Design of non binary LDPC codes using their binary image: algebraic properties.

Charly Poulliat ∗† Marc Fossorier † David Declercq ∗

∗ ETIS, ENSEA/UCP/CNRS
6 Avenue du Ponceau
F-95014 Cergy-Pontoise, France.
Email: {charly.poulliat, david.declercq}@ensea.fr

† Dept. of Electrical Eng.
University of Hawaii at Manoa
2540 Dole St., Honolulu, HI-96822, USA.
Email: marc@spectra.eng.hawaii.edu

Abstract—In this paper, we develop algebraic properties of regular \((2,t, r, N)\) non binary LDPC codes designed using their binary image. First, we characterize the algebraic properties of optimized rows of the parity check matrix \(H\) associated with a code, and then we study the algebraic properties of cycles and stopping sets associated with the underlying Tanner graph.

I. INTRODUCTION

Since their rediscovery by [14], Low Density Parity Check (LDPC) codes designed over GF\((q)\) have been shown to approach the Shannon limit performance for \(q = 2\) and very long code lengths [19]. Some efficient optimization methods of the code profile and the matrix structure have been derived for both long [19][3] and moderate [12] length cases. For fields with parameters \(q > 2\), it has been shown that the error performance can be improved for moderate code lengths by increasing \(q [5][4][10]\) and iterative decoding of non binary LDPC codes using the belief propagation (BP) algorithm or its simplified versions has been addressed by several authors [5][1][6].

The optimization of non binary LDPC codes can be addressed in order to meet different objectives: (i) performance, by trying to improve the waterfall region and/or to lower the error floor, and (ii) decoding complexity versus performance tradeoff, by trying to ensure good overall performance using only a limited coefficients set for some efficient and practical hardware implementation purposes. To our knowledge however, there exits no asymptotic method for the optimization of the code profile in the context of a code ensemble analysis. For finite length codes, the optimization problem is generally solved in a disjoint manner. First, the positions of the nonzero entries of the parity check matrix \(H\) associated with the non binary code are optimized in order to have some good girth properties and minimize the impact of the cycles, when using the BP algorithm on the associated Tanner graph. This can be efficiently done using the progressive edge growth (PEG) algorithm [12]. Then, the nonzero entries can be selected either randomly from a uniform distribution among nonzero elements of GF\((q)\) [12] or carefully to meet some optimization criteria as done in [4][15].

Recently, [17] address the problem of the selection and the matching of the parity check matrix nonzero entries assuming that the positions of nonzero entries in the parity check matrix \(H\) associated with the non binary code have been previously optimized. The proposed method is based on the binary image representation of the matrix \(H\) and of its components. First they address the problem of rows optimization as previously done in [5][15] in order to improve the waterfall region. Then, they address the problem of lowering the error floor: based on the observation that the columns involved in the minimum distance of the binary image of \(H\) are located on symbols belonging to the shortest length cycles and the associated stopping sets, they propose a method that intends to improve the minimum distance of the binary image of the code using the algebraic properties of both cycles and stopping sets. In this paper, we intend to develop the algebraic properties of codes designed using the binary image of that kind of codes.

The paper is organized as follows: in Section II, we briefly review the binary image construction of a non binary parity check matrix and the vector representation of the parity check equations. The algebraic properties of the rows of the parity check matrix are reviewed in Section III. Section IV provides a study of the binary representation of both cycles and stopping sets, and establishes links between those and the minimum distance property of the code. This study allows us to derive some bounds on the achievable minimum distance to get an insight of the behavior with respect to the code length \(N\). Finally conclusions and perspectives are drawn in Section V.

II. BINARY IMAGES OF A NON BINARY PARITY CHECK MATRIX \(H\)

Let us consider the parity check matrix \(H\) associated with a regular non binary LDPC code with the parameters \((t_c, t_r, N)\) representing the number of nonzero entries of \(H\) for the columns, the rows and the code length respectively. All the nonzero elements of \(H\) are elements of the Galois fields GF\((q)\), with \(q = 2^p\) and \(q\) is the order of the field. Nonzero elements belong to the set \(\mathcal{S} = \{\alpha^k : k = 0 \ldots q - 2\}\) where \(\alpha\) is the primitive element of the field.

A. Representation of the Galois field using matrices

The Galois field GF\((q)\), described usually using a polynomial (or vector) representation, can be also represented using
matrices [16]

**Definition 1:** If \( p(x) = a_0 + a_1 x + \ldots + a_p x^p \) is a polynomial of degree \( p \) having its coefficients in GF(2). The companion matrix of \( p(x) \) is the \( p \times p \) matrix

\[
A = \begin{pmatrix}
0 & 1 & 0 & \ldots & 0 \\
0 & 0 & 1 & \ldots & 0 \\
0 & 0 & 0 & \ldots & 1 \\
a_0 & a_1 & a_2 & \ldots & a_{p-1}
\end{pmatrix}
\]

The characteristic polynomial of this matrix is given by

\[
\det(A - xI) = p(x)
\]

where \( I \) is the identity matrix.

It can be shown [16] that the matrix \( A \) is the primitive element of the Galois field \( GF(2^p) \) under a matrix representation and thus the powers of \( A \) are the nonzero elements of this field, defining the set \( M = \{0, A^k : k = 0 \ldots q-2 \} \). Additions and multiplications in the field correspond to additions and multiplications of these matrices.

**B. Vector representation for the parity check equations**

Based on the matrix representation of each nonzero entry, we give thereafter the equivalent vector representation of the parity check equations associated with the rows of \( H \).

Let \( x = [x_1 \ldots x_N] \) be a codeword with \( N \) components. For the \( i \)-th parity equation of \( H \), we have

\[
\sum_{j : h_{ij} \neq 0} h_{ij} x_j = 0
\]

Translating (1) into the vector domain, we can write

\[
\sum_{j : h_{ij} \neq 0} H_{ij} x_j^t = 0^t
\]

where \( H_{ij} \) is the transpose of the matrix representation of the Galois field element \( h_{ij} \), \( x_j \) is the vector representation (binary mapping) of the symbol element \( x_j \) and \( ^t \) holds for transpose. \( 0 \) is all zero component vector.

Considering the \( i \)-th parity check equation of \( H \), we define

\[
H_i = [H_{i,j_0} \ldots H_{i,j_m} \ldots H_{i,j_{t-1}}]
\]

as the equivalent binary parity check matrix, with \( \{j_m : m = 0 \ldots t - 1 \} \) the indexes of the nonzero elements of the \( i \)-th row. Let \( X_i = [x_{j_0} \ldots x_{j_{t-1}}] \) be the binary representation of the symbols of the codeword \( x \) involved in the \( i \)-th parity check equation. When using the binary representation, the \( i \)-th parity check equation of \( H \), can be written as

\[
H_i X_i^t = 0^t
\]

We note \( d_{\text{min}}(i) \) the minimum distance of the binary code associated with \( H_i \).

**III. ALGEBRAIC PROPERTIES OF ROWS**

In this section, we investigate the algebraic properties of “good” rows for the parity check matrix regardless of the structure of the Tanner graph associated with it, if the design is performed using their binary image. In [17], the author compare the method proposed in [5][15] based on an instance of density evolution with their proposed method based on the algebraic properties of the binary image of the rows. They first show that the set of rows provided by [15] can easily be reduced and they give an analysis of these coefficient sets using the binary images of the code considered. By comparing the results of both methods, they observe that the best coefficient sets they obtain using the binary image of rows may encompass the sets given by [15]. Finally, a brief threshold study has shown that selecting good rows may improve the waterfall. After briefly reviewing the selection of rows using the binary image, we study the minimum distance properties of the rows for different configurations.

**A. Optimization using binary images**

We present the selection method that aims to select good rows using the equivalent binary parity check matrix of the rows. The optimization idea is that the higher \( d_{\text{min}} \) is, the more distinguishable, hence reliable, the messages passed from check nodes to data nodes using BP are. Therefore considering the equivalent binary parity check matrix \( H_1 \), we intend to maximize the minimum distance \( d_{\text{min}} \) of the associated binary code. Thus, the best \( t_r \)-uplets candidates are those with the largest \( d_{\text{min}} \) and among those \( t_r \)-uplets with maximum \( d_{\text{min}} \), the best are those with the smallest weight enumerator coefficient \( A(d_{\text{min}}) \).

1) Search procedure: Since finding good \( t_r \)-uplets can be computationally expensive, we provide next some guidelines to accelerate the search procedure of the primitive set of rows:

- \( d_{\text{min}} \) is the minimum number of columns of \( H_1 \) that are dependent, thus the minimum distance of a \( t_r \)-uplet is at most the minimum distance associated with any two sub-matrices \( H_{ij} \) and \( H_{ij} \) of \( H_1 \). The minimum distance associated with these two elements is greater or equal to 2. Whenever possible (i.e. when we consider a sufficiently high order compared to the \( t_r \)-uplet size we try to optimize), we focus on the \( t_r \)-uplets having a minimum distance greater or equal to 3.

- Since the rows of \( H \) can be some permutations or multiplication by a constant of \( t_r \)-uplets of the primitive set, each and every element of this set can be written as an ordered set with the following structure

\[
\alpha^0 \ldots \alpha^j \ldots \alpha^k, \ 0 < i < \ldots < j \ldots < k
\]

- Based on the previous remarks, the \( t_r \)-uplets can be derived from the \((t_r - 1)\)-uplets by adding an element \( \alpha^0 \) such as

\[
\alpha^0 \ldots \alpha^j \ldots \alpha^k, \ 0 < i < \ldots < j \ldots < k < l
\]

- Once we have computed the primitive set, as seen in the following example, it can be further reduced since some \( t_r \)-uplets can be related by a multiplication by a constant. Note that, for a field of order \( p \) and a given \( t_r \), the equivalent binary matrix \( H_1 \) has parameters \( (N = pt_r, K = p(t_r - 1)) \). Using the tables given in [2], we obtain an upper bound on the minimum distance.
2) Minimum distance properties of rows: In Figure 1, we compare the minimum distance properties of non-binary LDPC codes with \( H_{TG} \) being designed both randomly or using the PEG algorithm. We also give, as a reference, some binary minimum distances for codes obtained using the optimization method proposed in [17]. We assume the knowledge of the structure \( H_{TG} \) (randomly designed or optimized using instances of the PEG algorithm [12] or other good construction algorithms [20]).

### IV. ALGEBRAIC PROPERTIES OF CYCLES AND STOPPING SETS

We denote \( H_b \) the equivalent binary matrix of \( H \) in \( GF(2) \); \( H_b \) is obtained by replacing all elements in \( H \) by their \( p \times p \) binary matrix representation as described in Section II. We note \( N_0 = N_p \) the binary codeword length. Since the equivalent binary minimum distance is related to the minimum number of independent columns of \( H_b \), it is also strongly related to the topology of the Tanner graph associated with the matrix \( H \), noted \( H_{TG} \). \( H_{TG} \) represents the edge connection topology of the Tanner graph associated with \( H \); its matrix representation is therefore the matrix \( H \) with all the non-zero entries set to 1.

As in the binary case, it is likely that the cycles and the stopping sets in the graph \( H_{TG} \) remain key elements to lower the error floor on the Frame Error Rate (FER) performance of the code. Hence, we first analyze the equivalent binary representations of both the cycles and the stopping sets, induced by \( H_{TG} \) in \( H \), in order to link their algebraic properties to a “local” minimum distance. Then, we derive some bounds for the binary minimum distance of \( (2, t_r, N) \).

![Fig. 1. Minimum distance achievable for rows: (*) using the binary image, (□) Upper bound from [2].](image-url)
For a \((2, t_r, N)\)-regular graph, a stopping set that is not reduced to a single stopping set is composed of at least 3 imbricated cycles. Let \(d_s\) be the symbol weight of the pseudo codeword associated with a given stopping set. For a \((2, t_r, N)\)-regular graph, the minimum symbol weight of a stopping set is \(d_{s,\text{min}} = \lceil 3g/4 \rceil\), where \(g\) is the girth of the graph \(H_{TG}\). For a stopping set with symbol weight \(d_s \geq d_{s,\text{min}}\), the equivalent binary matrix is no longer a square matrix: its binary representation is at most \((d_s-1)p \times d_s p\) rectangular matrix \(H_{ss}\) with an associated minimum distance depending on the coefficients arrangements. As a result, each codeword associated with \(H_{ss}\) is a codeword of the code defined by \(H\). Thus, structurally, the code performance is drastically limited by the smallest stopping sets and their associated minimum distance.

Unfortunately, unlike for cycles, there is no way to "cancel" their influence by proper symbol assignment: since each stopping set has a minimum distance associated with it, the only way to ensure a good minimum distance for the whole code is to try to maximize the minimum distance over the stopping set ensemble. It is also important to note that the cycle cancellation for the smallest cycles is an important prerequisite to avoid "catastrophic" or badly conditioned stopping sets: the stopping sets contain cycles, and therefore ensuring cycle cancellation inherently avoids that some columns of the equivalent binary parity matrix in a stopping sets add to zero, which results in a possible loss of rank. Note that for graph \(H_{TG}\) with minimum column degree \(t_c = 2\), it is quite simple to identify the set of stopping sets with minimum weight \(d_{s,\text{min}}\); this could be done in conjunction with the PEG construction by adding a procedure which tests if a group of nodes contains 3 imbricated cycles.

### C. Achievable minimum distance

In this section, we study the achievable minimum distance for the binary image of non-binary \((2, t_r, N)\) regular LDPC codes. We first derive some bounds on the achievable minimum distance when we consider random and PEG Tanner graphs. Then we compare the bounds we obtained with the minimum distances of the matrices that we optimized in [17].

We suppose that the cycle cancellation has been efficiently done (i.e. the cycles are cancelled for sufficiently long lengths), involving that no low weight codewords are produced by cycles. In this context, low weight codewords are given by stopping sets.

For a given girth \(g\) of \(H_{TG}\), the minimum stopping set weight is lower bounded by

\[
d_{s,\text{min}} = \lceil 3g/4 \rceil
\]

Then, the matrix \(H_{ss}\) associated with the stopping sets with weight \(d_{s,\text{min}}\) has dimensions at most \((M_{ss} = (d_{s,\text{min}}-1)p, N_{ss} = d_{s,\text{min}} p)\). Using the maximum achievable minimum distance given by [2] for a code with the preceding parameters \((M_{ss}, N_{ss})\), we can obtain an upper bound on the maximum achievable binary minimum distance for that minimal stopping set with weight \(d_{s,\text{min}}\). Note that this resulting binary minimum distance is not an upper bound on any Tanner graph of girth \(g\) since (3) is not an upper bound.

Next, we aim to link the binary minimum distance \(d_{\text{min}}\) with the length \(N_b\) of the code associated with \(H_b\). For a \((t_s, t_r)\)-regular Tanner graph, an upper bound (UB) on the girth of the graph as a function of \(N\) has been derived in Lemma 3 [12]. Applying this results in the \((2, t_r, N)\) case, we can derive an UB on \(d_{s,\text{min}}\): 

**Lemma 1:** Let \(H_{TG}\) be a \((2, t_r, N)\) regular Tanner graph. The minimum stopping set weight \(d_{s,\text{min}}\) is upper bounded by

\[
d_{s,\text{min}} \leq \min(d_1, d_2)
\]

where

\[
d_1 = \begin{cases} 3\lceil t_1 \rceil + 2 & \text{if } \mathcal{I}_1 = 0 \\ 3\lceil t_1 \rceil + 3 & \text{otherwise} \end{cases}
\]

\[
d_2 = \begin{cases} 3\lceil t_2 \rceil + 2 & \text{if } \mathcal{I}_2 = 0 \\ 3\lceil t_2 \rceil + 3 & \text{otherwise} \end{cases}
\]

in which

\[
t_1 = \frac{\log ((M-1)(1-\frac{2}{t_r}) + 1)}{\log (t_r - 1)}
\]

\[
t_2 = \frac{\log ((N-1)(1-\frac{t_r}{2(t_r-1)}) + 1)}{\log (t_r - 1)}
\]

and \(\mathcal{I}_1\) is equal to 0 if and only if

\[
(t_r - 1)^{|t_1|} > M - 1 - \frac{t_r((t_r - 1)^{|t_1|} - 1)}{t_r - 2}
\]

and \(\mathcal{I}_2\) is equal to 0 if and only if

\[
(t_r - 1)^{|t_2|} > N - 1 - \frac{2(t_r - 1)((t_r - 1)^{|t_2|} - 1)}{t_r - 2}
\]

Note that, according to this lemma, \(d_{s,\text{min}}\) varies in \(O(\log (N))\). From [2], we can compute numerically a bound on \(d_{s,\text{min}}\) versus \(N_b = pN\) for a regular Tanner graph. But, since the UB in [2] for a code with parameters \((M_{ss}, N_{ss})\) does not provide an analytical behavior of \(d_{s,\text{min}}\) as a function of \(d_{s,\text{min}}\), we need to apply the Elias upper bound for the code with parameters \((M_{ss} = (d_{s,\text{min}}-1)p, N_{ss} = d_{s,\text{min}} p)\) [18]:

\[
d_{\text{min}} \leq 2A(1 - A)d_{s,\text{min}} p
\]

with \(A\) solution of

\[
1/d_{s,\text{min}} = 1 + A \log_2 (A) + (1-A) \log_2 (1-A), \quad 0 \leq A \leq 1/2
\]

Reporting (4)-(6) into (7), we can conclude that \(d_{s,\text{min}}\) scales as \(O(\log (N_b)) = O(\log (N))\). This can be related to a previous result from [8], where it is shown that the minimum distance of the binary \((2, t_r, N)\) regular LDPC codes can increase at most logarithmically with the codeword length \(N\). This behavior indicates that the non-binary \((2, t_r, N)\) regular LDPC codes exhibit poor minimum distances properties, that emphasizes the need for efficient methods to build codes with good minimum distance properties.

When considering regular PEG Tanner graphs, in order to derive a bound, we use the best optimized \((2, t_r, N)\) matrices \(H_{TG}\) obtained with a PEG construction, and we compute
the effective achievable minimum stopping set weight. Then, as above, we can derive a bound on the minimum distance achievable under PEG Tanner graph construction using the minimum weight stopping sets and the upper bound [2].

In Figures 2 and 3, we report the binary minimum distance we have computed for some codes optimized using the method in [17](Opt. codes) curve) and we present some bounds for (2, 4) LDPC codes with $GF(2^{56})$ and $GF(64)$: (Bound-random) is the bound derived from Lemma 1 and upper bound [2] and (UB-Opt codes) is an upper bound derived from the effective $d_{s,\text{min}}$ of optimized codes in [17] and the upper bound [2]. We observe that, in both cases, we have $d_{\text{min}} = O(\log(N))$. Note that in our PEG constructions, the effective minimal stopping set weight was either $d_{s,\text{min}}$ or $d_{s,\text{min}} + 1$, explaining why the PEG bound can be higher than the random graph upper bound based on minimal stopping set with weight strictly equal to $d_{s,\text{min}}$.

V. CONCLUSION

In this paper, we developed algebraic properties of regular $(2, t_r)$ non binary LDPC codes designed using their binary image. We characterized the algebraic properties of optimized rows, cycles and stopping sets. This analysis allowed us to show that the equivalent binary minimum distance of the non binary code associated with $H$ asymptotically increases with $\log(N)$, exhibiting poor minimum distance properties for that kind of codes, and emphasizing the need for efficient methods to construct codes with good minimum distance properties as done in [21][13][17].

REFERENCES