



# Good coupling between LDPC-Staircase and Reed-Solomon for the design of GLDPC codes for the Erasure Channel

Ferdaouss Mattoussi, Bessem Sayadi, Vincent Roca

## ► To cite this version:

Ferdaouss Mattoussi, Bessem Sayadi, Vincent Roca. Good coupling between LDPC-Staircase and Reed-Solomon for the design of GLDPC codes for the Erasure Channel. IEEE Wireless Communications and Networking Conference (WCNC'13), Apr 2013, Shanghai, China. IEEE, 2013. <hal-00781921>

**HAL Id: hal-00781921**

**<https://hal.inria.fr/hal-00781921>**

Submitted on 28 Jan 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Good coupling between LDPC-Staircase and Reed-Solomon for the design of GLDPC codes for the Erasure Channel

Ferdaouss Mattoussi\*

\*Inria, France

Bessem Sayadi<sup>‡</sup>

<sup>‡</sup>Alcatel-Lucent Bell Labs, France

Vincent Roca\*

{ferdaouss.mattoussi,vincent.roca}@inria.fr, bessem.sayadi@alcatel-lucent.com

**Abstract**—In this paper we analyze the design of Generalized LDPC-staircase (GLDPC-staircase) codes, where the base code is an LDPC-Staircase code and component codes are Reed-Solomon codes. More precisely we compare two schemes: scheme A has the property that on each check node of the base code the repair symbol generated by the LDPC code is also a Reed-Solomon repair symbol. On the opposite, with scheme B for each check node the repair symbols generated by the LDPC code are Reed-Solomon source symbols. In this work we perform a behavioral analysis of the two schemes in order to determine the best one for Iterative + Reed Solomon (IT+RS) and Maximum Likelihood (ML) decoding. To that purpose we use an asymptotic analysis using Density evolution (DE) and Extrinsic Information Transfer techniques, as well as a finite length analysis. We show that scheme A is globally the best solution since it significantly performs better than scheme B with an (IT+RS) decoding and yields similar performance with ML decoding.

1

## I. INTRODUCTION

Application Level Forward Erasure Correction (AL-FEC) codes are now a key component of reliable multicast/broadcast transmission systems. They are the key building block of the FLUTE/ALC (RFC 6726) [1] reliable multicast transport protocol that is used to push any kind of files (e.g. multimedia) for instance over the wireless 3G/4G channels (e.g. they are part of the 3GPP MBMS, DVB-H/SH IPDatacasting, or ISDB-Tmm services). They are also the key building block of robust streaming protocols, like the FECFRAME (RFC 6363) [2] transport protocol, that is also included in the above systems.

The codes we consider are AL-FEC codes for the erasure channel, that can be used in FLUTE/ALC and FECFRAME. Among them Low Density Parity Check (LDPC) codes are of particular interest. The LDPC codes have been intensively studied due to their near-Shannon limit performance under iterative Belief-Propagation (BP) decoding [3][4]. A  $(N, K)$  LDPC code, where  $N$  is the code length and  $K$  is its dimension, can be graphically represented as a bipartite graph with  $N$  "variable nodes" (VN) and  $M = N - K$  "check nodes" (CN). Equivalently, LDPC codes can be represented through their parity check matrix,  $H_L$ , that reflects the connections between (VN) and (CN). The degree of a VN or a CN is

defined as the number of edges connected to it. A VN of degree  $n$  can be interpreted as a "Length Repetition Code"  $(n, 1)$ , i.e. as a linear block code repeating  $n$  times its single information symbol towards the CN set. Similarly, a CN of degree  $n$  can be interpreted as a Single Parity Check (SPC) code  $(n, n - 1)$ , i.e. as a linear block code associated with one parity equation.

### A. Related Works

To improve error floor and minimal distance, a generalization of these codes was suggested by Tanner in [4], for which subsets of the variable nodes obey a more complex constraint than an SPC constraint. The SPC check nodes in a GLDPC structure are replaced with a generic linear block codes  $(n, k)$  referred to as sub-codes or component codes, while the sparse graph representation is kept unchanged. More powerful decoders at the check nodes have been investigated by several researchers in recent years after the work of Boutros et al. [5] and Lentmaier and Zigangirov [6] where BCH codes and Hamming codes were proposed as component codes respectively. Later several works, on several types of channels, have been carried out in order to afford very large minimum distance and exhibit performance approaching Shannon's limit. Each construction differs by modifying the linear block codes (components codes) on the check nodes such as [7][8][9][10][11][12], and/or the distribution of the structure of GLDPC codes [7] to offer a good balance between waterfall performance and error floor under iterative decoding.

Recently, a GLDPC code construction using LDPC-Staircase [13] (a structured code that shares similarities with IRA codes [14]) as the base codes and Reed-Solomon (RS) as component codes has been proposed in [15][16]. These codes, based on what we call in this work *scheme A*, can easily be tuned to behave either like predefined rate LDPC-Staircase codes at one extreme, or like a Reed Solomon code at another extreme. This construction also allows each RS component code to potentially produce a large number of repair symbols (called *extra-repair symbols*) on-demand, which means that a small rate is achievable, a feature that may be suited to situations where the channel conditions are worse than expected, or to fountain like content distribution applications. Finally [16] assesses the performance of GLDPC-Staircase codes when

<sup>1</sup>This work was supported by the ANR-09-VERS-019-02 grant (ARSSO project) and by the INRIA - Alcatel Lucent Bell Labs joint laboratory.

using a hybrid decoding scheme based on Iterative (IT), Reed-Solomon (RS) and Maximum Likelihood (ML) decoding, a solution that benefits from the same correction capabilities as an ML decoder but reduces decoding complexity by using (IT+RS) decoding first.

### B. Contributions of this Work

In this paper, we compare the GLDPC-Staircase construction scheme of [16] to another scheme in order to determine the best coupling between the LDPC-Staircase and RS codes for (IT+RS) and ML decoding. The first scheme, called scheme A, has the property that a generated *LDPC-Staircase repair symbol* of a given CN is also the first *RS repair symbol* of that CN (we explain how to achieve this later). On the opposite, in scheme B, this generated LDPC-Staircase repair symbol is a *RS source symbols* for that CN. Therefore, for a given CN, the number of symbols that must be available (i.e. either received or decoded) for the missing symbols of this CN to be recovered by RS decoding is higher than in scheme A. It follows that intuitively we expect better decoding performance with scheme A than with scheme B. The main contribution of this paper is to prove, both through a finite length analysis and through an asymptotic analysis based on Density evolution (DE) and EXtrinsic Information Transfer (EXIT) techniques that scheme A performs the best under (IT+RS) decoding, whereas both schemes behave the same under ML decoding. Another contribution is the extension of DE and EXIT techniques to GLDPC-Staircase codes, which is far from trivial.

The paper is organized as follows. Section II focuses on the design of GLDPC-Staircase codes with the two coding schemes. In section III we derive the DE equations of the two schemes and the decoding thresholds using EXIT functions. Section IV provides a finite length performance evaluation for the two schemes. Finally, we conclude.

## II. GLDPC-STAIRCASE CODES DESIGN

This section introduces the GLDPC-Staircase code design and continues with the encoding and decoding methods.

### A. Code Construction

GLDPC-Staircase codes are composed of:

- **LDPC-Staircase base codes** [13]: Let  $H_L$  be the (binary) parity-check matrix, of size  $M_L = N_L - K$  rows and  $N_L$  columns.  $N_L$  and  $K$  are respectively the LDPC-Staircase code length and dimension and  $R_L = K/N_L$  is the associated code rate.  $H_L$  has the form  $(H_1|H_2)$ .  $H_1$  is the  $M_L \times K$  left-hand side part (information) and each column has the same degree,  $N_1$  (number of "1s" per column).  $H_2$  is the  $M_L \times M_L$  right-hand side part (redundancy) and features a staircase (A.K.A. double diagonal) structure.  $H_1$  is created in a fully regular way, in order to have constant column and row degrees [13]. Each row  $H_1$  is of degree  $\frac{N_1}{r_L - 1}$ , and because of the staircase structure of  $H_2$ , a row  $m$  of  $H_L$  is of degree  $d_1 = \frac{N_1}{r_L - 1} + 1$  or  $d_{m>1} = \frac{N_1}{r_L - 1} + 2$ . These codes

are structured LDPC codes that can also be regarded as irregular codes if we consider the whole parity check matrix.

- **Reed Solomon inner codes**: Each RS code is associated to each row in  $H_L$  (i.e. CN) to generate  $E$  extra repair symbols. Therefore the  $E$  extra-repair symbols associated to the  $m^{th}$  row of  $H_L$  are generated by an  $RS(n_m, k_m)$  encoding over  $GF(2^8)$ .  $n_m$  and  $k_m$  are respectively the RS code length and dimension. The difference between the two schemes is in the definition of  $n_m$  and  $k_m$ :
  - for scheme A, in row  $m > 1$ , the various source symbols (from the user point of view) involved in this row plus the *previous repair symbol* are considered as source symbols from the RS point of view. The generated LDPC-Staircase repair symbol on this row plus the  $E$  extra-repair symbols are considered as repair symbols from the RS point of view (i.e. *the new LDPC-Staircase repair symbol is also an RS repair symbol*). For  $m = 1$  the only difference is the fact there is no previous repair symbol (it's the beginning of the staircase). So we have:  $n_m = k_m + 1 + E$ , and  $k_m = d_m - 1$ , no matter the row. To obtain this scheme we use "quasi" Hankel matrix as base matrix to construct the RS code as explained in [16][17].
  - for scheme B, in row  $m > 1$ , the various source symbols (from the user point of view) involved in this row plus the *two repair symbols* are considered as source symbols from the RS point of view. The  $E$  extra-repair symbols are considered as repair symbols from the RS point of view. So we have:  $n_m = k_m + E$ , and  $k_m = d_m$ , no matter the row. To obtain this scheme we use Vandermonde matrix as base matrix to construct the RS code.

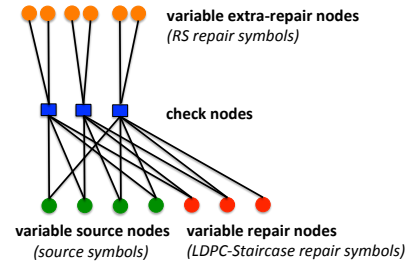


Fig. 1. GLDPC-Staircase(13, 4) code,  $E = 2$  extra repair symbols per CN.

The GLDPC-Staircase  $(N_G, K)$  codes, where  $N_G = N_L + M_L * E$ , can also be represented by a Tanner graph (Figure 1). All the CN have the same number,  $E$ , of extra-repair symbols. So, if  $r_L$  is the LDPC-Staircase base code rate, then the GLDPC-Staircase code rate is:

$$r_G = \frac{r_L}{1 + (1 - r_L)E} \quad (1)$$

### B. Encoding

Let  $S = (s_1, s_2, \dots, s_K)$  be the source symbols. For row  $m$  of  $H_L$ , let  $X = (x_1, \dots, x_{k_m})$  be the subset of  $S$  that are

involved in this row. Two types of repair symbols are produced during encoding:

- $M_L$  LDPC-Staircase repair symbols,  $(p_1, \dots, p_{M_L})$ : they are computed as usual, by "following the stairs" of  $H_L$ ;
- $M_L * E$  extra-repair symbols,  $(e_1, \dots, e_{M_L * E})$ : they are computed by multiplying the  $k_m$  symbols by the systematic generator matrix  $G_m$  of  $RS(n_m, k_m)$ . For scheme A, the  $k_m$  symbols consist of  $X$  symbols plus  $p_{m-1}$  (if  $m \neq 1$ ), whereas they consist of  $X$  plus the two LDPC repair symbols for scheme B.

An advantage of these two schemes is the fact that extra-repair symbols can be produced incrementally, on demand, rather than all at once. Another advantage, is the fact the  $E$  parameter can be rather large (it is only limited by the finite field size, usually  $GF(2^8)$ ). Said differently, GLDPC-Staircase codes can be dynamically be turned into small rate codes.

### C. Decoding

Two kinds of decoders are considered:

- Iterative plus Reed-Solomon (IT+RS) decoding: It consists of an IT decoding over the binary LDPC-Staircase system (i.e. by ignoring extra-repair symbols), associated to a RS decoding for each CN. Both decoders work together, in a recursive way, and RS decoding for a CN can sometimes recover fresh symbols (in the  $X$  plus  $p_m, p_{m-1}$  set) that may help IT decoding, and vice-versa.
- Maximum Likelihood (ML) decoding: in order to reduce decoding complexity, this is usually performed in a second step, if (IT+RS) decoding failed to recover all source symbols. This ML decoding consists of a Gaussian Elimination over the full system, considering both the binary and non-binary (over  $GF(2^8)$ ) equations (see [16]).

## III. ASYMPTOTIC PERFORMANCE ANALYSIS UNDER (IT+RS) AND ML DECODING

### A. Introduction

EXtrinsic Information Transfer (EXIT) curves are introduced in [18] as a technique to analyze the convergence of IT decoding process of parallel concatenated component codes. For binary LDPC codes, a slightly different definition of the EXIT curve is introduced in [19], where the EXIT curve is associated with the sparse graph *system* rather than with *component* codes. Roughly speaking, this EXIT curve gives the fraction of erased bits "contained" in the *extrinsic* information produced by the decoding algorithm, assuming that the code length tends to infinity and cycle free Tanner graph. It can be defined for any decoding algorithm (e.g. IT and ML) and it relates to the asymptotic performance of an ensemble of codes under the considered decoding.

In [20] we extend this method to the coding scheme A under (IT+RS) decoding and ML decoding. In case of (IT+RS) decoding, there is a tight relation between the EXIT curve and the density evolution (DE) equations of GLDPC Staircase codes under (IT+RS) decoding. For the ML decoding, it is shown in [19] that the area under the EXIT curve is always

equal to the asymptotic rate of the ensemble. This allows to derive an upper bound of the ML threshold, which is conjectured to be tight in quite general settings, especially for codes defined by almost regular graphs.

In this work we derive the DE equations of scheme B in order to determine its EXIT curve under (IT+RS) decoding (Section III-B2) and we provide the upper bound on the ML threshold (Section III-C). Then we discuss the two schemes using DE and EXIT functions (Sections III-B3 and III-C).

### B. Density Evolution (DE) Equations

1) *Preliminaries*: In the sequel, we denote by  $\hat{d}_v$  and  $\hat{d}_c$  the maximum variable and check node degrees in the bipartite (Tanner) graph associated with the LDPC-Staircase code. Following [21], we define the edge-perspective Degree Distribution (DD) polynomials by  $\lambda(x) = \sum_{d=1}^{\hat{d}_v} \lambda_d x^{d-1}$  and  $\rho(x) = \sum_{d=1}^{\hat{d}_c} \rho_d x^{d-1}$ , where  $\lambda_d$  (resp.  $\rho_d$ ) represents the fraction of edges connected to variable-nodes (resp. check-nodes) of degree  $d$ . From a node perspective, the DD polynomials are given by  $L(x) = \sum_{d=1}^{\hat{d}_v} L_d x^d$  and  $R(x) = \sum_{d=1}^{\hat{d}_c} R_d x^d$  where  $L_d$  (resp.  $R_d$ ) represents the fraction of variable-nodes (resp. check-nodes) of degree  $d$ .

Given a GLDPC-Staircase code, DD polynomials  $\lambda$  and  $\rho$  are defined by the underlying LDPC-Staircase code, defined by the bottom graph of Figure 1 (that is, not containing the extra-repair nodes). We denote by  $\mathcal{E}(\lambda, \rho, E)$  the ensemble of GLDPC-Staircase with edge-perspective DD polynomials  $\lambda$  and  $\rho$ , and with  $E$  extra-repair symbols per check-node.

Assume that an arbitrary code from  $\mathcal{E}(\lambda, \rho, E)$ , of length  $N_G$ , is used over the BEC, and let  $\varepsilon$  denote the channel erasure probability. The probability threshold of the ensemble  $\mathcal{E}(\lambda, \rho, E)$  is defined as the supremum value of  $\varepsilon$  (that is, the worst channel condition) that allows transmission with an arbitrary small error probability, assuming that  $N_G$  goes to infinity. The threshold value of a given ensemble of codes can be efficiently computed by using the *Density Evolution* (DE) method [21], which recursively computes the fraction of erased messages passed during the belief propagation decoding. Density evolution equations are given and derived in the next section for scheme A and B respectively.

2) *DE Equations for GLDPC-Staircase Codes*  $\mathcal{E}(\lambda, \rho, E)$ : In the sequel, the degree of a check or variable (source or repair) node will always refer to its degree in the underlying LDPC-Staircase code. We are interested in the erasure probability of messages exchanges by the IT+RS decoding along the edges of the LDPC-Staircase code. We denote by  $P_{\ell,A}$  ( $P_{\ell,B}$ ), the probability of a LDPC symbol (source or repair) node sending an erasure at iteration  $\ell$  using scheme A (scheme B). Similarly,  $Q_{\ell,A}$  ( $Q_{\ell,B}$ ) denotes the probability of a check node sending an erasure (to an LDPC symbol-node) at iteration  $\ell$  using scheme A (scheme B). Clearly,  $P_0$  is equal to the channel erasure probability  $\varepsilon$ .

**Scheme A:** The first repair symbol generated by any RS code is one of the repair symbols of the LDPC-staircase code.

Consider a constraint node  $c$  connected to symbol-nodes  $(v_1, \dots, v_d, e_1, \dots, e_E)$  where  $v_i$  denotes an LDPC (source or repair) symbol node and  $e_i$  denotes an extra-repair node. Since  $c$  corresponds to an RS code, it can recover the value of an LDPC symbol node, say  $v_1$ , if and only if the number of erasures among the other symbol-nodes  $(v_2, \dots, v_d, e_1, \dots, e_E)$  is less than or equal to  $E$ . Now, at iteration  $\ell$ , the LDPC symbols are erased with probability  $P_{\ell,A}$ , while extra repair symbols are always erased with probability  $\varepsilon$ , the channel erasure probability. It follows that the probability of a check node of degree  $d$  recovering the value of an LDPC symbol at iteration  $\ell + 1$ , denoted by  $\bar{Q}_{\ell+1,A}(d)$ , is given by:

$$\bar{Q}_{\ell+1,A}(d) = (1 - P_{\ell,A})^{d-1} + \sum_{i=1}^{\min(d-1, E)} \sum_{j=0}^{E-i} \binom{d-1}{i} P_{\ell,A}^i (1 - P_{\ell,A})^{d-1-i} \binom{E}{j} \varepsilon^j (1 - \varepsilon)^{E-j} \quad (2)$$

Averaging over all possible values of  $d$ , we get:

$$Q_{\ell+1,A} = 1 - \sum_{d=1}^{\hat{d}_c} \rho_d \bar{Q}_{\ell+1,A}(d) \quad (3)$$

Conversely, an LDPC symbol node  $v$  of degree  $d$ , connected to check nodes  $c_1, \dots, c_d$ , sends an erasure to the check node  $c_1$  iff it was erased by the channel, and it received erased messages from all check nodes  $c_2, \dots, c_d$ . Since this happens with probability  $\varepsilon \cdot Q_{\ell+1,A}^{d-1}$ , and averaging over all possible degrees  $d$ , we get:

$$P_{\ell+1,A} = \varepsilon \sum_{d=1}^{\hat{d}_v} \lambda_d Q_{\ell+1,A}^{d-1} = \varepsilon \lambda(Q_{\ell+1,A}) \quad (4)$$

For more details, the reader can refer to [20].

**Scheme B:** All the LDPC-staircase repair symbols are source symbols for the RS codes.

Consider a constraint node  $c$  connected to symbol-nodes  $(v_1, \dots, v_d, e_1, \dots, e_E)$  where  $v_i$  denotes an LDPC (source or repair) symbol node and  $e_i$  denotes an RS-repair node. The node  $c$  corresponds both to a parity check constraint between LDPC symbol nodes  $(v_1, \dots, v_d)$  and to RS linear constraints between all the symbol-nodes  $(v_1, \dots, v_d, e_1, \dots, e_E)$ . Thus,  $c$  can recover the value of an LDPC symbol node, say  $v_1$ , if and only if one of the following (disjoint conditions) holds:

- (1) there are no erased symbols among  $v_2, \dots, v_d$ ;
- (2) there is at least one erased symbol among  $v_2, \dots, v_d$ , but the number of erasures among all the symbol-nodes  $(v_1, \dots, v_d, e_1, \dots, e_E)$  is less than or equal to  $E - 1$ .

The second condition is also equivalent to the following one:

- (2') the number of erased symbols among  $v_2, \dots, v_d$  is equal to  $i$  and the number of erased symbols among  $e_1, \dots, e_E$  is equal to  $j$ , with  $1 \leq i \leq \min(d - 1, E - 1)$  and  $0 \leq j \leq E - 1 - i$ .

Now, at iteration  $\ell$ , the LDPC symbols are erased with probability  $P_{\ell,B}$ , while RS repair symbols are always erased with probability  $\varepsilon$ , the channel erasure probability. It follows that the probability of a check node of degree  $d$  recovering

the value of an LDPC symbol at iteration  $\ell + 1$ , denoted by  $\bar{Q}_{\ell+1,B}(d)$ , is given by:

$$\bar{Q}_{\ell+1,B}(d) = (1 - P_{\ell,B})^{d-1} + \sum_{i=1}^{\min(d-1, E-1)} \sum_{j=0}^{E-1-i} \binom{d-1}{i} P_{\ell,B}^i (1 - P_{\ell,B})^{d-1-i} \binom{E}{j} \varepsilon^j (1 - \varepsilon)^{E-j} \quad (5)$$

Averaging over all possible values of  $d$ , we get:

$$Q_{\ell+1,B} = 1 - \sum_{d=1}^{\hat{d}_c} \rho_d \bar{Q}_{\ell+1,B}(d) \quad (6)$$

Conversely, an LDPC symbol node  $v$  of degree  $d$ , connected to check nodes  $c_1, \dots, c_d$ , sends an erasure to the check node  $c_1$  iff it was erased by the channel, and it received erased messages from all check nodes  $c_2, \dots, c_d$ . Since this happens with probability  $\varepsilon \cdot Q_{\ell+1,B}^{d-1}$ , and averaging over all possible degrees  $d$ , we get:

$$P_{\ell+1,B} = \varepsilon \sum_{d=1}^{\hat{d}_v} \lambda_d Q_{\ell+1,B}^{d-1} = \varepsilon \lambda(Q_{\ell+1,B}) \quad (7)$$

For both schemes, we can determine a recursive relation between  $P_\ell$  and  $P_{\ell+1}$ , with  $P_0 = \varepsilon$ . The decoder can recover from a fraction of  $\varepsilon$  erased symbols iff  $\lim_{\ell \rightarrow +\infty} P_\ell = 0$ . Therefore, the threshold probability can be computed by:

$$\epsilon^{(\text{IT+RS})}(\lambda, \rho, E) = \max\{P_0 \mid \lim_{\ell \rightarrow +\infty} P_\ell = 0\} \quad (8)$$

When no confusion is possible, the above threshold value will be simply denoted by  $\epsilon^{(\text{IT+RS})}$ .

3) *Results and Comparison:* We plot in Figure 2, the evolution of the erasure probability transfer on the graph of GLDPC Staircase code with  $r_G = \frac{1}{2}$  and  $E=3$  for schemes A and B. These curves represent the value of the erasure probability on all the LDPC symbols during the propagation of the erasure probability between check nodes and variable nodes of the Tanner graph where the channel erasure probability is equal to 0.32. This figure shows that the initial fraction of erasure

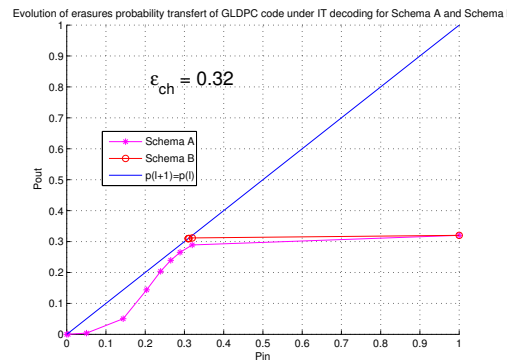


Fig. 2. The evolution, for schemes A and B, of the (IT+RS) decoding process for the (d,d) pair  $(\lambda(x), \rho(x)) = (0.0909 \cdot x^1 + 0.9091 \cdot x^4, x^{21})$  of LDPC Staircase, ( $r_G = \frac{1}{2}$   $E=3$ ) and  $\varepsilon = 0.32$ .

messages emitted by the LDPC variable nodes is  $P_\ell = 1$  in schemes A and B. After an iteration (at the next output of the LDPC variable nodes) this fraction has evolved to  $P_{\ell+1} = 0.32$  for the two schemes. After the second full iteration, i.e.

at the output of the LDPC variable nodes, we see an erasure fraction of scheme A equal to  $P = 0.2889$  whereas is equal to  $P = 0.3117$  for scheme B. This difference explains that the erasure probability in scheme A decreases more quickly than scheme B (i.e the correction of the erasure in scheme A is better). After that, the processes of the transfer continues in the same way. The figure shows also that the processes of DE for scheme B get stuck at a value  $> 0$  ( $P = 0.3094$ ), whereas for scheme A the processes finishes with  $P = 0$ . This means that at  $\epsilon = 0.32$  the GLDPC-Staircase code converges only with scheme A (i.e can recover all the erased LDPC Symbols).

### C. Exit Functions

To derive the  $\epsilon^{(IT+RS)}$  and the upper bound of the ML threshold ( $\bar{\epsilon}^{ML} \geq \epsilon^{ML}$ ) obtained by the scheme A, in [20] we extended the EXIT function method proposed in [19]. This upper bound is conjectured to be tight in a quite general settings, especially for codes defined by almost regular graphs (e.g. LDPC-Staircase codes) (see [20] for details). In this section, we apply and extend the same method to determine  $\epsilon^{(IT+RS)}$  and  $\bar{\epsilon}^{ML}$  for scheme B. Table I provides the comparison in terms of  $\epsilon^{(IT+RS)}$  and  $\bar{\epsilon}^{ML}$  between scheme A and B for  $r_G = \frac{1}{2}$  and we vary the base code rate  $r_L$  (i.e vary E). This table shows that, for different values of  $E > 0$ , the (IT+RS) decoding threshold of scheme A is higher than that of scheme B. On the opposite, the ML decoding thresholds of the two schemes, for different values of E, are almost equivalent.

Threshold	E	scheme A	scheme B
$\epsilon^{(IT+RS)}$	0	0.4380	0.4380
	1	0.3943	0.2709
	2	0.3481	0.2643
	3	0.3443	0.2819
$\bar{\epsilon}^{ML}$	0	0.4946	0.4946
	1	0.4993	0.4998
	2	0.4999	0.4999
	3	0.4999	0.4999

TABLE I  
THRESHOLD COMPARISON OF (IT+RS) AND ML DECODING, WITH  $r_G = \frac{1}{2}$ .

Table II provides the comparison in terms of  $\epsilon^{(IT+RS)}$  and  $\bar{\epsilon}^{ML}$  between scheme A and B for different values of  $r_G$  with  $r_L = \frac{2}{3}$ . We can do the same remarks as that of table I.

Threshold	$r_G$	scheme A	scheme B	Shannon limit
$\epsilon^{(IT+RS)}$	(E=0)	0.2709	0.2709	0.3333
	(E=1)	0.3943	0.2709	0.5
	(E=2)	0.4744	0.3639	0.6
	(E=3)	0.5332	0.4394	0.6667
$\bar{\epsilon}^{ML}$	(E=0)	0.3298	0.3298	0.3333
	(E=1)	0.4993	0.4998	0.5
	(E=2)	0.5998	0.5999	0.6
	(E=3)	0.6665	0.6666	0.6667

TABLE II  
THRESHOLD COMPARISON OF (IT+RS) AND ML DECODING FOR DIFFERENT VALUES OF  $r_G$ , WITH  $r_L = \frac{2}{3}$ .

## IV. FINITE LENGTH ANALYSIS FOR SCHEMES A AND B UNDER (IT+RS) AND ML DECODING

We now perform a finite length analysis of both schemes under (IT+RS) or ML decoding. We measure the average

decoding inefficiency ratio (i.e. the ratio between the number of symbols needed for decoding to succeed over the number of source symbols,  $\frac{nb_{symbols\_needed}}{K} = 1 + \epsilon$ , where  $\epsilon$  is also called transmission overhead), and the decoding failure probability (i.e. the probability that at least one erased source symbol is not recovered) as a function of the transmission overhead. In all cases we consider a random transmission order of all symbols and IID random losses<sup>2</sup>.

### A. Average Decoding Performance

Figures 3(c) and 3(d) provide the average (over 1,000 different codes and transmission schemes) decoding inefficiency ratio of both schemes under ML decoding for two code rates,  $\frac{1}{2}$  and  $\frac{1}{3}$ . They show that no matter the block size (i.e. code dimension, K), both schemes perform the same, with results close to that of MDS codes (characterized by an decoding inefficiency ratio always equal to 1). This results holds for the two code rates considered. Figures 3(a) and 3(b) do the same in case of (IT+RS) decoding only (no ML). They show that scheme A outperforms scheme B in all cases. This is made possible by a higher number of RS repair symbols with scheme A, which mechanically increases the probability of decoding an erased symbol for this CN. This increase also avoids stopping sets associated to short cycles, which means that scheme A performs better than scheme B under (IT+RS) decoding. However we observe the same performance with ML decoding because both schemes have the same system to resolve, the difference being only on the nature of symbols.

### B. Decoding Failure Probability

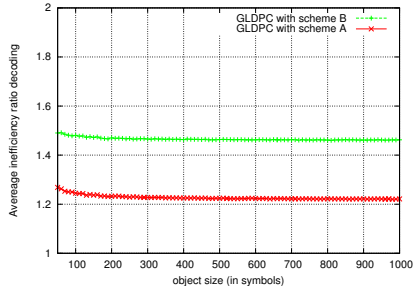
We now analyze the impacts of the coding scheme on the ML decoding failure probability Figure 4 shows no significant behavior difference for both schemes: they have a very small decoding overhead, close to that of MDS codes (vertical line), and receiving 6 symbols in addition to  $K$  enables the decoding failure probability to fall bellow  $10^{-5}$ , an excellent result.

## V. CONCLUSIONS

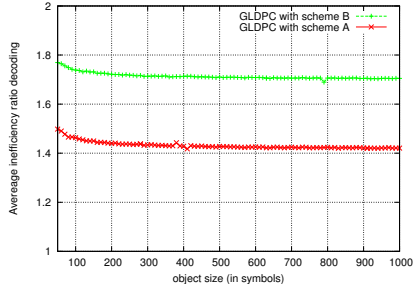
This work considers two different ways of coupling LDPC-Staircase and Reed-Solomon codes for (IT+RS) and ML decoding. The difference in the resulting GLDPC-Staircase codes lies in the repair symbol generated by the LDPC code being part (scheme A) or not (scheme B) of the RS repair symbols. Finite length and asymptotic analysis show that scheme A is the best under (IT+RS) decoding whereas the two schemes behave the same under ML decoding. Since (IT+RS) decoding is significantly faster than ML decoding and since (IT+RS) is used first during decoding in order to decode totally (good reception conditions) or partially the linear system, we highly recommended to use scheme A. This work also confirms the excellent performance of GLDPC-Staircase codes (when  $K = 1,000$ , decoding failure probability falls bellow  $10^{-5}$  with a 6 symbol overhead only), even for very small blocks and not only asymptotically.

<sup>2</sup>Since the symbol transmission order is randomized, the IID or non-IID nature of erasures does not impact the results.

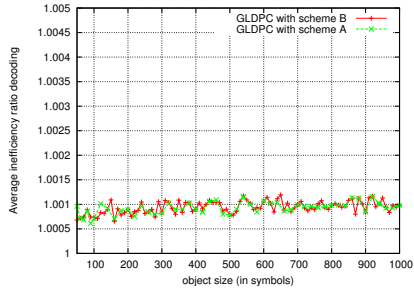




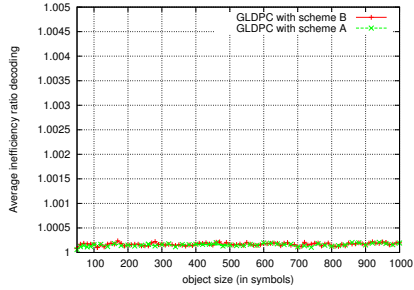
(a) (IT+RS) decoding,  $r_G = \frac{1}{2}$  ( $E = 1$ )



(b) (IT+RS) decoding,  $r_G = \frac{1}{3}$  ( $E = 3$ )



(c) ML decoding,  $r_G = \frac{1}{2}$  ( $E = 1$ )



(d) ML decoding,  $r_G = \frac{1}{3}$  ( $E = 3$ )

Fig. 3. Average performance under (IT+RS) or ML decoding, with rate  $r_G = \frac{1}{2}$  or  $\frac{1}{3}$ , as a function of  $K$ .

## REFERENCES

- [1] T. Paila, M. Luby, R. Lehtonen, V. Roca, and R. Walsh, *FLUTE - File Delivery over Unidirectional Transport*, Nov. 2012, IETF RMT Working Group, Request For Comments, RFC 6726 (obsoletes RFC 3926).
- [2] Mark Watson, Ali Begen, and Vincent Roca, *Forward Error Correction (FEC) Framework*, June 2011, IETF Request for Comments, RFC 6363 (Standards Track/Proposed Standard).
- [3] D. MacKay and R. Neal, "Near Shannon limit performance of low density parity check codes," *IET Electronics Letters*, vol. 33, no. 6, 1997.

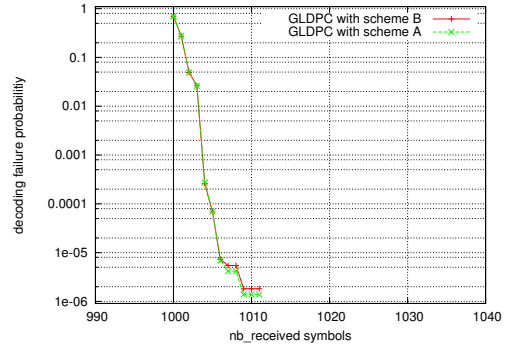


Fig. 4. Decoding failure probability under ML, with rate  $r_G = \frac{1}{2}$ ,  $E = 1$ , and  $k = 1000$  symbols, as a function of the overhead.

- [4] R. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. on Information Theory*, vol. 27, no. 5, Sept. 1981.
- [5] J. Boutros, O. Pothier, and G. Zémor, "Generalized Low Density (Tanner) Codes," *IEEE Int. Conference on Communications (ICC'99)*, vol. 1, 1999.
- [6] M. Lentmaier and K. Zigangirov, "On Generalized Low-Density Parity-Check Codes Based on Hamming Component Codes," *IEEE Trans. on Communications*, vol. 3, no. 8, 1999.
- [7] G. Yue, L. Ping, and X. Wang, "Generalized Low-Density Parity-Check Codes Based on Hadamard Constraints," *IEEE Trans. on Information Theory*, vol. 53, no. 3, pp. 1058–1079, 2007.
- [8] J. Chen and R. Tanner, "A Hybrid Coding Scheme for the Gilbert Elliott Channel," *IEEE Trans. on Communications*, vol. 54, no. 10, Oct. 2006.
- [9] N. Miladinovic and M. Fossorier, "Generalized LDPC Codes with Reed-Solomon and BCH Codes as Component Codes for Binary Channels," *IEEE Global Telecommunications Conference (GLOBECOM)*, 2005.
- [10] I. Djordjevic, O. Milenkovic, and B. Vasic, "Generalized Low-Density Parity-Check Codes for Optical Communication Systems," *Lightwave Technology*, vol. 23, no. 5, 2005.
- [11] Y. Wang and M. Fossorier, "Doubly Generalized LDPC Codes," *IEEE Int. Symposium on Information Theory*, July 2006.
- [12] E. Paolini, M. Fossorier, and M. Chiani, "Analysis of Doubly-Generalized LDPC Codes with Random Component Codes for the Binary Erasure Channel," *Proceedings of Allerton Conference on Communications, Control and Computing*, 2006.
- [13] V. Roca, C. Neumann, and D. Furodet, *Low Density Parity Check (LDPC) Staircase and Triangle Forward Error Correction (FEC) Schemes*, jun 2008, IETF Request for Comments, RFC 5170.
- [14] H. Jin, A. Khandekar, and R. McEliece, "Irregular repeat-accumulate codes," in *Int. Symp. on Turbo codes and Related Topics*, Sept. 2010.
- [15] M. Cunche, V. Savin, V. Roca, G. Kraidy, A. Soro, and J. Lacan, "Low-rate coding using incremental redundancy for GLDPC codes," *IEEE Int. Workshop on Satellite and Space Communications*, Oct. 2008.
- [16] F. Mattoussi, V. Roca, and B. Sayadi, "Design of small rate, close to ideal, gldpc-staircase al-fec codes for the erasure channel," in *IEEE Global Communications Conference (GLOBECOM 2012)*, Dec. 2012.
- [17] F. Mattoussi, V. Roca, and B. Sayadi, "Complexity comparison of the use of vandermonde versus hankel matrices to build systematic mds reed solomon codes," *IEEE Int. Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2012.
- [18] S. Ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. on Communications*, vol. 49, no. 10, 2001.
- [19] C. Méasson and R. Urbanke, "An upper-bound for the ML threshold of iterative coding systems over the BEC," in *Proc. of the 41st Allerton Conf. on Communications, Control and Computing*, October 2003, p. 3.
- [20] F. Mattoussi, V. Savin, V. Roca, and B. Sayadi, "Optimization with exit functions of gldpc-staircase codes for the bec," *IEEE Int. Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2012.
- [21] T.J. Richardson and R.L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. on Information Theory*, vol. 47, no. 2, 2001.