Absorbing Sets of Generalized LDPC Codes

Marco Ferrari*, Alessandro Tomasoni*, Luca Barletta† and Sandro Bellini†
*CNR-IEIIT, Politecnico di Milano, Milan, Italy, Email: marco.ferrari@ieiit.cnr.it, alessandro.tomasoni@ieiit.cnr.it
†Politecnico di Milano, Milan, Italy, Email: luca.barletta@polimi.it, sandro.bellini@polimi.it

***Abstract*—In this paper we propose a definition of Absorbing Sets for binary Generalized LDPC (GLDPC) codes. We show that under practical Max-Log iterative decoding, our AS definition enables a local description of the message evolution with the iterations, with a simplified model very similar to the one used for the analysis of Min-Sum LDPC decoding. Accordingly, these ASs exhibit a threshold behavior also in GLDPC codes.**

***Index Terms*—Generalized Low-Density Parity-Check codes, Error floor, Absorbing sets, Max-Log decoding, Tanner graph.**

I. Introduction

After the introduction of Turbo-Codes and the rediscovery of Low-Density Parity-Check (LDPC) codes, the idea of Generalized LDPC (GLDPC) codes was also recovered from [1]. GLDPC codes raised new attention as a compromise between the two aforementioned classes of iterative concatenated codes, especially because they appeared not to suffer from the *error floor* phenomenon. In [2] and [3] it is proven that in the ensembles of GLDPC codes with Hamming component codes there exist codes with minimum Hamming distance growing linearly with the block-size, even with low Variable Node (VN) degree $d_v = 2$ (GLDPC codes are usually considered with VN-degree 2 since decoding complexity is minimized and the code rate is maximized). In other terms, GLDPC codes show *good spectral shape behavior* [4].

The good spectral shape behavior, although necessary for floorless codes, is not sufficient under suboptimal decoding, such as message passing on graphs with loops. For GLDPC codes it has been observed that iterative decoders can fail, ending without valid decisions, both over the Binary Erasure Channel (BEC) and over the Binary Symmetric Channel (BSC). In [5] the definition of *Stopping Set* (SS) from [6] is generalized to *Stopping Set* of order m, as a subset $\mathcal{S}$ of VNs whose neighboring Check Nodes (CNs) are connected to $\mathcal{S}$ at least m times. In [5] SSs are identified as the main cause of error floors both on the BEC and on the BSC under iterative Hard Bounded Distance decoding. In [7], the asymptotic exponent of the SS size distribution in GLDPC codes is investigated in conjunction with the Hamming weight distribution of the code.

Absorbing sets (ASs), defined in [8], are combinatorial substructures of the Tanner graph in LDPC codes that describe the dominant decoding failures of various soft message passing decoders over AWGN channels [8], [9]. Recently Non-Binary (NB) LDPC codes have gained new interest and the definition of ASs has been extended to NB-LDPC codes in [10]. These ASs are named Generalized AS (GAS) in [11].

Elementary ASs (i.e. with CNs connected no more than twice to the VNs of the AS) enable a linear state-space model for the local analysis of the iterative decoder (see [12] and references therein). In [13] and [14] we studied through a linear state-space model with saturation, the behavior of practical iterative decoders in binary LDPC Tanner graphs with ASs and we defined an AS parameter, the *threshold*, that discriminates the existence/non-existence of misleading equilibria for the iterative decoder.

In this paper we propose a definition of ASs for GLDPC binary codes that captures decoding failures of practical, Max-Log [15] iterative decoders, over AWGN channels. We focus on degree-2 VNs, for which the GAS definition cannot be trivially extended to GLDPC codes. We show that our definition of ASs for GLDPC codes, under Max-Log decoding, enables a linear model similar to that used in [13], [14] for binary LDPC codes. Therefore also GLDPC decoders exhibit a threshold behavior in presence of ASs. We show a couple of examples of GLDPC codes with ASs of size provably smaller than the minimum Hamming distance of the code, that can indeed entrap the iterative decoder. Thereby these ASs are responsible for an error floor whose probability also depends on the multiplicity of these structures inside the graph. Finally, we discuss the problem of the search of these ASs in GLDPC codes with extended Hamming component codes and we check their multiplicity against a probabilistic computation.

II. Generalized LDPC Codes and Notation

A binary regular GLDPC code, with N_v VNs of degree $d_v = 2$ and N_c CNs, is defined by the biadjacency matrix Γ and by the code constraints imposed by the CNs. The CNs could be a mixture of various component codes. In this paper, to keep notation simple, we assume one type of component code only, $\mathcal{C}(N, K)$. The matrix Γ has $d_v = 2$ ones per column, N ones per row, and size $N_c \times N_v$, where $N_c = 2N_v/N$. Each row of Γ has ones in the columns corresponding to the N VNs that are constrained to form a codeword of $\mathcal{C}$. Replacing each 1 in Γ with a column of the parity check matrix H_c of $\mathcal{C}$ we obtain the parity check matrix H of the GLDPC code, of size $(N-K)N_c \times N_v$. The design code rate is $R = 1-(N-K)N_c/N_v = 2R_c - 1$ with $R_c = K/N$.

In Fig. 1 we draw the bipartite graph of a GLDPC code with the above constraints. We order the VNs according to the first set of component codewords, and we let a permutation matrix π assign the VNs to the CNs according to the matrix Γ, and their position inside each codeword of $\mathcal{C}$. Iterative decoding is

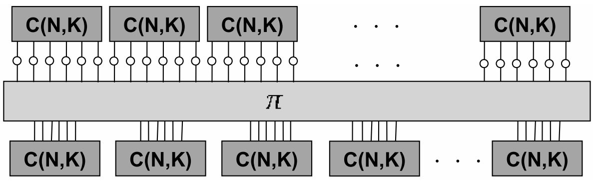

Fig. 1. GLDPC Tanner graph with $d_v = 2$ and component code $\mathcal{C}(N, K)$.

run by activating the CNs on one side of π, then the VNs, then the CNs on the other side of π and finally the VNs again, and then iterating this procedure. An optimal MAP decoder for $\mathcal{C}$, when activated with input LLRs $L_k, (k = 1 \ldots N)$, computes extrinsic messages $E_j, (j = 1 \ldots N)$ for the VNs, by

$$E_j = \log \frac{\sum_{\mathbf{c}\in\mathcal{C}:c_j=+1} \prod_{k=1}^{N} \exp(c_k L_k/2)}{\sum_{\mathbf{c}\in\mathcal{C}:c_j=-1} \prod_{k=1}^{N} \exp(c_k L_k/2)} - L_j. \quad (1)$$

When activated, each VN $v_i, (i = 1 \ldots N_v)$ computes the a posteriori LLR adding the two extrinsic LLRs E_i' and E_i'' received by the neighboring CNs, with the channel LLR λ_i

$$O_i = \lambda_i + E_i' + E_i'' \quad (2)$$

and the input messages for the two CNs by

$$L_i' = \lambda_i + E_i'', \quad L_i'' = \lambda_i + E_i'. \quad (3)$$

III. Absorbing Sets of GLDPC Codes

An absorbing set [8] in LDPC codes is a subset of VNs that, although not forming a codeword, locally satisfy a majority of neighboring CNs of each VN. These subgraphs can lock the iterative decoders to wrong decisions, despite some CNs left unsatisfied, because iterative decoding processes messages only at a local level. Assume to transmit the all-zero codeword, corresponding to symbols $c_i = +1, \forall i$: messages greater than zero correspond to correct decisions, whereas negative messages correspond to errors. Suppose that, at a certain iteration, the decoder has negative decisions for all the VNs of an AS. Satisfied CNs propagate negative messages that reinforce the wrong decisions. Unsatisfied CNs try to correct these values forwarding positive messages, but they are a minority and thus can fail to correct the decisions.

In GLDPC codes, CNs compute messages based on the component code $\mathcal{C}$ as in (1). In practical implementations, MAP decoders (1) are generally replaced by their *Max-Log* versions, and messages are quantized and saturated to a maximal value. Assuming Max-Log decoding and a function *sat* that clips extrinsic messages to their maximum value,[1] (1) is replaced by

$$E_j = sat\left[\max_{\mathbf{c}\in\mathcal{C}:c_j=+1} \sum_{k\neq j}^{N} \frac{c_k L_k}{2} - \max_{\mathbf{c}\in\mathcal{C}:c_j=-1} \sum_{k\neq j}^{N} \frac{c_k L_k}{2}\right] \quad (4)$$

[1]Apart from saturation, in Eq. (4) E_j is linear in the inputs L_k. The saturation level can be set arbitrarily as long as all L_k and E_j are scaled accordingly. In this paper, as in [13], we assume a function $sat(x)$ that clips x to ± 1 and input LLRs L_k scaled by the maximal extrinsic value E_{max}.

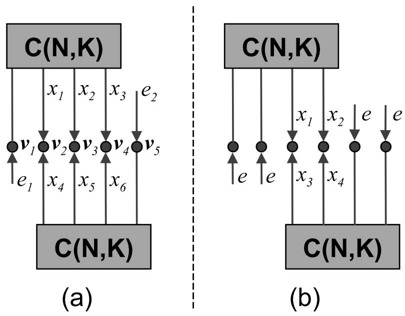

Fig. 2. Examples of Absorbing Sets for GLDPC codes with component codes of minimum Hamming distance $d_H = 4$.

By (4) it is apparent that a CN propagates a negative message E_i whenever the most likely codeword (neglecting L_i) has $c_i = -1$. And this can happen, even with one single negative input message L_k, provided its reliability is higher than the sum of the positive ones. In other words, a simple classification of neighboring CNs *satisfied/unsatisfied* does not capture the harmful subgraphs that entrap the decoder.

In GLDPC codes, a set of VNs with values -1 must match a codeword of $\mathcal{C}$ to locally satisfy the CN. This set must have at least size d_H, where d_H is the minimum Hamming distance of $\mathcal{C}$. Suppose that the two `max` operators in (4) select $\mathbf{c} = +\mathbf{1}$ and a codeword of weight d_H, e.g., without loss of generality, with $c_k = -1$ in positions $k = 1, 2 \ldots d_H$. The output (4) of the CN does not depend on the rest of the LLRs in the component codeword, and it reads

$$E_j = sat\left[L_1 + L_2 + \ldots L_{d_H} - L_j, \right] \quad j = 1 \ldots d_H. \quad (5)$$

Thereby, under Max-Log decoding, we can write a quasi-linear relation between input and output messages inside the set of VNs that correspond to low weight codewords of the component codes.

A. Saturated Linear Model for CN and VN Decoders

For instance, assume that $\mathcal{C}$ is an extended Hamming (eH) code with $d_H = 4$, and that the permutation matrix π allows subgraphs like that drawn in Fig. 2 (a). The subset of VNs $\mathcal{D} = \{v_1, v_2, v_3, v_4, v_5\}$, of size $a = 5$, is the union of the VNs that form two minimum Hamming weight codewords of $\mathcal{C}$, constrained by the two CNs. Let $\mathbf{x} = [x_1, \ldots x_6]^T$ be the extrinsic messages sent by the two CNs to the VNs in $\mathcal{D}_2 \subset \mathcal{D}$ which are connected to both of them, i.e., $\mathcal{D}_2 = \{v_2, v_3, v_4\}$. Using (5), we can write a quasi-linear system (apart from saturation) that describes the iterative activation of VNs and CNs and involves the messages $\mathbf{x}^{(k)}$ generated by the CNs at the kth iteration, the channel LLRs $\boldsymbol{\lambda}$, and the messages $\mathbf{e} = [e_1, e_2]^T$ received by v_1 and v_5, respectively, from CNs outside the subgraph. In matrix form, the system reads

$$\mathbf{x}^{(k)} = sat\left(\mathbf{A}\mathbf{x}^{(k-1)} + \mathbf{R}\mathbf{e} + \mathbf{C}\boldsymbol{\lambda}\right) \quad (6)$$

where the 6×6 *routing matrix* $\mathbf{A}$ and the 6×2 *external LLRs matrix* $\mathbf{R}$ forward the internal extrinsic messages $\mathbf{x}^{(k-1)}$, and the external extrinsic messages $\mathbf{e}$, respectively. The 6×5 *channel LLRs matrix* $\mathbf{C}$ combines the channel LLRs λ_i of each VN $v_i, (i = 1...5)$ inside each message x_j. I.e.,

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix},$$

$$\mathbf{R}^T = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}.$$

Note that, unlike in the state-space linear model of [13], here the CNs generate the linear combinations of internal messages, whereas the degree-2 VNs swap the messages of the two incoming edges. The system (6), that locally describes iterative decoding, allows *misleading equilibria*, i.e. pairs of vectors $(\mathbf{x}, \boldsymbol{\lambda})$ such that $\mathbf{x} = sat\left(\mathbf{Ax} + \mathbf{Re} + \mathbf{C}\boldsymbol{\lambda}\right)$ is stable along the iterations and produces wrong decisions. For instance, the pair $(\mathbf{x}, \boldsymbol{\lambda}) = (-\mathbf{1}, -\mathbf{1})$ is an equilibrium for any vector $\mathbf{e}$ since $\mathbf{e} \leq \mathbf{1}$ because of saturation,[2] and it corresponds to wrong decisions about all VNs in $\mathcal{D}$ since $O_i < 0, i = 1..5$ according to (2). These decisions cannot be changed by further iterations, independently of the messages incoming from the external graph, despite the CNs are not satisfied. In other words, the subgraph of Fig. 2 (a) is an *absorbing set*.

We can look for sufficient conditions for system (6) to converge to an equilibrium corresponding to correct decisions. As in [13], we assume that in the rest of the graph messages converge towards correct decisions and we start the analysis of the iterations when the messages received by $\mathcal{D}$ from external CNs are already saturated to their maximal value, $\mathbf{e} = +\mathbf{1}$. This point of view is chosen to decouple the dynamical behavior of the decoder inside and outside the AS. By this choice we should use an initial vector $\mathbf{x}^{(0)}$ which is the result of the message evolution up to that iteration, which is unknown. Since we are looking for sufficient conditions, we can consider any starting configuration $\mathbf{x}^{(0)}$. If no $\mathbf{x}^{(0)}$ results in a convergence failure, the AS cannot trap the decoder independently of this message evolution.

Note that the ith row weight of $\mathbf{A}$ is equal to the number of internal messages x_j that are added to the external messages e_k to compute the extrinsic LLR x_i as by (5). Since we are assuming that $\mathbf{e} = +\mathbf{1}$, we have $\mathbf{A1} + \mathbf{Re} = (d_H - 1)\mathbf{1}$, and (6) can be rewritten as

$$\mathbf{x}^{(k)} = sat\left(\mathbf{A}\left(\mathbf{x}^{(k-1)} - \mathbf{1}\right) + (d_H - 1)\,\mathbf{1} + \mathbf{C}\boldsymbol{\lambda}\right). \quad (7)$$

If we let $\mathbf{C}\boldsymbol{\lambda} = \boldsymbol{\mu} = [\mu_1, \mu_2 ... \mu_6]^T$, the system (7) is formally identical to the system assumed in [13] and [14]. The only difference is that each entry μ_i is the sum of the $d_H - 1$ independent channel LLRs of the VNs that complete a codeword of weight d_H with the recipient of x_i.

[2]In our notation, $\mathbf{1}$ is the all-ones column vector and the inequalities when applied to vectors are to be meant component-wise.

The formal equivalence of the dynamical system (7) with [14, Eq.(8)], reveals a threshold behavior similar to ASs for binary LDPC codes. Given the equilibrium of the system (7), i.e. pairs $(\mathbf{x}, \boldsymbol{\mu})$ such that $\mathbf{x} = f\left(\mathbf{x}, \boldsymbol{\mu}\right)$ with

$$f\left(\mathbf{x}, \boldsymbol{\mu}\right) = sat\left(\mathbf{A}\left(\mathbf{x} - \mathbf{1}\right) + (d_H - 1)\,\mathbf{1} + \boldsymbol{\mu}\right), \quad (8)$$

we can compute τ_μ defined as

$$\tau_\mu = \max_{(\boldsymbol{\mu}, \mathbf{x})} \quad \min(\boldsymbol{\mu}) \quad (9)$$
$$s.t. \quad -\mathbf{1} \leq \mathbf{x} \leq \mathbf{1},\, \exists j : x_j < 1,\, \mathbf{x} = f\left(\mathbf{x}, \boldsymbol{\mu}\right)$$

In [13] it is proven that if $\mu_i > \tau_\mu, \forall i$, no misleading equilibrium, nor periodic or aperiodic sequence $\mathbf{x}^{(k)}$ is generated by (7) for any initial state $\mathbf{x}^{(0)}$. Thus (7) converges to $\mathbf{x} = +\mathbf{1}$ that is the only equilibrium allowed. This equilibrium leads the VNs to correct decisions, since the a posteriori LLR is equal to $O_i = 2 + \lambda_i$.[3] The corresponding threshold for each channel LLR λ_i can be taken as $\tau = \tau_\mu/(d_H - 1)$. If $\lambda_i > \tau, \forall i$ then $\mu_i > (d_H - 1)\tau = \tau_\mu, \forall i$. Since in this condition there exist no equilibrium with $\mathbf{x} \neq +\mathbf{1}$, the decoder cannot be trapped by this AS.

If no channel LLR can take values below the AS threshold τ, the AS cannot trap the decoder, and we say it is *deactivated*. As shown in [13], a practical way to deactivate an AS with threshold $\tau < 0$, is by setting different saturation levels λ_{max} and E_{max} for the channel and extrinsic LLRs, representing them with a different number of bits, say q_I and q, respectively. If $\tau < -\lambda_{max}/E_{max}$, the AS is deactivated.

The AS of Fig. 2 (a) has threshold $\tau = 0$ since $\tau_\mu = 0$, and it cannot be deactivated. We verified by simulation over a real GLDPC graph, namely $\mathcal{C}_1$, with eH (128,120) component codes and blocksize $N_v = 16384$, that these ASs can indeed trap the iterative decoder. Using $\lambda_{max} = 7$ for the channel LLRs and increasing values E_{max} by using $q = 4, 5$ and 6 bits to represent the extrinsic LLRs, did lower the Word Error Rate (WER) contribution of each one of these ASs (from $3 \cdot 10^{-8}$, to $2.5 \cdot 10^{-9}$ and $2 \cdot 10^{-11}$ respectively, at SNR $E_s/N_0 = 3.6$ dB, 20 iterations) but they could trap the decoder anyway.

A different type of AS, also found in the GLDPC graph, has the subgraph drawn in Fig. 2 (b) and can be analyzed with the same method. Since its threshold $\tau = -2/3$, it can trap the decoder with $q = 4$, but it is deactivated with $q = 5$ since $-\lambda_{max}/E_{max} = -7/15 > \tau$. Importance Sampling (IS) simulation with received vectors biased in the direction of these ASs, did not deliver any error event with $q \geq 5$.

An intuitive picture of the decoders behavior with these ASs is shown in Fig. 3 where the two plots (a) and (b) refer to the ASs of Fig. 2 (a) and (b), respectively. We plot the received vectors that generated a decoding failure, separating the two components r_1 and r_2: r_1 is the component (normalized by $1/a$) along the direction joining the a-length transmitted vector $+\mathbf{1}$ and the a-length AS vector $-\mathbf{1}$; r_2 is the orthogonal component, in the a dimensional subspace. In Fig. 3 (a) we see that error events are registered for any value of q. The

[3]Here, the dynamic range of λ_i is assumed not higher than E_i' and E_i''.

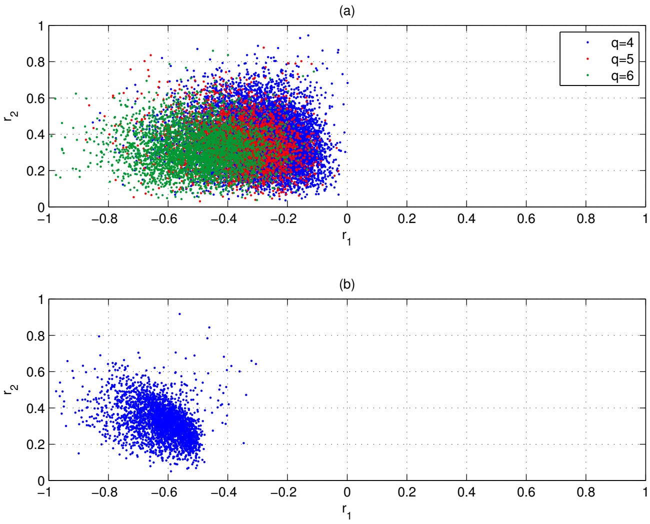

Fig. 3. Error regions for the iterative GLDPC decoders with the small Absorbing Sets of Fig. 2 (a) and (b).

error region becomes slightly smaller increasing q, but does not disappear. On the contrary in Fig. 3 (b) we see error events for the $q = 4$ decoder only, and the error region is faraway from $+\mathbf{1}$. Even with $q = 4$, the WER contribution of each of the AS of Fig. 2 (b) is much smaller (approximately $7 \cdot 10^{-12}$) and the error-floor is dominated by the ASs of Fig. 2 (a).

B. GLDPC Absorbing Sets Definition

The most important difference between the two ASs of Fig. 2 is that their CNs receive a different number of positive messages e from the external graph. We define the *degree of dissatisfaction* (o) of a CN by negative decisions about the VNs in $\mathcal{D}$, as the number of messages received from outside the AS. In the AS we need to consider all CNs with a degree of dissatisfaction $o \leq w - 2$, where w is the Hamming weight of the codeword of $\mathcal{C}$ considered. In fact, a CN with $o = w - 1$ behaves as the external graph. In other words, we include in the AS all the CNs that exchange at least two messages with the VNs of degree two in the AS.

Definition III.1. *In the Tanner graph of a binary GLDPC code, with VN-degree $d_v = 2$, an Absorbing Set is a subgraph with a subset $\mathcal{E}$ of the CNs, and a subset $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2$ of the VNs, where $\mathcal{D}_i$ are the VNs in $\mathcal{D}$ with i neighboring CNs in $\mathcal{E}$, if*

1) *$\mathcal{D}$ is the union of low Hamming weight codewords for each CN $c \in \mathcal{E}$.*
2) *$\mathcal{E}$ is the subset of the neighboring CNs of $\mathcal{D}$, that are connected at least twice to $\mathcal{D}_2$.*

If we are interested in the smallest ASs, we need to consider the minimum weight (d_H) codewords for each CN in $\mathcal{E}$. Each AS can be classified by a triplet (a, b, o) where $a = |\mathcal{D}|$, $b = |\mathcal{E}|$ and o is the degree of dissatisfaction of the CNs in $\mathcal{E}$. For instance, the AS in Fig. 2 (a) is a (5,2,1) AS, whereas in Fig. 2 (b) we have a (6,2,2) AS.[4] We can imagine other ASs but these two have the minimum size a. However, constraints on the adjacency matrix can be easily imposed to exclude these small ASs. In the next subsection we discuss this topic.

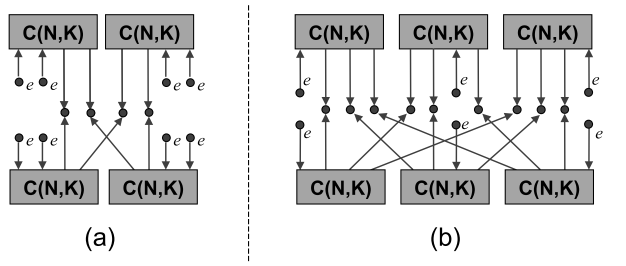

Fig. 4. Smallest ASs compatible with girth-8 constrained adjacency matrix and $d_H = 4$ component codes: AS (12,4,2) (a) and AS (15,6,1) (b).

C. Absorbing Sets of Girth-Constrained GLDPC Codes

Subgraphs like those drawn in Fig. 2, can be found in GLDPC codes with random interleaving π, but they cannot occur in graphs with adjacency matrix Γ of *girth* 8, i.e., with the property that any two CNs share no more than one VN (see, for instance, [16] or [4]). Consider again $d_H = 4$ component codes, but with a girth-8 adjacency matrix Γ. The smallest possible AS that can exist in this GLDPC graph is a (12,4,2) AS, that involves four CNs with $o = 2$ and 12 VNs, and it is represented in Fig. 4 (a). The corresponding system of equations (6) returns a threshold $\tau = -2/3$. These ASs can therefore be easily deactivated.

The smallest possible AS with $o = 1$ for all CNs is the (15,6,1) AS shown in Fig. 4 (b). This is a more dangerous AS, and cannot be deactivated since its threshold is $\tau = 0$. We checked the behavior of these ASs by IS simulation over a GLDPC graph $\mathcal{C}_2$ with girth-8 adjacency matrix Γ built by circulant blocks [16], extended Hamming (64,57) component codes and blocksize $N_v = 32768$ ($R = 25/32$). We verified that the ASs shown in Fig. 4 (b) can trap iterative decoders with $q = 4$, 5 or 6 bits for the representation of the extrinsic LLRs. In particular at $E_s/N_0 = 2.5$ dB we found that each AS contribution to the total WER is $3 \cdot 10^{-19}$, $6 \cdot 10^{-22}$ and $5 \cdot 10^{-26}$ with $q = 4$, 5 or 6, respectively.

IV. Search and Enumeration of GLDPC AS

The error probability due to the ASs with a certain topology also depends on their multiplicity. Their search and enumeration in a specific graph requires the inspection of both the adjacency matrix Γ and of the component codebook $\mathcal{C}$. This search is quite complex in general. Hamming codes exhibit the simplifying property that any pair of ones can be completed with a third one in a specific position to get a codeword. This property is inherited by eH codes: any triplet of ones can be turned into a codeword with a single fourth one in a specific position. Our inspection can thus focus on Γ, enumerating all

[4] In general, different CNs inside an AS could have different degrees o, but in our examples, this does not occur, so we take it as a scalar value.

triplets of VNs shared by two CNs. The two codewords of weight 4 including those VNs can be identified later.

We want to enumerate in $\mathcal{C}_1$ the ASs (5,2,1) with subgraph shown in Fig. 2 (a). The GLDPC code $\mathcal{C}_1$ has blocksize $N_v = 16384$ bits, constrained by $N_c = 2N = 256$ eH(128,120) CNs, $R = 2R_c - 1 = 7/8$, and a purely random permutation matrix π. We stress the fact that the Hamming weight of these ASs ($a = 5$) is smaller than the minimum Hamming distance d_{min} of $\mathcal{C}_1$, since we checked that there is no Hamming weight 4 or 6 codeword allowed by π, hence $d_{min} \geq 8$ in $\mathcal{C}_1$. We have one AS (5,2,1) for every triplet of bits shared by two CNs, which can be enumerated. The number of bits shared by a pair of CNs under a random permutation π, as a first approximation, has a binomial probability distribution of parameter $2/N_c = 1/N$. The expected number of AS (5,2,1) in a code like $\mathcal{C}_1$ is

$$A_5 = N^2 \sum_{k=1}^{N} \binom{k}{3} \binom{N}{k} \frac{1}{N}^k \left(1 - \frac{1}{N}\right)^{N-k} \approx 2667. \quad (10)$$

The exhaustive inspection of $\mathcal{C}_1$ enumerated 2705 ASs (5,2,1). These are responsible for an error-floor at WER $8 \cdot 10^{-5}, 7 \cdot 10^{-6}$ and $5 \cdot 10^{-8}$ for Max-Log decoders with $q = 4$, 5 and 6, respectively (at $E_s/N_0 = 3.6$ dB, 20 iterations).

Later, we chose eH(64,57) component codes for a Quasi-Cyclic GLDPC code $\mathcal{C}_2$ with blocksize $N_v = 32768$, $R = 2R_c - 1 = 25/32$ and a girth-8 adjacency matrix Γ built as in [16]. The matrix Γ has $d_v = 2$ row-blocks of $N = 64$ circulant matrices of size $S \times S$, with $S = 512$. The shifts of the first row-block were set to zero. The shifts of the second row-block $s_1, s_2 ... s_N$ have been chosen randomly, but all distinct to guarantee girth $g = 8$. With $g = 8$ the minimum Hamming distance of the code is $d_{min} \geq 16$ [4] and thus larger than the most critical AS analyzed, of size $a = 15$.

To enumerate the ASs (15,6,1), we need to look in the graph of $\mathcal{C}_2$, for triplets of cycles of length 8 that share 9 VNs and 6 CNs. For each triplet we have exactly one AS (15,6,1). The exhaustive inspection of the graph enumerated about $4400 \times S \approx 2.3 \cdot 10^6$ ASs (15,6,1). We can check this number against a probabilistic argument. Select three VNs, in columns c_1, c_2, c_3 from three different column-blocks of Γ, with shifts s_1, s_2, s_3 in the second row-block. We have $S^3 \binom{N}{3}$ different choices. Pick any two of these three VNs. The probability that there exists a cycle of length 8 across these two VNs is the probability that there exist in Γ two circulant blocks of shifts $s_1 \pm (c_1 - c_2) \mod S$. This probability is $(N/S)^2$ by random choice of the shifts, hence the probability that all the three pairs belong to cycles of length 8 is $(N/S)^6$. Finally, if a triplet of cycles like this exists, it is counted 6 times by this combinatorial argument. As a first approximation, the expected number of these ASs (15,6,1) is

$$A_{15} = \frac{1}{6} S^3 \binom{N}{3} \left(\frac{N}{S}\right)^6 \approx 3 \cdot 10^6. \quad (11)$$

Taking the multiplicity into account, the total estimated WER contribution of these ASs is $7 \cdot 10^{-13}, 10^{-15}$ and $2 \cdot 10^{-19}$ for Max-Log decoders with $q = 4$, 5 and 6, respectively (at $E_s/N_0 = 2.5$ dB, 20 iterations).

V. Conclusions

In this paper we have proposed a definition for combinatorial substructures of the Tanner graph of binary VN-degree 2, GLDPC codes, that can trap practical Max-Log decoders over AWGN channels, i.e., *Absorbing Sets* of GLDPC codes. For these structures we can derive a quasi-linear model that reveals a threshold behavior similar to ASs in binary LDPC codes. The model predictions have been checked via IS simulation over two examples. Design constraints on the adjacency matrix of the code can avoid the smallest structures, but larger ASs able to trap the iterative decoders do exist. In case of extended Hamming component codes we enumerated by exhaustive search the most critical ASs and we checked our results against combinatorial arguments.

References

[1] R. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inf. Theory*, vol. 27, no. 5, pp. 533–547, May 1981.

[2] M. Lentmaier and K. Zigangirov, "On generalized low-density parity-check codes based on Hamming component codes," *IEEE Commun. Letters*, vol. 3, no. 8, pp. 248–250, Aug. 1999.

[3] J. Bouthros, O. Pothier, and G. Zemor, "Generalized low density (Tanner) codes," in *Proc. of IEEE Int. Conf. on Commun. (ICC'99)*, Vancouver, Canada, June 6-10 1999, pp. 441–445.

[4] M. Lentmaier, G. Liva, E. Paolini, and G. Fettweis, "From product codes to structured generalized LDPC codes," in *Proc. of 5th Int. ICST Conf. on Commun. and Networking*, Beijing, China, Aug. 25-27 2010, pp. 1–8.

[5] N. Miladinovic and M. Fossorier, "Generalized LDPC codes and generalized stopping sets," *IEEE Trans. Commun.*, vol. 56, no. 2, pp. 201–212, Feb. 2008.

[6] C. Di, D. Proietti, E. Telatar, T. J. Richardson, and R. Urbanke, "Finite-Length Analysis of Low-Density Parity-Check Codes on the Binary Erasure Channel," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1570 – 1579, June 2002.

[7] E. Paolini, M. Flanagan, M. Chiani, and M. Fossorier, "Spectral shape of check-hybrid GLDPC codes," in *Proc. IEEE Int. Conf. Communications*, Capetown, SA, May 23-27 2010, pp. 1–6.

[8] L. Dolecek, Z. Zhang, V. Anantharam, M. Wainwright, and B. Nikolic, "Analysis of absorbing sets and fully absorbing sets of array-based LDPC codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 181–201, Jan. 2010.

[9] Z. Zhang, L. Dolecek, B. Nikolic, V. Anantharam, and M. Wainwright, "Design of LDPC decoders for improved low error rate performance: quantization and algorithm choices," *IEEE Trans. Commun.*, vol. 57, no. 11, pp. 3258–3268, Nov. 2009.

[10] B. Amiri, J. Kliewer, and L. Dolecek, "Analysis and enumeration of absorbing sets for non-binary graph-based codes," *IEEE Trans. Commun.*, vol. 62, no. 2, pp. 398–409, Feb. 2014.

[11] A. Hareedy, C. Lanka, and L. Dolecek, "A general non-binary LDPC code optimization framework suitable for dense flash memory and magnetic storage," *IEEE J. Select. Areas Commun.*, vol. 34, no. 9, pp. 2402–2415, Sep. 2016.

[12] B. K. Butler and P. H. Siegel, "Error floor approximation for LDPC codes in the AWGN channel," *IEEE Trans. Inf. Theory*, vol. 60, no. 12, pp. 7416–7441, Dec. 2014.

[13] A. Tomasoni, S. Bellini, and M. Ferrari, "Thresholds of absorbing sets in low-density-parity-check codes," *IEEE Trans. Commun.*, vol. 65, no. 8, pp. 3238–3249, Aug. 2017.

[14] M. Ferrari, A. Tomasoni, and S. Bellini, "Analysis of practical LDPC decoders in Tanner graphs with absorbing sets," in *Proc. of IEEE Info. Theory Workshop*, Kaohsiung, Taiwan, Nov. 6-10 2017, pp. 141–145.

[15] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inf. Theory*, vol. 42, no. 2, pp. 429–445, Feb. 1996.

[16] 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.