that $\mathrm{rank}(\mathcal{S}_t) = k$ for all $\mathcal{S}_t$. □

Example 2: All the codewords of the $(7,3)$ simplex code of Example 1 have Hamming weight 4. As one of these codewords is $[1,1,1,0,0,0,1]$, Proposition 2 guarantees that if we remove the first three and the last column from $\boldsymbol{G}$ given in Example 1 (or from any generator matrix obtained performing row summations on $\boldsymbol{G}$) we obtain a (3×3) matrix with rank smaller than 3. On the other hand, by Proposition 3 we know that, even if we remove any set of three or less columns, the rank of $\boldsymbol{G}$ remains unchanged.

In [14], the concept of *independent set* was introduced. Given a $(k \times n)$ rank-r binary matrix, an independent set of size s is defined as any set of s columns such that removing these columns from the matrix leads to a $(k \times (n-s))$ matrix with a rank smaller than r. By Proposition 2 we now state that a necessary and sufficient condition for a set of s columns to be an independent set of a $(k \times n)$ generator matrix is that the s columns cover at least one codeword. Moreover, by Proposition 3 we recognize that any set of $s < d_{\min}$ columns cannot form an independent set for the generator matrix.

ACKNOWLEDGMENT

The authors wish to thank the anonymous reviewers for their valuable comments and the Associate Editor for very efficient handling of the review process.

REFERENCES

[1] R. Gallager, *Low-Density Parity-Check Codes*. Cambridge, MA: MIT Press, 1963.
[2] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inf. Theory*, vol. IT-27, no. 5, pp. 533–547, Sep. 1981.
[3] Y. Wang and M. Fossorier, "Doubly generalized low-density parity-check codes," in *Proc. 2006 IEEE Int. Symp. Information Theory*, Seattle, WA, Jul. 2006, pp. 669–673.
[4] J. Boutros, O. Pothier, and G. Zémor, "Generalized low density (Tanner) codes," in *Proc. 1999 IEEE Int. Conf. Communications*, Vancouver, BC, Canada, Jun. 1999, vol. 1, pp. 441–445.
[5] M. Lentmaier and K. Zhigangirov, "On generalized low-density parity-check codes based on Hamming component codes," *IEEE Commun. Lett.*, vol. 3, no. 8, pp. 248–250, Aug. 1999.
[6] J. P. Tillich, "The average weight distribution of Tanner code ensembles and a way to modify them to improve their weight distribution," in *Proc. 2004 IEEE Int. Symp. Information Theory*, Chicago, IL, Jun./Jul. 2004.
[7] G. Yue, L. Ping, and X. Wang, "Generalized low-density parity-check codes based on Hadamard constraints," *IEEE Trans. Inf. Theory*, vol. 53, no. 3, pp. 1058–1079, Mar. 2007.
[8] N. Miladinovic and M. Fossorier, "Generalized LDPC codes and generalized stopping sets," *IEEE Trans. Commun.*, vol. 56, no. 2, pp. 201–212, Feb. 2008.
[9] J. Chen and R. Tanner, "A hybrid coding scheme for the Gilbert-Elliott channel," *IEEE Trans. Commun.*, vol. 54, no. 10, pp. 1787–1796, Oct. 2006.
[10] G. Liva, W. Ryan, and M. Chiani, "Quasi-cyclic generalized LDPC codes with low error floors," *IEEE Trans. Commun.*, vol. 56, no. 1, pp. 49–57, Jan. 2008.
[11] E. Paolini, M. Chiani, and M. Fossorier, "On the growth rate of irregular GLDPC codes weight distribution," in *Proc. 2008 Int. Symp. Spread Spectrum Techniques and Applications*, Bologna, Italy, Aug. 2008, pp. 790–794.
[12] E. Paolini, "Iterative Decoding Methods Based on Low-Density Graphs," Ph.D. dissertation, Univ. Bologna, Bologna, Italy, May 2007.
[13] Y. Wang and M. Fossorier, "EXIT chart analysis for doubly generalized LDPC codes," in *Proc. 2006 IEEE Global Telecommunications Conf.* , San Francisco, CA, Nov. 2006, pp. 1–6.
[14] 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*, submitted for publication.
[15] Y. Wang and M. Fossorier, "Doubly generalized LDPC codes over the AWGN channel," *IEEE Trans. Commun.*, to be published.
[16] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.
[17] M. Luby, M. Mitzenmacher, M. Shokrollahi, and D. Spielman, "Efficient erasure correcting codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 569–584, Feb. 2001.
[18] T. Richardson, M. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.
[19] T. Richardson and R. Urbanke, "Modern Coding Theory," 2007 [Online]. Available: http://lthcwww.epfl.ch/mct/index.php, preprint
[20] C. Di, R. Urbanke, and T. Richardson, "Weight distribution of low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 4839–4855, Nov. 2006.
[21] 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.
[22] T. Helleseth, T. Kløve, and V. I. Levenshtein, "On the information function of an error-correcting code," *IEEE Trans. Inf. Theory*, vol. 43, no. 2, pp. 549–557, Mar. 1997.
[23] M. Shokrollahi, *Capacity-Achieving Sequences*. Minneapolis, MN: Inst. Mathematics and its Applications (IMA), 2000, vol. 123, IMA Volumes in Mathematics and its Applications, pp. 153–166.
[24] K. Price and R. Storn, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optimiz.*, vol. 11, pp. 341–359, 1997.
[25] M. A. Shokrollahi and R. Storn, "Design of efficient erasure codes with differential evolution," in *Proc. 2000 IEEE Int. Symp. on Information Theory*, Sorrento, Italy, Jun. 2000, p. 5.

Enrico Paolini (S'03–M'07) received the Dr. Ing. degree (with honors) in telecommunications engineering and the Ph.D. degree in telecommunications engineering from the University of Bologna, Bologna, Italy, in 2003 and 2007, respectively.

During his Ph.D. studies he was a Visiting Research Scholar at the University of Hawaii at Manoa. Currently, he holds a postdoctoral position at the Department of Electronics, Computer Science and Systems (DEIS) of the University of Bologna. His research interests include error-control coding (with emphasis on LDPC codes and their generalizations, iterative decoding algorithms, reduced-complexity maximum-likelihood decoding for erasure channels), and distributed radar systems based on ultrawideband. In the field of error-correcting codes, has been involved since 2004 in activities with the European Space Agency (ESA).

Dr. Paolini is a member of the IEEE Communications Society and of the IEEE Information Theory Society.

Marc P. C. Fossorier (S'89–M'90–SM'00–F'06) received the B.E. degree from the National Institute of Applied Sciences (INSA.), Lyon, France, in 1987 and the M.S. and Ph.D. degrees in 1991 and 1994, respectively, all in electrical engineering.

His research interests include decoding techniques for linear codes, communication algorithms, and statistics. He is a recipient of a 1998 NSF Career Development Award. He has served as Associate Editor for the IEEE TRANSACTIONS ON INFORMATION THEORY from 2003 to 2006, as Editor for the IEEE COMMUNICATIONS LETTERS from 1999 to 2008, as Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS from 1996 to 2003, and as Treasurer of the IEEE Information Theory Society from 1999 to 2003. From 2002 to 2008, he was also an Elected Member of the Board of Governors of the IEEE Information Theory Society which he served as Second Vice-President and First Vice-President. He was Co-Chairman of the 2007 International Symposium on Information Theory (ISIT), Program Co-Chairman for the 2000 International Symposium on Information Theory and Its Applications (ISITA), and Editor for the Proceedings of the 2006, 2003, and 1999 Symposia on Applied Algebra, Algebraic Algorithms, and Error Correcting Codes (AAECC).

Marco Chiani (M'94–SM'02) was born in Rimini, Italy, in April 1964. He received the Dr. Ing. degree (*magna cum laude*) in electronic engineering and the Ph.D. degree in electronic and computer science from the University of Bologna, Bologna, Italy, in 1989 and 1993, respectively.

He is a Full Professor at the II Engineering Faculty, University of Bologna, where he is the Chair in Telecommunication. During the summer of 2001 he was a Visiting Scientist at AT&T Research Laboratories in Middletown, NJ. He is a frequent visitor at the Massachusetts Institute of Technology (MIT), where he presently holds a Research Affiliate appointment. His research interests include wireless communication systems, MIMO systems, wireless multimedia, low-density parity-check codes (LDPCC) and UWB. He is leading the research unit of CNIT/University of Bologna on Joint Source and Channel Coding for wireless video and is a consultant to the European Space Agency (ESA-ESOC) for the design and evaluation of error correcting codes based on LDPCC for space CCSDS applications.

Prof. Chiani has chaired, organized sessions, and served on the Technical Program Committees at several IEEE International Conferences. In January 2006, he received the ICNEWS award "For Fundamental Contributions to the Theory and Practice of Wireless Communications." He was the recipient of the 2008 IEEE ComSoc Radio Communications Committee Outstanding Service Award. He is the Past Chair (2002–2004) of the Radio Communications Committee of the IEEE Communication Society and past Editor of Wireless Communication (2000–2007) for the IEEE TRANSACTIONS ON COMMUNICATIONS.