Rate Splitting Issue for Finite Length Raptor Codes

Auguste Venkiah, Charly Poulliat, David Declercq
ETIS - CNRS - ENSEA - University of Cergy-Pontoise, F-95000
email:{venkiah,poulliat,declercq}@ensea.fr

***Abstract*— In this paper, we discuss the rate splitting issue for the design of finite length Raptor codes, in a joint decoding framework. We show that the choice of a rate lower than usually proposed for the precode enables to design Raptor codes that perform well at small lengths, with almost no asymptotic loss. We show in particular that the error floor can be greatly reduced by properly choosing the rate splitting between the precode and the LT code. Those behaviors are demonstrated both on the BEC and the BIAWGN channel.**

I. Introduction

The[1] main property of Fountain codes, originally introduced [1] for communicating over an erasure channel with unknown erasure probability, is that they are naturally rateless. A fountain code produces a potentially limitless number of independent output symbols. Whereas traditional block codes are characterized by their design rate and require puncturing to achieve higher rates, a fountain code achieves this naturally by adapting the number of output symbols. LT codes, the first class of efficient fountain codes introduced by Luby [2], are fully characterized by an output degree distribution, and good performance is achieved by optimizing this distribution. Unfortunately, arbitrarily low decoding error probability can only be obtained at complexity that is too high to ensure linear encoding and decoding time. To circumvent this problem, Raptor codes were introduced by Shokrollahi [3] as an extension of LT codes. A Raptor code is built from the concatenation of an LT code and an outer code, called precode, which is a high rate error correcting block code. In Raptor codes, the LT code does not necessarily reach full symbol recovery, but needs only to recover a large enough proportion of input symbols, the precode being in charge of recovering the remaining erased symbols. The concatenated structure of Raptor codes enables the design of output degree distributions for the LT code with constant average degree *i.e.* with the property of linear encoding and decoding time.

There are two main issues for the design of a Raptor code: the design of an output degree distribution, and the choice of the precode. Much attention has been paid to the design of output degree distributions, for various types of channels, both when the two component codes of the Raptor code are decoded sequentially [3], [4], or jointly [5]. Very little attention has been paid on the precode design problem. Although it has been suggested for practical constructions [3] to use a concatenation of Hamming codes and LDPC code as precode, we consider the general definition where any high rate error correcting code can be considered as a precode, and focus on LDPC precoded Raptor codes. We do not consider concatenation with Hamming codes for the precode, because these constructions are specifically designed for the BEC case, and our results prove that with our approach, the use of Hamming codes is not necessary. Moreover, we restricted ourselves to *regular* LDPC precodes because for relatively high rates, regular codes are known to have good thresholds, close to the irregular thresholds. We present in this paper an analysis of a rate splitting approach between an LDPC precode and the fountain, when the two code components are decoded jointly.

A. The rate-splitting issue

In the literature, the rate of the precode is usually chosen very close to 1. Indeed, the optimization of output degree distributions allows designing LT codes such that the fraction of unrecovered input symbols is extremely low. Choosing a very high rate precode is a valid strategy when the Raptor code is decoded sequentially, but could be a suboptimal choice when we consider iterative joint decoding of the precode and the LT code [5]. In this latter case, if the output degree distribution is matched – with proper optimization – to the EXIT chart of a *lower rate* precode, there is almost no asymptotic loss *i.e* no loss in the waterfall region. By *lower rate*, we mean rates that are between $R = 0.9$ and $R = 0.95$, whereas typically in the existing literature, *very high rate* codes, *e.g.* $R = 0.98$, are considered.

Then arises the natural question of the optimal repartition of the overall rate between the LT code and the precode. The use of a lower rate precode can be very attractive in practice, especially for the design of Raptor codes with short or moderate information block lengths. For short to moderate lengths, the topology of the overall Tanner graph in terms of short cycles and subsequent stopping/trapping sets needs to be considered. The problem of using a very high rate LDPC precode is then that is introduces a large number of length-4 cycles, resulting in error floors which are unacceptably high. More precisely, the code length such that an LDPC code of girth 6 exists grows exponentially with the check node degree d_v [6], hence with the code rate. Using a lower rate precode has the main objective of improving the Raptor code in the error floor region for finite block lengths.

We now explain why the use of a lower rate precode does not affect the overall rate of the Raptor code. Let R be the rate of the Raptor code which is the concatenation of an LT code of rate R_{LT}, and a precode of rate R_p. For a channel with capacity C, the optimization for the sequential decoding

[1]This work was supported by the ANR-RNRT under Grant ANR-05-RNRT-019-04

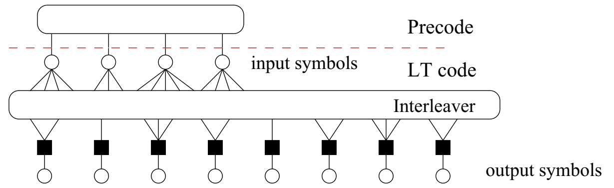

Fig. 1. Description of a Raptor code: Tanner graph of an LT code + precode. The black squares represent the parity check nodes and the circles represent variable nodes associated with input symbols or output symbols.

scheme always gives $R_{LT} < C$. Since $R_p < 1$, the total rate of the precode $R = R_p R_{LT}$ is smaller than R_{LT}, and therefore the rate of the precode R_p appears to be a burden in terms of the total rate of the Raptor code. However, in the case of the optimization for joint decoding the optimized output degree distributions can have a rate $R_{LT} > C$, which allows considering precodes with lower rates *and* still have a total asymptotic rate R close to the capacity.

Therefore, jointly decoded Raptor codes allow to study the problem of the overall rate distribution and its repartition between the LT code and the precode, which is the problem that we address in this paper.

B. Outline of the paper

In section II, we briefly present the construction of Raptor codes to introduce the notations. Then, in sections III and IV, we address the rate splitting issue for two different channels, namely the BEC and the BIAWGNC, and we show that for the design of finite length Raptor codes, it is crucial to use lower rate precodes. For a wide rage of *a priori* fixed precode rates $R_p \in [0.9; 0.9625]$, we design output degree distributions for the LT code using density evolution (DE) and taking into account the information brought by the precode, that is assuming jointly decoded Raptor codes. The case of the BEC is solved with exact density evolution in section III, and the case of the BIAWGNC is solved using a Gaussian approximation (EXIT type) of DE in section IV. In both cases, the asymptotic distributions will be tested for small ($K = 1024$) to moderate ($K = 8192$) block lengths, and simulation results show an impressive gain in the error floor regions for decreasing precode rate.

II. LT CODES AND RAPTOR CODES

We call *input symbols* the set of information symbols to be transmitted and *output symbols* the symbols produced by an LT code from the input symbols. The input symbols are not transmitted over the channel. At the receiver side, belief propagation (BP) decoding is used to recover iteratively the input symbols. An LT code is described by its *output degree distribution* [2]: to generate an output symbol, a degree d is sampled from that distribution, independently from the past samples, and the output symbol is then formed as the sum of a uniformly randomly chosen subset of size d of the input symbols. We will refer to the check nodes connected to the output symbols of the LT code as *dynamic check nodes*.

Let $\Omega_1, \Omega_2, \ldots, \Omega_{d_c}$ be the distribution weights on $1, 2, \ldots, d_c$ so that Ω_d denotes the probability of choosing the value d. We denote the output degree distribution using its generator polynomial: $\Omega(x) = \sum_{i=1}^{d_c} \Omega_i x^i$, which is associated with the corresponding edge degree distribution in the Tanner graph $\omega(x) = \sum_{i=1}^{d_c} \omega_i x^{i-1} = \Omega'(x)/\Omega'(1)$ [3].

Because the input symbols are chosen uniformly at random, their node degree distribution is binomial, and can be approximated by a Poisson distribution with parameter α [4]. Thus, the input symbol node degree distribution is defined as: $I(x) = e^{\alpha(x-1)}$ Then, the associated input symbol edge degree distribution $\iota(x) = I'(x)/I'(1)$ also equals $e^{\alpha(x-1)}$. Both distributions are of mean α.

A Raptor code is an LT code concatenated with an outer code called “precode”, which is a high rate error correcting block code. The input symbols of the LT code are then formed by a codeword of the precode. The Tanner graph of a Raptor code is presented on Fig. 1.

III. JOINTLY DECODED RAPTOR CODES FOR THE BEC

In this section, we derive the asymptotic analysis of the joint decoding of Raptor codes over the BEC. For our study, we use Density Evolution (DE) under the treelike assumption, and use the same system model as in [5]. Density evolution is the main tool for the analysis of graphical codes such as LDPC codes and Raptor codes. DE can be expressed analytically in the case of an erasure channel, and we refer the reader to [7] for a detailed presentation of this technique.

The analysis is presented from the fountain point of view, and we track the evolution the messages that are related to the fountain part of the Tanner graph. Indeed, our objective is to optimize the distribution of the fountain part of the Raptor code, namely $\omega(x)$, taking into account the contribution of the precode through its transfer function.

A. Asymptotic analysis of jointly decoded Raptor codes

1) Density evolution: Because the channel is symmetric, we can assume without loss of generality that the all-zero codeword has been transmitted. In that case, the messages on the edges of the decoding graph are 1 and 0, where the value 0 indicates an erasure: the value is 0 iff the corresponding edge is connected to an input symbol which has not been recovered. When the precode is an LDPC code with data node and check edge distributions $\tilde{\lambda}(x)$ and $\rho(x)$, its extrinsic transfer function [8] is given by:

$$T(x) = 1 - \tilde{\lambda}\big(1 - \rho(x)\big) \tag{1}$$

As done in [5], we assume the reinitialization of the decoder, which is a pessimistic assumption because we under-estimate the information provided by the precode. We assume that the initial messages from check nodes to variable nodes are set to 0, which means that the values of the messages on the LDPC graph are not kept from one global iteration to the next one. Note that this pessimistic assumption is crucial to have a *linear* optimization problem, and nevertheless leads to the design of efficient output degree distributions.

We denote $p^{(l)}$ (resp. $q^{(l)}$) the probability that an edge connecting a dynamic check node to an input symbol (resp. an input symbol to a dynamic check node) carries the value 1 at the l^{th} decoding iteration. We denote by $u^{(l)}$ the extrinsic information passed by the LT code to the precode, at the l^{th} decoding iteration. As the input symbols are of average degree α, we have:

$$v^{(l)} = T(u^{(l)}) = T\big(1 - e^{-\alpha q^{(l-1)}}\big) \tag{2}$$

The extrinsic information passed by the precode to the LT code is then $v^{(l)} = T(u^{(l)})$. When accounting for the transfer function of the precode, the update rules for the messages in the Tanner graph can be written as follows:

$$p^{(l)} = 1 - (1 - v^{(l)})e^{-\alpha q^{(l-1)}} \tag{3}$$

$$1 - q^{(l)} = \omega(p^{(l)}) \tag{4}$$

$$\begin{aligned} q^{(l)} &= F(q^{(l-1)}) \\ &= \omega\bigg(1 - e^{-\alpha q^{(l-1)}}\Big(1 - T\big(1 - e^{-\alpha q^{(l-1)}}\big)\Big)\bigg) \end{aligned} \tag{5}$$

Combining (3), (4) and (2) gives (5), that describes the evolution through one joint decoding iteration of the probability of erasure at the output of the dynamic check nodes. Note that this expression is linear with respect to the coefficients of $\omega(x)$, which is the distribution that we intend to optimize. We point out that (5) is general since it reduces to the classical sequential decoding case by setting the extrinsic transfer function to $x \mapsto T(x) = 0 \quad \forall x \in [0;1]$, thus assuming that no information is propagated from the precode to the fountain.

Following the same approach developed in [5] for the BIAWGNC case, we derive the following lower bounds on ω_1 and ω_2:

Proposition 1: The decoding process can begin iff $q^{(1)} > \varepsilon$, for some arbitrary $\varepsilon > 0$, which gives:

$$\omega_1 > \varepsilon \tag{6}$$

Therefore, one must have $\omega_1 > 0$ for the decoding process to begin, and ε appears to be a design parameter that will constrain the optimization problem,

Proposition 2: For an output degree distribution that is to be capacity achieving, we have:

$$\omega_2 > \frac{1}{\alpha} \tag{7}$$

2) Optimization of output degree distributions: The optimization of an output distribution consists in maximizing the rate of the corresponding LT code, *i.e.* maximizing $\Omega'(1) = \sum_i \Omega_i i$, which is equivalent to minimizing $\sum_i \omega_i/i$. Moreover, according to the previous section, several constraints must be satisfied. Since $\omega(x)$ is a probability distribution, its coefficients must sum up to 1. We call this the proportion constraint $[\mathrm{C}_1]$. Moreover, to ensure the convergence of the iterative process we must have $F(x) > x$. However, this inequality cannot hold for each and every value of x: using the same derivation technique as in [5], it can be shown that the fixed point of $F(.)$ is smaller than 1. Therefore, we must fix a margin $\delta > 0$ away from 1, and then by discretizing $[0; 1-\delta]$ and requiring inequality to hold on the discretization points, we obtain a set of inequalities that need to be satisfied: they define the convergence constraint $[\mathrm{C}_2]$. The starting condition (proposition 1) must also be satisfied and defines the constraint $[\mathrm{C}_3]$. Moreover, the edge proportion of output symbols of degree 2 is lower bounded by proposition 2, defining the constraint $[\mathrm{C}_4]$. Finally, $x \mapsto T(x)$ is defined according to (1) for an LDPC code, or could be estimated with Monte Carlo simulations if another component code is used as a precode. The transfer function $T(\cdot)$ appears in the general density evolution and therefore in constraint $[\mathrm{C}_2]$. For a given value of α, and a given channel parameter σ^2, the cost function and the constraints are linear with respect to the unknown coefficients ω_i. Therefore, the optimization of an output degree distribution can be written as a linear optimization problem that can be efficiently solved with linear programming.

For a given α, the optimization problem can be stated as follows:

$$\omega_{opt}(x) = \arg\min_{\omega(x)} \sum_j \frac{\omega_j}{j} \tag{8}$$

subject to the constraints:

$[\mathrm{C}_1]$ $\sum_i \omega_i = 1$
$[\mathrm{C}_2]$ $F(x) > x \quad \forall x \in [0; 1-\delta]$ for some $\delta > 0$
$[\mathrm{C}_3]$ $\omega_1 > \varepsilon$ for some $\varepsilon > 0$
$[\mathrm{C}_4]$ $\omega_2 > \frac{1}{\alpha}$

If we denote x_p the threshold in terms of mutual information of the LDPC precode, then the convergence of the LT code should be such that at some point of the decoding process, $u^{(l)}$ becomes larger than the precodes threshold x_{p}, *i.e.* δ must be chosen such that $1 - \delta > x_p$.

By applying this optimization procedure for increasing values of α, it appears that there is a value α_{opt} that maximizes the rate R_{LT}. The value α_{opt} gives the optimal distribution that maximizes the overall rate.

B. Simulation results

We optimized output degree distributions for 4 different precodes with different rates. We restricted ourselves to regular LDPC precodes because for high rates, regular codes are known to have good thresholds, close to the irregular thresholds.

Figures 2, 3, 4 and 5 show simulation results for Raptor codes of length K=1024, 2048, 4096 and 8192 respectively. Each figure compares Raptor codes with the following regular LDPC precodes:

- $(d_v, d_c) = (3, 30)$ regular LDPC code of rate R=0.9
- $(d_v, d_c) = (3, 40)$ regular LDPC code of rate R=0.925
- $(d_v, d_c) = (3, 60)$ regular LDPC code of rate R=0.95
- $(d_v, d_c) = (3, 80)$ regular LDPC code of rate R=0.9625

The different LDPC precodes were constructed with a PEG-based algorithm that minimizes the multiplicity of the girth.

For $K = 1024$ input bits (Figure 2), which can be considered as a very small code, only the lowest considered rate

$R = 0.9$ shows good performance. For all other precode rates, the code exhibits an error floor behavior, which can be explained by the following remarks.

If we denote by X-cycle a cycle of length X, the (d_v, d_c) LDPC precodes of size $K = 1024$ had the following properties:

- (3,80) code: 5356 4-cycles, and 716492 6-cycles.
- (3,60) code: 2328 4-cycles and 295760 6-cycles,
- (3,40) code: 90 4-cycles and 83869 6-cycles
- (3,30) code: 31394 6-cycles (no 4-cycles).

Therefore, the expected number of small stopping sets is much higher for the (3,80) code than for the (3,30) code. We insist on the fact that the 4-cycles do not result from a poor construction, but from the fact that it is not possible to construct girth 6 regular $(3, d_c)$ LDPC codes for $d_c > 39$ [6].

Similarly, for $K = 2048$ input bits (Figure 3), the lowest rates $R = 0.9$ and $R = 0.925$ show good performance, whereas the Raptor codes with precodes of higher rates exhibit an error floor behavior. For $K = 4096$ input bits (Figure 4, only the precode of highest rate $R = 0.9625$ exhibit an error floor behavior. In fact, all the curves that exhibit an error floor have a precode with cycles of length 4.

For $K = 8192$, all precodes are of girth 6, and Fig 5 shows that the corresponding Raptor codes have similar performance. This shows that as long as joint optimization using the precode transfer function is performed, a lower rate precode does not impact the performance of the Raptor code in the waterfall region. Note hat none of our simulations show error floors despite the fact that we do not use Hamming codes.

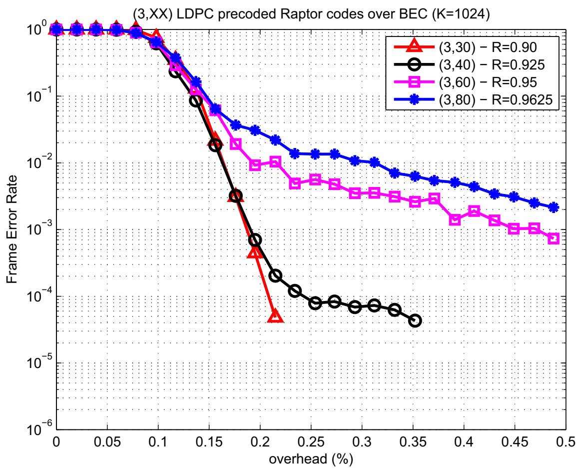

Fig. 2. Performance of LDPC precoded Raptor codes of size K=1024

IV. Design of Raptor codes for the BIAWGNC

In this section, we consider the binary input additive white Gaussian noise channel (BIAWGNC). We show through simulation results that the approach of rate splitting presented in the previous section for the BEC is also efficient for this channel.

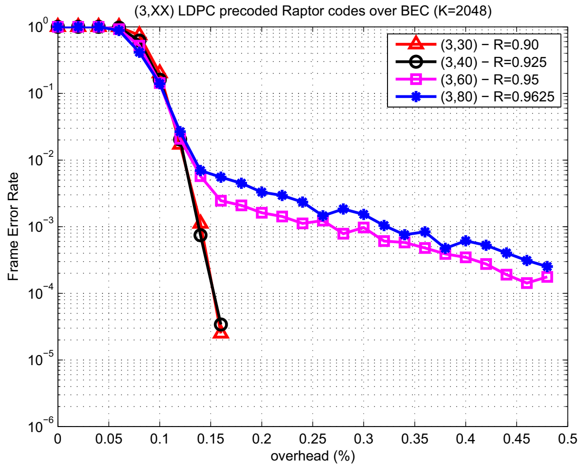

Fig. 3. Performance of LDPC precoded Raptor codes of size K=2048

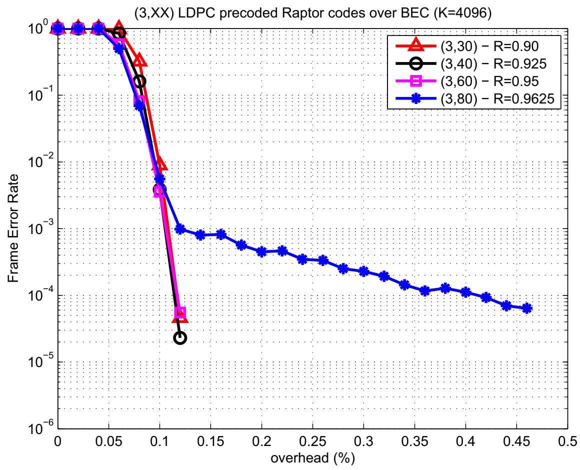

Fig. 4. Performance of LDPC precoded Raptor codes of size K=4096

A. *Asymptotic analysis of LDPC precoded Raptor codes*

In [5], we give an asymptotic analysis of the joint decoding Raptor codes on a BIAWGNC, using Information Content (IC) evolution under Gaussian Approximation (GA) and treelike assumption. The messages on the decoding graph are the log density ratios (LDR) of the probability weights. They are modeled by a random variable which is assumed to be Gaussian distributed with mean m and variance $\sigma^2 = 2m$ [9]. Thus, the density of the messages is symmetric.

Assuming that extrinsic information is exchanged between the precode and the fountain at each decoding iteration, they track the evolution of the IC of the messages that are related to the fountain part of the Tanner graph. We denote $x_u^{(l)}$ (resp. $x_v^{(l)}$) the IC associated to messages on an edge connecting a dynamic check node to an input symbol (resp. an input symbol to a dynamic check node) at the l^{th} decoding iteration. Moreover, we denote by $x_{\text{ext}}^{(l-1)}$ the extrinsic information passed by the LT code to the precode, at the l^{th} decoding

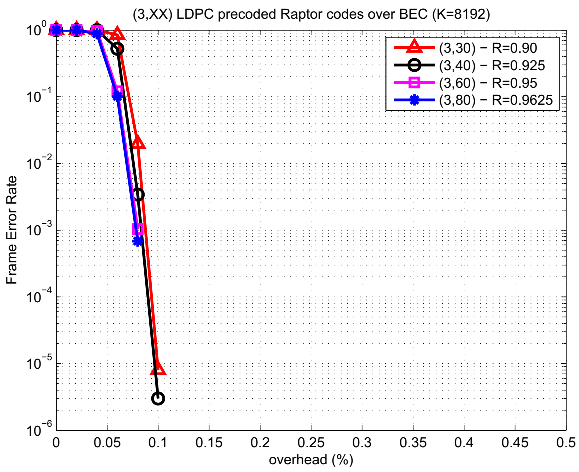

Fig. 5. Performance of LDPC precoded Raptor codes of size K=8192

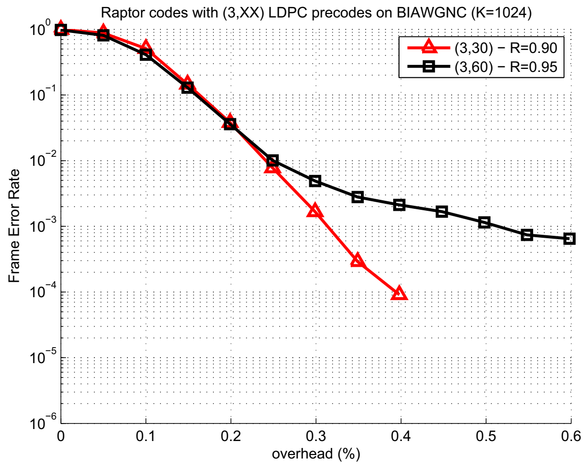

Fig. 6. Performance of LDPC precoded Raptor codes of size K=1024 bits over a BIAWGNC of capacity $C = 0.5$

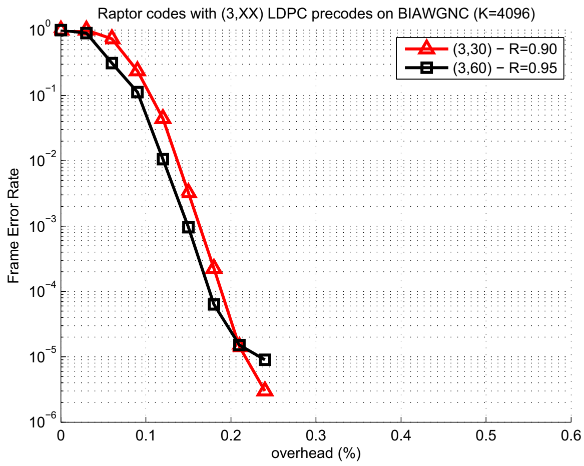

Fig. 7. Performance of LDPC precoded Raptor codes of size K=4096 bits over a BIAWGNC of capacity $C = 0.5$

iteration, and $T(\cdot) : x \mapsto T(x)$ the IC transfer function of the precode. The extrinsic information passed by the precode to the LT code is then $T(x_{\text{ext}}^{(l)})$.

One main result of [5] is the monodimensionnal recursive equation $x_u^{(l)} = F(x_u^{(l-1)}, \sigma^2, T(\cdot))$ that describes the evolution through one joint decoding iteration of the IC of the LDRs at the output of the dynamic check nodes (fountain part). This equation enables to optimize a distribution for a given transfer function *i.e.* for a given precode. The reader is referred to [5] for a more detailed presentation of the analysis and the optimisation method.

B. Simulation results

We optimized for distributions for two different precodes of rates $R = 0.90$ and $R = 0.95$, and compared Raptor codes of size $K = 1024$ and $K = 4096$ input bits, The simulation results are reported on figures 6 and 7 show that, similarly to what has been observed on the BEC, a high rate precode causes an error floor at very short lengths.

V. Conclusion

In this paper, we showed that it is possible to design Raptor codes that perform well at small lengths on an erasure channel, by choosing lower rate precodes and decoding the precode and the LT code *jointly*. Choosing lower rate precodes gives almost no asymptotic loss, and therefore no loss in the waterfall region, but the corresponding Raptor codes do not exhibit an error floor phenomenon at short lengths. Motivated by the results on the erasure channel, we showed that the rate splitting technique remains efficient for the BIAWGN channel.

Lower rate LDPC precodes are less computationally expensive to decode, since complexity is largely due to the average check node degree. Moreover with our approach, it is not necessary to concatenate the precode with a Hamming code.

References

[1] J. W. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "A digital fountain approach to distribution of bulk data," *Proc. of ACM SIGCOMM 98*, pp. 56–67, Sept. 1998.

[2] M. Luby, "LT codes," *Proc. of the 43rd Annual IEEE Symposium on the Foundations of Computer Science (STOC)*, pp. 271–280, 2002.

[3] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inform. Theory*, vol. 52, pp. 2551–2567, June 2006.

[4] O. Etesami and A. Shokrollahi, "Raptor codes on binary memoryless symmetric channels," *IEEE Trans. Inform. Theory*, vol. 52, pp. 2033–2051, May 2006.

[5] A. Venkiah, C. Poulliat, and D. Declercq, "Analysis and design of raptor codes for joint decoding using information content evolution," in *Proc. of the IEEE International Symposium on Information Theory (ISIT), France*, June 2007.

[6] 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, Jan. 2005.

[7] M. Luby, M. Mitzenmacher, and A. Shokrollahi, "Analysis of random processes via and-or tree evaluation," in *Proc of ACM-SIAM*, 1998.

[8] S. T. Brinck, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. on Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.

[9] S.-Y. Chung, T. J. Richardson, and R. Urbanke, "Analysis of sum product decoding of low density parity check codes using a gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 657–670, Feb. 2001.