Analysis and Design of Ultra-Sparse Non-Binary Cluster-LDPC Codes

David DECLERCQ
ETIS, ENSEA / Univ. Cergy-Pontoise / CNRS UMR-8051
F-95000, Cergy-Pontoise, France
declercq@ensea.fr

Valentin SAVIN
CEA-LETI, MINATEC campus
F-38054, Grenoble, France
valentin.savin@cea.fr

Lam PHAM SY
EUTELSAT SA
F-75015, Paris, France
lphamsy@eutelsat.fr

***Abstract*—This paper continues a previous work on non-binary cluster-LDPC codes. Such codes are defined by locally dense parity-check matrices, with (possibly rectangular) dense clusters of bits, but which are cluster-wise sparse. We derive a lower bound on the minimum distance of non-binary cluster-LDPC codes that is based on the topological properties of the underlying bipartite graph. We also propose an optimization procedure, which allows designing finite length codes with large minimum distance, even in the extreme case of codes defined by *ultra-sparse* graphs, *i.e.* graphs with all symbol-nodes of degree $d_v = 2$. Furthermore, we provide asymptotic thresholds of ensembles of non-binary cluster-LDPC codes, which are computed exactly under the Belief Propagation decoding, and upper-bounded under the Maximum a Posteriori (MAP) decoding. We show that the MAP-threshold upper bounds, which are conjunctured to be tight, quickly approach the channel capacity, which confirms the excellent minimal distance properties of non-binary cluster-LDPC codes.**

I. Introduction

In this paper we consider a very general construction of non-binary low-density codes, known as *non-binary cluster-LDPC* (NB-cluster-LDPC) codes [1]. The sparse parity check matrix H is organized in clusters of bits, where each cluster is either a full-rank or a zero $p \times r$ binary matrix determined by p consecutive rows and r consecutive columns of H. The case $p = r$ corresponds to the family of non-binary LDPC codes defined over general linear groups [2], while the case $p < r$ corresponds to the more general codes studied in this paper. This type of codes has been first introduced in [3], in the case of general ensemble of irregular graphs, and can be seen as a non-binary generalization of the GLD codes proposed in [4].

The case of codes for which the underlying bipartite graph is *ultra-sparse*, meaning that each symbol node is connected to exactly $d_v = 2$ linear-constraint nodes, is of particular interest. First, very large girths can be obtained for Tanner graphs with $d_v = 2$, as demonstrated in [5], [6]. It has also been pointed out [3], [7] that when the size of the non-binary alphabet grows, the best decoding thresholds are obtained for average density of edges closer and closer to $d_v = 2$. Practically, for NB codes defined over finite fields $\mathbb{F}_q$ with $q \geq 64$, the best codes, both asymptotically and at finite lengths, are *ultra-sparse* codes. Despite those advantages, ultra-sparse LDPC codes over finite fields or general linear groups suffer from a serious drawback, as their minimum distance is limited and grows at best as $\mathcal{O}(\log(N))$, where N is the number of symbol nodes [8].

This work was supported by the French ANR grant no. 2009 VERS 019 04 (ARSSO project) and the FP7-ICT NoE NEWCOM++

In [1], the authors showed that the new class of NB-cluster-LDPC codes does not suffer from the same limitations. Precisely, it has been proven that the minimum distance of ultra-sparse cluster codes grows linearly with the code length, provided that the minimum distance of the *component code*, which defines the generalized parity-checks, is above the second eigenvalue of the graph. Using the Gilbert-Varshamov bound on the minimum distance and the Friedman theorem [9] on the second eigenvalue of regular graphs, it has been shown that there exist ensembles of ultra-sparse cluster codes with linear growing minimum distance. However, for these ensembles the size of clusters could be far too large for practical applications.

This work continues the research begun in our previous work [1]. We derive a new lower bound on the minimum distance of cluster-codes that is based on the topological, rather than algebraic, properties of the underlying bipartite graph. Although the bound we propose is sublinear in the code length, it has the advantage of being applicable to any code length and cluster sizes, and it also allows deriving a methodology for the design of finite length cluster codes. Since the asymptotic growth of the minimum distance determines the asymptotic performance of the Maximum A Posteriori (MAP) decoding over the Binary Erasure Channel (BEC), we also present an analysis of the asymptotic performance of cluster codes in terms of EXIT curves. By using the area theorem [10], we derive an upper bound on the MAP threshold, which is conjunctured to be tight in quite general settings, especially for codes defined by regular graphs. The results we obtain tend to confirm that the growth rate of the minimum distance of designed cluster codes is actually linear in the code length. Finally, we propose an optimization procedure for finite length codes, which allows designing NB-cluster-LDPC codes with large minimum distance, while preserving the constraint of ultra-sparse Tanner graphs. The performance of the designed codes is assessed by Monte-Carlo simulation, which demonstrates the excellent performance of our codes under iterative Belief-Propagation decoding, on both the waterfall and the error-floor region.

II. Non-binary cluster-LDPC codes

A NB-cluster-LDPC code is defined by the structure of its binary parity-check matrix H_{bin}, which is organized in dense clusters of bits, but which is globally cluster-wise sparse. We consider two sequences of integers $\{p_i\}_{i=1,\ldots,M}$

and $\{r_j\}_{j=1,\ldots,N}$ that define the clustering of the parity-check matrix $H_{\text{bin}} = (h_{i,j})_{\substack{i=1,\ldots,M \\ j=1,\ldots,N}}$. Each cluster $h_{i,j}$ is a binary matrix of size $p_i \times r_j$, which is either all-zero or full-rank. We further assume that for any full-rank cluster $h_{i,j}$, we have $r_j \leq p_i$. We note that the binary parity check matrix H_{bin} is of size $M_{\text{bin}} \times N_{\text{bin}}$, where $M_{\text{bin}} = \sum_{i=1}^{M} p_i$ and $N_{\text{bin}} = \sum_{j=1}^{N} r_j$. The parameter N_{bin} will be referred to as the *binary length* of the NB-cluster-LDPC code.

A codeword $X = (X_1, \ldots, X_N)$ is a sequence of binary vectors $X_j \in \mathbb{F}_2^{r_j}$, verifying the linear system $\left\{\sum_{j=1}^{N} h_{i,j} X_j = 0, \forall i = 1, \ldots, M\right\}$. We shall also refer to X_j as the jth coded symbol. From a non-binary perspective, the ith non-binary linear constraint of the above system corresponds to p_i binary parity-checks, while the jth non-binary coded symbol X_j corresponds to r_j coded bits. Under these settings, each non-zero cluster gives an injective linear transformation $h_{i,j} : \mathbb{F}_2^{r_j} \rightarrow \mathbb{F}_2^{p_i}$, from the alphabet of the jth coded symbol to the vector-space of the ith non-binary constraint.

Let $\{h_{i,j_k}\}_{k=1,\ldots,d}$ be the non-zero clusters in the ith non-binary linear constraint. They form a binary code $\mathcal{C}_i$ with length $\sum_{k=1}^{d} r_{j_k}$ bits and co-dimension p_i. Throughout the paper, this binary code will be referred to as the ith *component code*; hence, $\mathcal{C}_i$ is the kernel of the binary matrix H_i^{c}:

$$H_i^{\text{c}} = [h_{i,j_1} \; h_{i,j_2} \; \cdots \; h_{i,j_d}]$$

Note also that two component matrices $H_{i_1}^{\text{c}}$ and $H_{i_2}^{\text{c}}$ might be equal, possibly after some column or row permutation, even if the number and the size of non-zero clusters participating in the non-binary constraints i_1 and i_2 are different. Put differently, the same component matrix $H_{i_1}^{\text{c}} = H_{i_2}^{\text{c}}$ can be clusterized in different manners.

The non-binary nature of NB-cluster-LDPC codes is better highlighted through their graphical representation. The bipartite (Tanner) graph Γ, associated with the NB-cluster-LDPC code, consists of N *symbol-nodes* and M *constraint nodes*, which are connected according to the non-zero clusters $h_{i,j}$. The jth symbol-node represents a plurality of r_j coded bits, while the ith constraint-node represents a plurality of p_i binary parity-checks. The nodes are connected by an edge if and only if the cluster $h_{i,j}$ is non-zero, in which case the corresponding edge is considered as being *labeled* by $h_{i,j}$. The labels on the edges correspond to the generalization of the non-zero field values for usual NB-LDPC codes [11].

The Belief Propagation (BP) decoder for NB-cluster-LDPC codes is similar to the one for usual NB-LDPC codes on Galois fields [7], [12], and is based on local Bayesian update rules following the non-binary constraints structure. As a consequence, BP decoding can be performed in the Fourier domain to speed-up the simulations [11], and the low complexity decoders proposed for NB-LDPC codes [13], [14] can be easily extended to NB-cluster-LDPC codes.

III. Minimum distance of NB-cluster-LDPC codes

This section deals with the minimum distance properties of NB-cluster-LDPC codes. We only consider NB-cluster-LDPC codes defined by regular $(d_v = 2, d_c)$ graphs, which represents the most problematic case in terms of minimum distance. We use the notation from the previous section, and we also denote by $\Im(h_{i,j})$ the *image* of the linear application:

$$h_{i,j} : \mathbb{F}_2^{r_j} \rightarrow \mathbb{F}_2^{p_i}$$

Since $h_{i,j}$ is injective, $\Im(h_{i,j}) \subseteq \mathbb{F}_2^{p_i}$ is a vector subspace of dimension r_j.

The following theorem provides a lower bound on the minimum distance Δ of the NB-cluster-LDPC code, depending on the girth g of its Tanner graph, namely the length of its shortest cycle.

Theorem 1: Assume that $\Im(h_{i,j'}) \cap \Im(h_{i,j''}) = \{0\}$ for any two edges $i \sim j'$ and $i \sim j''$, adjacent to the same constraint node i. Then:

$$\Delta \geq 2^{\lfloor\!\lfloor \frac{g}{4} \rfloor\!\rfloor + 2} - 3,$$

where g is the girth of the bipartite graph Γ, and $\lfloor\!\lfloor x \rfloor\!\rfloor$ denotes the largest integer strictly less than x (thus, $\lfloor\!\lfloor x \rfloor\!\rfloor = x - 1$ if x is integer, and $\lfloor\!\lfloor x \rfloor\!\rfloor = \lfloor x \rfloor$ otherwise).

Proof. Let $X = (X_1, \ldots, X_N)$ be a non-zero codeword, where $X_j = (x_{j,1}, \ldots, x_{j,r_j}) \in \mathbb{F}_2^{r_j}$. Let j_0 be an *active symbol node*, meaning that X_{j_0} contains a non-zero bit, and let $\mathcal{V}_\ell$ be the set of nodes at topological distance ℓ from j_0. Thus, $\mathcal{V}_0 = \{j_0\}$, $\mathcal{V}_1$ is the set of constraint-nodes connected to j_0, $\mathcal{V}_2$ is the set of symbol-nodes connected to j_0 by a path of length 2, etc. Assume that ℓ is odd and $\ell < g/2$. Then any constraint node $i \in \mathcal{V}_\ell$ is connected to a unique symbol node belonging to $\mathcal{V}_{\ell-1}$ and to $d_c - 1$ symbol nodes belonging to $\mathcal{V}_{\ell+1}$. Consider $i \in \mathcal{V}_\ell$, such that its parent symbol node $j \in \mathcal{V}_{\ell-1}$ is active (*i.e.* X_j contains a non-zero bit). Since X verifies the linear constraint corresponding to i, we have:

$$h_{i,j} X_j + \sum_{j \neq j' \sim i} h_{i,j'} X_{j'} = 0$$

On the other hand, $\Im(h_{i,j}) \cap \Im(h_{i,j'}) = \{0\}$, for any $j' \neq j$. From the above equation, it follows that at least two children symbol nodes $(j', j'') \in \mathcal{V}_{\ell+1}$ have to be active. Since there are no cycles of length $< g$, the minimum distance is lower bounded by the number of active-symbol nodes at distance $\ell < g/2$, with ℓ even, which is given by:

$$1 + \sum_{\substack{2 \leq \ell < g/2 \\ \ell \text{ even}}} 2^{\frac{\ell}{2}-1} \times 4 = 1 + 4 \sum_{1 \leq t < g/4} 2^{t-1} = 2^{\lfloor\!\lfloor \frac{g}{4} \rfloor\!\rfloor + 2} - 3$$

□

Corollary 2: Under the assumptions of the above theorem, if the bipartite graph Γ is constructed by using the Progressive Edge Growth [15] (PEG) algorithm, then:

$$\Delta \geq 2 \left(\frac{d_c - 2}{d_c - 1} N + 1 \right)^{\log_{(d_c-1)}(\sqrt{2})} - 3$$

Proof. Follows from the above theorem and from [15, Theorem1] that states a lower bound on the girth of regular bipartite graphs built by the PEG algorithm. □

The Theorem 1 provides a lower bound on the minimum distance of NB-cluster-LDPC codes, under the assumption that

Table I

SET OF 16 PAIRWISE DISJOINT CLUSTERS OF SIZE $(p = 8, r = 4)$ OBTAINED FROM THE EXTENDED-HAMMING CODE [128, 120, 4].

1010	1101	0011	0001	1101	0000	1100	1110	0111	0001	0111	1111	1110	0110	1000	1001
0000	1000	0100	0100	1000	1001	1100	1010	1000	1100	1111	0111	0010	1110	0101	1010
1011	1100	0110	1001	0100	0111	1111	0100	1011	0100	0000	0001	1000	1010	0101	1101
1010	0100	1110	0111	0000	0111	1001	1101	1110	1111	0001	0111	0101	0010	1010	0110
1011	1000	0001	0101	0010	1011	1001	1000	0111	0111	0100	0011	1011	0101	0011	1100
0100	0101	0101	1001	1100	1100	0010	1001	0111	1111	1010	0000	0010	1101	0010	1110
1101	0100	1101	1001	0001	0110	1000	0110	1101	1100	0010	0100	0111	1000	0111	1001
1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111	1111

the labels $h_{i,j}$ of the edges incident to a constraint node $i = 1, \ldots, M$ have pairwise disjoint images. If such an assumption is met, then for each constraint node i, the images of the labels of the incident edges $j \sim i$ define d_c vector subspaces $\{\Im(h_{i,j})\}_{j\sim i}$ of $\mathbb{F}_2^{p_i}$, of dimensions $\{r_j\}_{j\sim i}$, which are pairwise disjoint. Conversely, if for each constraint node i there exist d_c pairwise disjoint vector subspaces of $\mathbb{F}_2^{p_i}$, of dimensions $\{r_j\}_{j\sim i}$, then it is possible to label the edges of Γ, so that the assumption of Theorem 1 is met.

In order to verify the efficiency of this constraint, and then have an indication whether the lower bound of Theorem 1 is tight or not, let us denote by $d(r, p)$ the maximum number of pairwise disjoint vector subspaces of $\mathbb{F}_2^p$, for some given cluster dimension $r \leq p$. Clearly, $d(1, p) = 2^p - 1$ and $d(r, p) = 1$ for any $\left\lfloor \frac{p}{2} \right\rfloor < r \leq p$. Moreover, for any $1 \leq r \leq p$, the following inequalities hold:

$$\left\lceil \frac{2^{p-r+1} - 1}{2^r - 1} \right\rceil \leq d(r, p) \leq \left\lfloor \frac{2^p - 1}{2^r - 1} \right\rfloor \tag{1}$$

The proof is omitted due to space limitation. Now, consider a NB-cluster-LDPC code, and for each constraint node i, let $\bar{r}_i = \max_{j\sim i} r_j \leq p_i$ be the maximum dimension of its neighbor symbol nodes. If $d_c \leq d(\bar{r}_i, p_i), \forall i$, then for each constraint node i there exist d_c pairwise disjoint vector subspaces of $\mathbb{F}_2^{p_i}$, of dimension $\bar{r}_i$. Since $\bar{r}_i \geq r_j, \forall j \sim i$, it follows that it is possible to label the edges of Γ, so that the assumption of Theorem 1 is met, and provides a useful tool for NB-cluster-LDPC code design with good minimum distance. As an example, we give in Table I a set of 16 clusters of size $(p = 8, r = 4)$ with pairwise disjoint images, obtained by selecting columns from the $(N_{\text{bin}} = 128, K_{\text{bin}} = 120)$ extended-Hamming code with minimum distance 4. Note that the maximum number of clusters indicated by equation (1) is $d(4, 8) \leq 17$.

IV. BP AND MAP THRESHOLDS OVER THE BEC

Throughout this section, we assume that the transmission takes place over a Binary Erasure Channel (BEC). We consider the ensemble of cluster codes $\mathcal{E}(\lambda, \rho, r, p)$, where λ and ρ denote the edge-perspective degree distribution polynomials of the subjacent bipartite graph, each graph edge corresponding to a full-rank cluster of size $p \times r$ $(p \geq r)$ in the binary parity-check matrix of the code.

EXtrinsic Information Transfer (EXIT) curves were first introduced in [16] as a technique to analyse the convergence of the iterative decoding process of parallel concatenated codes. A more general definition has been introduced in [10], where the EXIT curve is associated with the sparse graph system rather than the component (concatenated) codes. The advantage of this approach is that it allows defining EXIT curves for any decoding algorithm (e.g. IT or MAP decoding, the last one being equivalent to the Maximum Likelihood decoding over the BEC).

For given code length N_{bin} and channel erasure probability ε, we denote by $h_{N_{\text{bin}}}(\varepsilon)$ the probability that a bit remains erased after decoding, assuming that the received value (if any) of this particular bit has not been submitted to the decoder. The EXIT function $h(\varepsilon)$ is defined as the (pointwise) limit of $h_{N_{\text{bin}}}(\varepsilon)$, when the code length N_{bin} goes to infinity. The BP and MAP thresholds of $\mathcal{E}(\lambda, \rho, r, p)$ are given by:

$$\begin{aligned} \epsilon_{\text{BP}} &= \sup\{\varepsilon \in [0, 1] \mid h_{\text{BP}}(\varepsilon) = 0\}, \\ \epsilon_{\text{MAP}} &= \sup\{\varepsilon \in [0, 1] \mid h_{\text{MAP}}(\varepsilon) = 0\}, \end{aligned}$$

where subscripts BP and MAP are used to indicate the decoding algorithm.

Explicit formulae for the computation of the BP-EXIT curve can be derived by using density evolution (DE) techniques, similar to those presented in [17]. We have derived DE equations for NB-cluster-LDPC codes, however, due to space limitation they are not presented here, but will be included in an extended version of this paper. The exact computation of the MAP-EXIT function is generally unworkable. However, using the area theorem [10], we have $\int_{\epsilon_{\text{MAP}}}^{1} h_{\text{MAP}}(\varepsilon)\, d\varepsilon = R$, where R is the designed coding rate of the given ensemble. Moreover, since the BP decoding is suboptimal with respect to the MAP decoding, we have $h_{\text{BP}}(\varepsilon) \geq h_{\text{MAP}}(\varepsilon)$. Hence, if for some $\bar{\epsilon}_{\text{MAP}}$, $\int_{\bar{\epsilon}_{\text{MAP}}}^{1} h_{\text{BP}}(\varepsilon)\, d\varepsilon = R$, we necessarily have $\bar{\epsilon}_{\text{MAP}} \geq \epsilon_{\text{MAP}}$. This gives an upper bound on the MAP-threshold, which is known to be tight in the binary case [10].

Table II shows the BP thresholds and the MAP threshold upper bounds for various ensembles of ultra-sparse cluster codes with designed code rate $R = 1/2$. Parameters r and p are shown in the table, $\lambda = x$ (corresponding to $d_v = 2$), while ρ is determined, "as regular as possible", according to parameters λ, r, p, and R. We can observe that for both $r = 1$ and $r = 2$, the BP threshold value increases with p for $r \leq p \leq 4$, and it starts decreasing from this point. This can be explained as follows. By increasing the value of p, we also increase the check-node degree ($d_v = 2$ and $R = 1/2$, hence $d_c = 4\frac{p}{r}$). Consequently, component codes become longer and the expected value of the number of erased bits on each component code increases. If the expected value of the number of erased bits on component codes increases faster than their minimal distance, the iterative decoding is penalized, due to the fact that each component code is affected by too many erased bits. On the other hand, the performance of the MAP decoding depends on the minimal distance of the cluster code. As we can observe, the upper bound of the MAP threshold

Table II
BP THRESHOLDS AND THE MAP-THRESHOLD UPPER BOUNDS FOR ENSEMBLES $\mathcal{E}(\lambda = x, \rho, r, p)$, WITH DESIGNED CODE RATE $R = 1/2$.

	p	1	2	3	4	5	6	7	8	9	10	16
$r=1$	ϵ_{BP}	0.3283	0.4092	0.4442	0.4497	0.4445	0.4356	0.4262	0.4171	0.4089	0.4016	0.3715
	$\bar{\epsilon}_{\text{MAP}}$	0.3283	0.4120	0.4585	0.4795	0.4896	0.4946	0.4971	0.4984	0.4991	0.4995	0.49998
$r=2$	ϵ_{BP}	N/A	0.4092	0.4489	0.4572	0.4527	0.4438	0.4338	0.4241	0.4153	0.4074	0.3752
	$\bar{\epsilon}_{\text{MAP}}$		0.4092	0.4541	0.4779	0.4889	0.4942	0.4969	0.4983	0.4991	0.4995	0.49998

increases with p (and quickly approaches the Shannon limit), which suggests that the minimum distance of cluster codes is improved by increasing p.

V. DESIGN OF FINITE LENGTH NB-CLUSTER-LDPC CODES

We propose in this section an optimization procedure which designs NB-cluster-LDPC codes with large minimum distance, while preserving the constraint of ultra-sparse Tanner graphs with $d_v = 2$. Let us assume that one wants to design a NB-cluster-LDPC code with a target codeword length N^*_{bin} and a target coding rate R^*. We also assume that the clusters have constant number of rows $(p^*_1, \ldots, p^*_M) = (p^*, \ldots, p^*)$, while their number of columns might be different $(r^*_1, \ldots, r^*_N)$, and not fixed *a priori*, as long as the constraint on the codeword length $\sum_i r^*_i = N^*_{\text{bin}}$ is verified.

As indicated by [1, Theorem 1], good NB-cluster-LDPC codes should have bipartite graph with the smallest possible second eigenvalue $\lambda_2(\Gamma)$, and component codes $\mathcal{C}_i$ with the largest possible minimum distance δ. Following [1], we refer to δ as the local minimum distance of the NB-cluster-LDPC code. One possible way of designing a NB-cluster-LDPC code with a given local minimum distance δ is by considering a linear code $\mathcal{C}$ of length d, dimension $(d-p)$, and minimum distance δ, and labeling the edges incident to each constraint node of Γ by an appropriate clustering of a parity-check matrix H^{c} of $\mathcal{C}$. In order to differentiate the structure of the clusters from one constraint node to another, a simple procedure consists in performing a random permutation of the columns of H^{c}, with the constraint that the clusters are full rank, which is ensured if $\delta \geq \min_i (r^*_i)$.

We moreover advise to follow the guidelines of Theorem 1 and form the clusters such that they have pairwise disjoint images. This constraint, together with the minimization of $\lambda_2(\Gamma)$ and the maximization of δ, form the main guidelines of the first step of the proposed optimized code design algorithm. The principle of the optimization is to start from a NB-cluster-LDPC with larger length $N^{(0)}_{\text{bin}} > N^*_{\text{bin}}$ and larger rate $R^{(0)} > R^*$, but with the same co-dimension $M^{(0)}_{\text{bin}} = M^*_{\text{bin}}$, than the ones of the target code. The first step of optimization is then to build $H^{(0)}_{\text{bin}}$ with the above mentioned guidelines. Then the construction algorithm performs an iterative shortening of the code by removing columns of the binary parity-check matrix, with the goal of increasing monotonically the minimum distance at each iteration of shortening. At each iteration k, we make use of the improved impulse algorithm [18] to identify low Hamming weight codewords of $H^{(k)}_{\text{bin}}$, and *kill them* in order to ensure by this careful design that the obtained NB-cluster-LDPC has a large minimum distance. We proceed in an iterative fashion as summarized in Algorithm 1 described below.

Algorithm 1 NB-cluster-LDPC code Optimization

[Initialization]: design of $H^{(0)}_{\text{bin}}$ with constant cluster size $\left(p^{(0)} = p^*, r^{(0)}\right)$

(a) *Initialization-step2:* build a Tanner graph Γ with $(d_v, d_c) = (2, d_c)$ such that $R^{(0)} = 1 - d_v/d_c > R^*$, and $N^{(0)}$ variable nodes. The graph should be built with the largest possible girth, minimum number of shortest cycles, and minimum value of its second eigenvalue [1],

(b) *Initialization-step2:* choose a component code H^{c} of size $p^* \times d_c * r^{(0)}$ and with the largest possible local minimum-distance d,

(c) *Initialization-step3:* for each NB-contraint node of the Tanner graph, randomly permute the columns of H^{c}, and assign to the d_c edges clusters of size $p^* \times r^{(0)}$ with $r^{(0)}$ consecutive columns of the permuted component code. Consider only permutations such that the clusters are full rank, and have pairwise disjoint images. This initialization ensures that all the component codes of the initialized NB-cluster-LDPC code are identical and with the same minimum distance.

[Iterative Shortening]:

(1) Start with $k = 0$,

(2) Run the impulse algorithm [18] to build a set $\mathcal{S}^{(k)}$ of low weight codewords of $H^{(k)}$ with Hamming weights $\in \{\Delta^{(k)}, \ldots \Delta^{(k)} + \varepsilon\}$,

(3) Select n_k columns from $H^{(k)}_{\text{bin}}$ which participate the most in the low-weight codewords $\mathcal{S}^{(k)}$,

(4) Shorten the code by pruning the n_k columns, in order to obtain a new code $H^{(k+1)}_{\text{bin}}$ with codeword length $N^{(k+1)}_{\text{bin}} = N^{(k)}_{\text{bin}} - n_k$,

(5) $k = k+1$, goto (2) until the target length is reached $N^{(k+1)}_{\text{bin}} = N^*_{\text{bin}}$.

Let us discuss first the proposed initialization steps. For the step (a) we construct several graphs Γ by using the modified Random-PEG algorithm [6], which is able to design minimum size graphs for $d_v = 2$ and a given girth g. We then choose a graph Γ with minimum second eigenvalue. For the step (b), we choose a component code H^{c} with the highest possible local minimum distance from existing codes in the literature [19]. Finally, the step (c) consists in selecting columns of H^{c} such as to build clusters with pairwise disjoint images. An efficient implementation of this step is to select at random the clusters from a pre-determined set of clusters with pairwise disjoint images, as the set given on Table I for clusters of sizes $(p = 8, r = 4)$.

We now discuss the details of the iterative shortening procedure, which relies heavily on the ability to identify low-weight codewords of a sparse linear code. We have adapted the "improved impulse method" algorithm proposed in [18] to NB-cluster-LDPC codes. This algorithm finds the low-weight codewords of sparse linear codes, and was proved to be largely more efficient than other techniques on benchmark LDPC codes. The iterative shortening process ensures that the final NB-cluster-LDPC code has no codeword of Hamming weights much smaller than its typical expected minimum distance. We give in Table III an example of the iterative shortening process, for the case of $N^*_{\text{bin}} = 256$ and $R^* = 1/2$. The initial code defined by the parity check matrix $H^{(0)}_{\text{bin}}$ is built from a $(d_v = 2, d_c = 16)$ graph with rate $R^{(0)} = 0.75$ and clusters of sizes $(p, r^{(0)}) = (8, 4)$ taken at random from the Table I. We indicate at each step the code rate and length of $H^{(k)}_{\text{bin}}$, as well as the low weight spectrum computed with the improved impulse method. The successive 7 steps of shortening gradually improve the low weight spectrum, as the code rate decreases

Table III
RESULTS OF SUCCESSIVE SHORTENINGS OF NB-CLUSTER-LDPC CODES

(k)	(0)		(1)		(2)		(3)		(4)		(5)		(6)		(7)	
$N_{\text{bin}}^{(k)}$	512		464		432		400		352		304		280		2506	
$R^{(k)}$	0.750		0.724		0.704		0.680		0.636		0.579		0.543		0.500	
weight	10	11	12	13	12	13	13	14	15	16	18	19	19	20	23	24
#codewords	2	12	11	125	2	32	5	42	3	43	10	53	3	19	82	253

from $R^{(0)} = 0.75$ to $R^{(7)} = R^* = 0.5$, reaching a final NB-cluster-LDPC code with minimum distance $\Delta = 23$, which is a good value considering that the NB-cluster-LDPC code is based on an ultra-sparse Tanner graph with girth $g = 8$. Note that a direct design with only the initialization step and no shortening procedure gave a NB-cluster-LDPC code with minimum distance $\Delta = 20$.

The optimization algorithm presented in this section should be used only to design NB-cluster-LDPC codes at short to moderate codeword lengths. For long codes the iterative shortening steps become cumbersome, and for $N_{\text{bin}}^* > 2000$ the NB-cluster-LDPC code design is limited at the initialization step of the optimization algorithm, and other refinement techniques need to be used without relying on the knowledge of low weight codewords.

VI. NUMERICAL RESULTS

In this section, the performance of NB-cluster-LDPC codes is assessed through Monte-Carlo simulation over the Binary-Input AWGN channel. In Figure 1, we show the comparison of regular binary LDPC codes from [20], and two NB-cluster-LDPC codes with $p^* = 8$, optimized with Algorithm 1. The BP decoding performs a maximum of 100 iterations and it stops when a valid codeword is found by syndrome computation. As expected, the NB-cluster-LDPC codes combine the two advantages of having good performance in the waterfall region due to the sparse representation of their Tanner graphs, as well as larger minimum distance than other classes of codes. Since it has been shown in [8] that the error floor of ultra-sparse non-binary LDPC codes with $d_v = 2$ comes from low-weight codewords and not from the presence of the so-called pseudo-codewords, the large minimum-distance of NB-cluster-LDPC codes also determines very good performance in the error floor region. NB-cluster-LDPC codes did not show any error floor until FER $= 10^{-8}$, which, coupled with the waterfall performance, is a very good and unpreceding result for iteratively decodable codes at short block-lengths.

VII. CONCLUSION

In this paper, we derived a lower bound on the minimum distance of ultra-sparse NB-cluster-LDPC codes, and provided asymptotic thresholds for the BP and MAP decoders. We also proposed a construction of finite-length codes with large minimum distances, and we showed that the constructed codes exhibit very good performance under BP decoding on both the waterfall and the error-floor region.

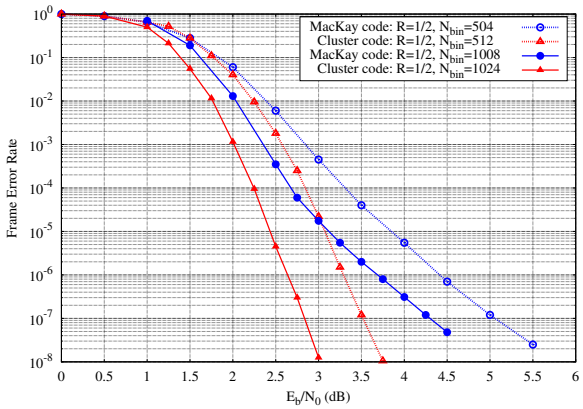

Figure 1. Performance comparison between ultra-sparse ($d_v = 2$) NB-cluster-LDPC codes and regular ($d_v = 3, d_c = 6$) binary LDPC codes.

REFERENCES

[1] V. Savin and D. Declercq, "Linear growing minimum distance of ultra-sparse non-binary cluster-LDPC codes," in *IEEE International Symposium on Information Theory (ISIT)*, 2011, pp. 523–527.

[2] V. Rathi and R. Urbanke, "Density evolution, thresholds and the stability condition for non-binary LDPC codes," *IEE Proceedings-Communications*, vol. 152, no. 6, pp. 1069–1074, Dec. 2005.

[3] L. Sassatelli and D. Declercq, "Non-binary hybrid LDPC codes," *in IEEE Trans. Info. Theo.*, vol. 56, no. 10, pp. 5314–5334, October 2010.

[4] J. J. Boutros, O. Pothier, and G. Zémor, "Generalized low density (Tanner) codes," in *IEEE Int. Conf. on Comm.*, Vancouver, June 1999.

[5] X.-Y. Hu, E. Eleftheriou, and D.M. Arnold, "Regular and irregular progressive edge-growth Tanner graphs," *IEEE Trans. on Inf. Theory*, vol. 51, no. 1, pp. 386–398, January 2005.

[6] A. Venkiah, D. Declercq, and C. Poulliat, "Design of cages with a randomized progressive edge growth algorithm," *IEEE Commun. Letters*, vol. 12, no. 4, pp. 301–303, April 2008.

[7] M.C. Davey and D. MacKay, "Low-density parity-check codes over GF(q)," *IEEE Commun. letters*, vol. 2, pp. 165–167, June 1998.

[8] C. Poulliat, M. Fossorier, and D. Declercq, "Design of regular $(2, d_c)$-LDPC codes over GF(q) using their binary images," *IEEE Trans. on Commun.*, vol. 56, no. 10, pp. 1626–1635, October 2008.

[9] J. Friedman, "A proof of Alon's second eigenvalue conjecture and related problems," *Memoirs of the American Mathematical Society*, vol. 195, pp. 100 pp., 2008.

[10] C. Measson, A. Montanari, and R. Urbanke, "Maxwell Construction: The Hidden Bridge Between Iterative and Maximum a Posteriori Decoding," *Information Theory, IEEE Transactions on*, vol. 54, no. 12, pp. 5277–5307, 2008.

[11] A. Goupil, M. Colas, G. Gelle, and D. Declercq, "FFT-based BP decoding of general LDPC codes over abelian groups," *IEEE Trans. on Commun.*, vol. 55, no. 4, pp. 644–649, April 2007.

[12] L. Barnault and D. Declercq, "Fast decoding algorithm for ldpc over gf(2^q)," in *in the proc. of ITW'03*, Paris, France, 2003.

[13] D. Declercq and M. Fossorier, "Decoding algorithms for nonbinary ldpc codes over gf(q)," *IEEE Trans. on Commun.*, vol. 55, no. 4, pp. 633–643, April 2007.

[14] A. Voicila, D. Declercq, F. Verdier, M. Fossorier, and P. Urard, "Low complexity decoding for non-binary ldpc codes in high order fields," *IEEE Trans. on Commun.*, vol. 58, no. 5, pp. 1365–1375, May 2010.

[15] X. Y. Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular progressive edge-growth tanner graphs," *IEEE Trans. Inform. Theory*, vol. 51, no. 1, pp. 386–398, 2005.

[16] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *Communications, IEEE Transactions on*, vol. 49, no. 10, pp. 1727–1737, 2002.

[17] V. Rathi and R. Urbanke, "Thresholds and the stability condition for non-binary LDPC codes," *IEE Proc. Comm.*, vol. 152, no. 6, 2005.

[18] D. Declercq and M. Fossorier, "Improved impulse method to evaluate the low weight profile of sparse binary linear codes," in *Proc. of ISIT'08*, Toronto, Canada, July 2008.

[19] A.E. Brouwer and T. Verhoeff, "An updated table of minimum-distance for binary linear codes," *IEEE Trans. on Info. Theory*, vol. 39, no. 2, pp. 662–677, March 1993.

[20] D. MacKay, "Webpage," http://www.inference.phy.cam.ac.uk/mackay/codes/data.html.