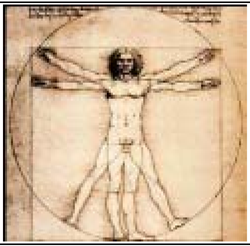

INFSO-ICT-216203 DAVINCI

D4.2

Non Binary root-LDPC codes for the Block Fading Channel

Contractual Date of Delivery to the CEC:	**30 June 2009** *(as specified in the contract)*
Actual Date of Delivery to the CEC:	**16 June 2009**
Author(s):	Iryna Andriyanova, Ezio Biglieri
Participant(s):	ENSEA, WISER
Workpackage:	*WP4*
Estimated person months:	*7*
Security:	**PU**
Nature:	**RP**
Version:	1.0
Total number of pages:	14

Abstract: This report provides a description of the work done in the first part of the project devoted to the code design over non-ergodic transmission channels.
A full-diversity non-binary LDPC structure for two block-fading channels is presented and analyzed. The behavior of the iterative threshold versus channel fading coefficients is established numerically, for both random and root LDPC ensembles. The found dependence is claimed to be general within the ensemble class. Using it, one can perform a semi-numerical approximation of the iterative threshold, by simulating only several values of iterative threshold and by performing the nonlinear curve fitting.
The question of finding the iterative threshold by simulation is also addressed. An EXIT chart method based on the Gaussian approximation is proposed. This method can be applied to random LDPC ensembles in the region of the so called balanced fading.

Keyword list: DA VINCI, deliverable, internal report.

Transmission over Non-ergodic Binary-Input Block-Fading Channels Using Non-binary Root-LDPC Codes

Iryna Andriyanova, Ezio Biglieri

June 16, 2009

Contents

1 Introduction: State of the Art and Project Objectives

1.1 State of the Art

The behaviour of sparse-graph codes over binary-input memoryless symmetric channels is quite well understood, and it has been subject of many research articles. The most relevant of them can be found in [9]. However, when considering an application of sparse-graph codes to wireless communication systems, another channel model, the one of the Rayleigh fading channel, became of great importance. The investigation of sparse-graph codes design for transmission over fading channels has been done and presented in [6].

Recently, the following transmission model has been proposed [5]: a coded data block (codeword) is divided into sub-blocks sent over independent transmission channels. In wireless systems, the channels are assumed to be Rayleigh block-fading (BF). The equivalent transmission channel is non-ergodic. Two important parameters of the systems are *diversity* and *coding gain*. From the point of view of the code design the following question arises: how to design a sparse-graph code ensemble which would *guarantee the full diversity* and would *maximize the coding gain*?

The issue of designing full-diversity sparse-graph codes had been first treated in [5, 4]. First, it has been shown that standard sparse graph codes, applied directly, only give diversity 1, which means that they provide no diversity (the maximum achievable diversity is equal to the number of independent block-fading channels) and are not at all adapted to the considered transmission model. Further, a new family of *structured* LDPC codes, guaranteeing the full diversity over 2 BF channels, has been proposed. This particular structure (called the *root* structure) has been then generalized to k BF channels in [3]. In this case it guarantees the diversity k.

1.2 Objectives

Full-diversity root-LDPC codes mentioned above are based on a standard, unstructured LDPC code ensemble. The choice of the ensemble affects the coding gain which is to be maximized.

The goal of our project is to propose an efficient root-LDPC ensemble having the best coding gain possible, and the accent of the investigation is put on non-binary codes, due to the better parameter flexibility and potentially a better performance under iterative decoding of the same complexity than for binary codes.

It is to be noted that we assume the transmission over *binary-input* block-fading channels, even when using codes over larger alphabets. In this case the alphabet size q is supposed to be a power of 2. Another assumption is that the transmission takes place over only two independent Rayleigh BF channels. Obtained results can be easily generalized then to another number of channels.

Notice that the described transmission model corresponds to a variety of transmission systems as for instance to MIMO systems or wireless networks. The transmission over Rayleigh fading channel with k fading sub-blocks corresponds to the scenario when a large block of coded information (a piece of video sequence for instance) cannot be sent in one transmission block entirely; it is divided into k parts each of them is transmitted through another wireless channel, in series or in parallel.

1.3 Followed Procedure

The project can be divided into three main parts, presented below:

1. Investigation of the behaviour of a code ensemble (root-LDPC or standard LDPC) related to fading coefficients of BF channels:
   (a) numerical approximation of the dependence of the iterative decoding threshold γ^* from fading coefficients α's
   (b) semi-analytical derivation of the approximation and of its parameters
2. Generalization of the root structure to non-binary LDPC codes:
   (a) definition of non-binary root-LDPC codes
   (b) investigation of algorithms to perform the density evolution
3. Optimization of parameters of non-binary root-LDPC codes:
   (a) degree distribution optimization
   (b) optimization of local structures

This report states our first progress in working on the project. It concerns points 1 and 2 of the procedure and is organized as follows. In Section 2 we define a non-binary root-LDPC ensemble and show that it gives full diversity. Section 3 is devoted to the iterative threshold behaviour versus fading coefficients. Section 4 is related to the density evolution for non-binary LDPC codes. Section 5 contains a discussion about the made progress and the future work on the project.

2 Generalization of Root-LDPC Codes to Larger Alphabets

In this section we present the root-LDPC structure for two binary-input BF channels over non-binary alphabets. The binary root-LDPC construction is first reminded. Then it is generalized it to larger alphabets; the conservation of the full-diversity property is also discussed.

2.1 Binary Root-LDPC Codes: Reminder

The construction of binary root LDPC codes of diversity 2 is defined as follows. Given an initial (λ, ρ) LDPC ensemble, one defines a rate-1/2 (λ, ρ) root-LDPC ensemble of diversity 2 through the following set of degree multinomials $\lambda_{root}(x)$ and $\rho_{root}(x)$:

$$\lambda_{root}(x) = \frac{1}{2}\sum_i \left(\frac{\lambda_i}{i}\mu_1 x_1^i + \frac{(i-1)\lambda_i}{i}\mu_1 x_2^i + \lambda_i\mu_1 x_3^i + \lambda_i\mu_2 x_4^i + \frac{(i-1)\lambda_i}{i}\mu_2 x_5^i + \frac{\lambda_i}{i}\mu_2 x_6^i\right), \quad (1)$$

$$\rho_{root}(x) = \frac{1}{2}\sum_i \rho_i \left(x_1 \sum_j \binom{i}{j} f_e^j x_4^j g_e^{i-j} x_5^{i-j} + x_6 \sum_k \binom{i}{k} f_e^k x_3^k g_e^{i-k} x_2^{i-k}\right). \quad (2)$$

In the equations above we have used a multi-edge LDPC notation [8]. The variables μ_1 and μ_2 correspond to two fading channels and the variables $x_1, x_2, \ldots, x_6$ - to 6 edge types in the bipartite graph. Note that x_3 and x_4 are related to "parity" edges, while all the other variables - to "information" edges. The quantities f_e and g_e denote respectively the probability that an edge connected to given check node is the parity or the information one. Clearly, $f_e + g_e = 1$.

The structure of the root ensemble is represented in Fig.1 (picture taken from [3]). The following, equivalent, notation is used:

$$\bar{d}_v = \sum_i \lambda_i/i \qquad \bar{d}_c = \sum_i \rho_i/i \quad (3)$$

$$\mathring{\lambda}(x) = \bar{d}_v \sum_i \frac{\lambda_i}{i} x^{i-1} \qquad \mathring{\rho}(x) = \bar{d}_c \sum_i \frac{\rho_i}{i} x^{i-1} \quad (4)$$

$$\tilde{\lambda}(x) = \frac{\bar{d}_v}{\bar{d}_v - 1} \sum_i \frac{\lambda_i(i-1)}{i} x^{i-2} \qquad \tilde{\rho}(x) = \frac{\bar{d}_c}{\bar{d}_c - 1} \sum_i \frac{\rho_i(i-1)}{i} x^{i-2} \quad (5)$$

2.2 Full-Diversity Root-LDPC codes over $GF(q)$

Now we generalize the root structure to larger alphabets. A possible way to do it is to apply the root construction from Section 2.1 **symbol-wise**. Hence we define the following code ensemble:

Definition 1 *A (λ, ρ) root-LDPC ensemble over $GF(q)$ is the code ensemble for which the related bipartite graph of which has the structure given by Eqns (1) and (2) or equivalently by Fig.1. Moreover this bipartite graph has the following properties:*

- *to each variable node of it, a symbol from $GF(q)$ is assigned;*

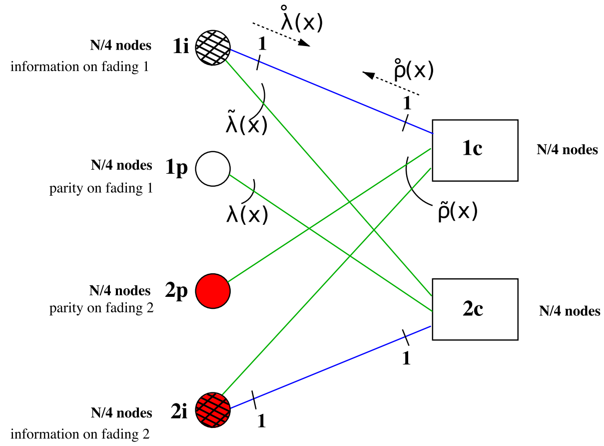

Figure 1: Structure of a (λ, ρ) root-LDPC code ensemble of diversity 2.

- *each check node perform operations over $GF(q)$;*
- *to each edge, the linear bijective transform of form αx, $\alpha \in GF^*(q)$, is assigned.*

Notice that the presented construction gives a code ensemble of rate 1/2, exactly as in the binary case. Also recall that in the context of this project q is assumed to be a power of 2, i.e. $q = 2^m$.

It is easy to see that the following fact holds:

Fact 1 *Under iterative decoding algorithm, the non-binary root-LDPC ensemble defined above has full diversity.*

Proof : The proof of Fact 1 is very similar to the binary case. Symbols within a codeword are divided into two equal sub-words to be transmitted on two independent BF channels with fading coefficients α_1 and α_2. Inside each sub-word we distinguish *information* and *parity* symbols. One needs to show that the full-diversity property holds for information symbols only. For this it is sufficient to notice that, thanks to the root structure, the only outage event occurs when both sub-words are erased by corresponding channels (i.e. $\alpha_1 = \alpha_2 = 0$). Because of the channel independence, this happens with probability p^2, where p is the probability to have $\alpha = 0$ for a BF channel. The power of p is 2, thus the obtained diversity is 2, which is the maximum possible value. □

To summarize all the said above, if one wants to define a non-binary root ensemble of full diversity from a binary full-diversity ensemble, a very simple rule needs to be followed: to obtain the non-binary ensemble, generalize the structure of the corresponding binary root ensemble from bits to symbols over $GF(q)$.

3 Iterative Threshold versus Fading

We consider the following channel model. The transmission takes place with a help of an LDPC ensemble (root or standard one, binary or non-binary). A codeword being divided into two equal sub-words, each of them is transmitted over an independent Rayleigh fading channel. The signal-to-noise ratio (SNR) is denoted by γ. Under the power constraint, $\gamma = 1/\sigma^2$, where σ^2 is the channel variance. The channel fading coefficients are respectively denoted by α_1 and α_2. Note that the channel model is symmetric. At the receiver side, an iterative decoding is performed to recover the transmitted codeword.

Let us denote by $\gamma^*(\alpha_1, \alpha_2)$ the iterative decoding threshold of the considered code ensemble for some fixed values α_1 and α_2. The function $\gamma^*(\alpha_1, \alpha_2)$ for $\alpha_1,\ \alpha_2 \in (0, 1]$ is called the *code boundary* as it separates the region of values (α_1, α_2) for which the outage event happens from the region of non-outage.

Our ultimate goal being to design a code ensemble whose code boundary approaches as close as possible to the *outage capacity boundary* (Fig.2), we want to find a way to approximate $\gamma^*(\alpha_1, \alpha_2)$, analytically or semi-numerically.

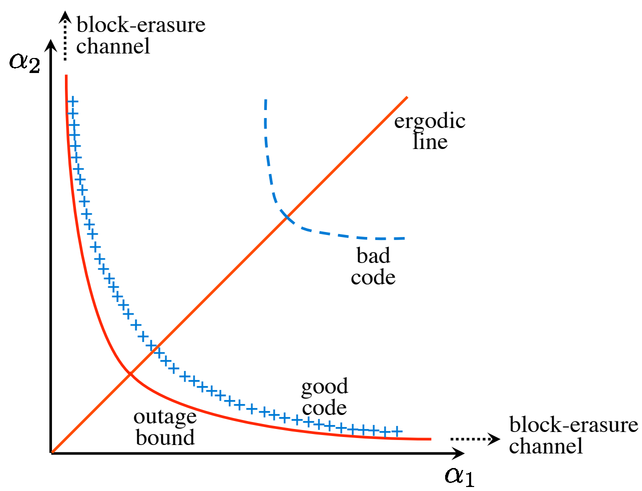

Figure 2: [4] Illustration of code boundaries close and far from the outage capacity boundary (red curve). The straight red line corresponds to the case $\alpha_1 = \alpha_2$ when the channel is ergodic.

3.1 Interesting Relation for $\gamma^*(\alpha_1, \alpha_2)$

Before proceeding further, we establish a very interesting expression for $\gamma^*(\alpha_1, \alpha_2)$:

Lemma 1

$$\gamma^*(\alpha_1, \alpha_2) = \frac{\gamma^*(\alpha_1/\alpha_2, 1)}{\alpha_2^2} = \frac{\gamma^*(1, \alpha_2/\alpha_1)}{\alpha_1^2}.$$

Proof: By direct calculation, the pdf of the LLR channel estimate, corresponding to the threshold,

$$\begin{aligned} p_{\Lambda_0^*(\alpha_1,\alpha_2)} &= \frac{1}{2}\mathcal{N}(2\alpha_1^2\gamma^*, 4\alpha_1^2\gamma^*) + \frac{1}{2}\mathcal{N}(2\alpha_2^2\gamma^*, 4\alpha_2^2\gamma^*) \\ &= \frac{1}{2\alpha_1^2}\mathcal{N}(2\gamma^*, 4\gamma^*) + \frac{1}{2\alpha_1^2}\mathcal{N}(2\frac{\alpha_2^2}{\alpha_1^2}\gamma^*, 4\frac{\alpha_2^2}{\alpha_1^2}\gamma^*) \\ &= \frac{p_{\Lambda_0^*(1,\alpha_2/\alpha_1)}}{\alpha_1^2}. \end{aligned}$$

$p_{\Lambda_0^*(\alpha_1,\alpha_2)} = p_{\Lambda_0^*(\alpha_1/\alpha_2)}/\alpha_2^2$ by symmetry. □

This simple lemma says that, for a given code ensemble, it is sufficient to compute the function $\gamma^*(1,\alpha)$ $(\alpha \in (0,\infty))$ in order to define the iterative threshold for any couple of fading coefficients (α_1,α_2). Since now we denote $\gamma^*(1,\alpha)$ by $\gamma^*(\alpha)$ and investigate its behaviour.

3.2 Numerical Approximation for $\gamma^*(\alpha)$

Intensive numerical simulations of $\gamma^*(\alpha)$ for various code ensembles show that the behaviour of $\gamma^*(\alpha)$ differs much regarding on which ensemble, a random or root one, is considered. Thus we propose two following approximations $\hat{\gamma}^*$, one for each case.
For a random LDPC ensemble,

$$\hat{\gamma}^*(\alpha) = \frac{a(\alpha)}{\alpha^2} + \frac{b(\alpha)}{\alpha} + a(\alpha),$$

where

$$\begin{aligned} a(\alpha) &= K_a e^{-\tau_a \alpha}, \\ b(\alpha) &= K_b(1 - e^{-\tau_b \alpha}). \end{aligned}$$

For a root-LDPC ensemble,

$$\hat{\gamma}^*(\alpha) = \frac{c(\alpha)}{\alpha}, \tag{6}$$

with

$$c(\alpha) = K_a(1 - e^{-\tau_a \alpha}) + K_b e^{-\tau_b \alpha}.$$

The described behaviour of $\gamma^*(\alpha)$ is the same for any alphabet size and is also applicable to non-binary codes. Parameters K_a, K_b, τ_a and τ_b seem to only depend on the code ensemble.

Remarkably, proposed approximations follow closely simulation results for various code ensembles. Moreover, they capture the difference in diversity for random and root ensembles: clearly, the root code boundary lies closer to the outage capacity boundary than the random one.

As an example, in Fig.3 and Fig.4 simulation and approximation results are compared respectively for random and root $(3,6)$ LDPC codes over $GF(2)$. In

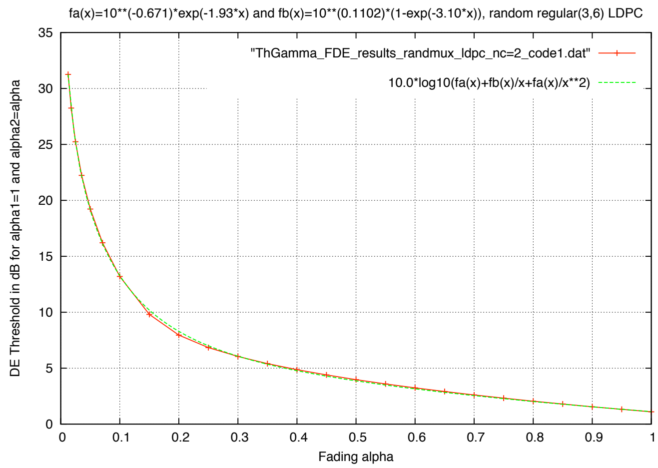

Figure 3: Approximation $\hat{\gamma}^*(\alpha)$ compared with $\gamma^*(\alpha)$ obtained by simulations. The code ensemble - $(3,6)$ random-LDPC.

both cases, a very good match between numerical results and approximations is observed. Estimated parameters for $\hat{\gamma}^*$ are $K_a = 10^{0.11}$, $K_b = 10^{0.65}$, $\tau_a = 18$ and $\tau_b = 18$.

By using the proposed approximations, one needs to simulate at least four points of $\gamma^*(\alpha)$ to approximate γ^* for all values of α by curve fitting for both root and random LDPC ensembles. Given the simulated points, we use the free curve fitting toolbox for Matlab called EzyFit which can be downloaded at *http://www.fast.u-psud.fr/ezyfit/*. As example, Figure 5 presents a result of such an approximation for a random LDPC ensemble over GF(4). We obtain that

$$K_a = 10^{0.11}, \quad K_b = 10^{12.3}, \quad \tau_a = 13.77, \quad \tau_b = 13.77,$$

which gives us quite an accurate prediction of $\gamma^*(\alpha)$ in the region of interest (α small).

3.3 Informal Justification

The derivation of expressions $\hat{\gamma}^*$ will be given in [1]. Here we will only give a brief informal justification of the presented expressions. The diversity order of a code ensemble depends on how the iterative threshold scales with α in the unbalanced regime $\alpha << 1$. For codes with diversity 1 (random ensembles) the scaling is of order $1/\alpha^2$ i.e. $\alpha \sim 1/\sqrt{\gamma^*}$. By putting this expression to the formula of the outage code boundary, we get that P_{out} scales as $1/\gamma^*$ as thus indeed has a diversity 1. Clearly, for full-diversity codes, for which $P_{out} \sim 1/(\gamma^*)^2$, $\alpha \sim 1/\gamma^*$.

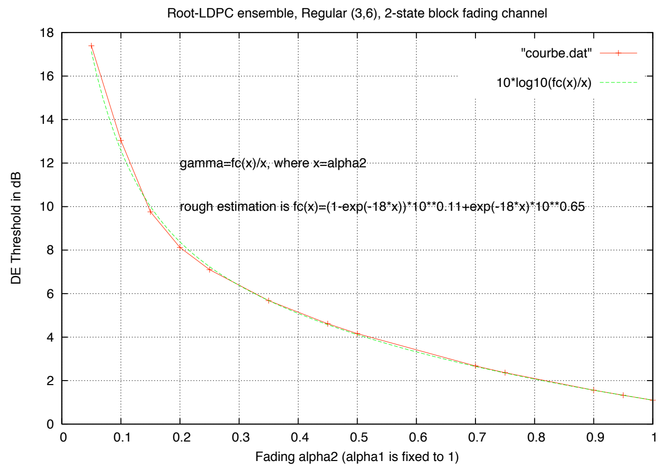

Figure 4: Approximation $\hat{\gamma}^*(\alpha)$ compared with $\gamma^*(\alpha)$ obtained by simulations. The code ensemble - $(3, 6)$ root-LDPC.

4 Density Evolution for Non-Binary Root-LDPC Codes

Note that in order to estimate parameters of the proposed approximations, one needs to simulate several points of $\gamma^*(\alpha)$, i.e. to find iterative decoding thresholds for different couples of fading coefficients $\alpha_1 = 1$ and $\alpha_2 = \alpha$, where α is fixed.

First consider a binary code ensemble. In the special case of $\alpha_1 = \alpha_2 = 1$, the message pdfs, computed during the density evolution process, can be accurately approximated as Gaussians [2], and hence only one parameter is to be tracked. This simplifies a lot the calculation of γ^*. In general, however, the message distributions are not Gaussian, and one tracks the message probability density functions during the density evolution process.

The case of non-binary code ensembles is much more involved. Messages in the iterative decoder are no more scalars, but vectors of length $q-1$, hence their pdfs have to be tracked in the $(q-1)$-dimensional space. Note that the tracking of pdfs, as it was done in the binary case, becomes prohibitively complex. At our knowledge, the issue of performing the density evolution in the general non-binary setting has not been addressed yet. As for the special case $\alpha_1 = \alpha_2 = 1$, a simple one-dimensional EXIT chart method under Gaussian approximation has been proposed in [2]. Unfortunately, [10] suggests that the Gaussian approximation is not accurate. In the recent work [7], the authors generalize the Gaussian approximation to the case of non-binary channel inputs. .

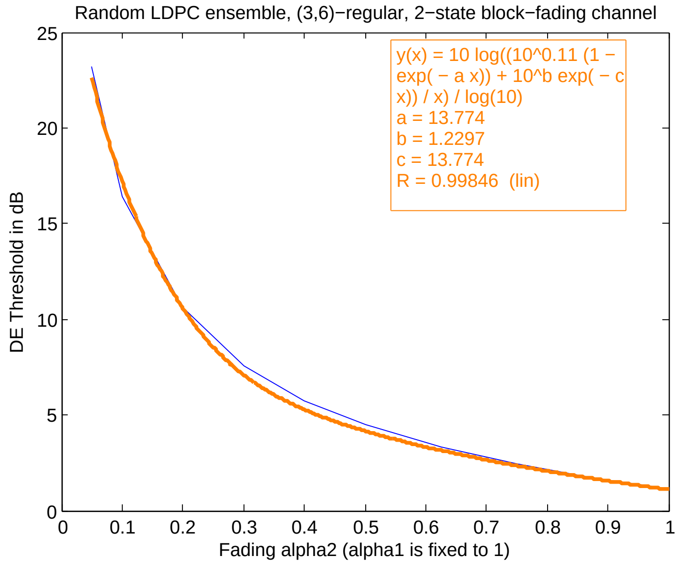

Figure 5: $\gamma^*(\alpha)$ of $(3,6)$ random LDPC codes over $GF(4)$ (blue thin line) approximated by (6) (thick orange line).

4.1 Density Evolution for random LDPC ensembles close to $\alpha_2 = 1$

Notice that we need to estimate $\gamma^*(\alpha)$ either for root or for random LDPC ensemble with the same parameters λ and ρ. We adopt the EXIT chart method mentioned above and, in order to not increase the dimension of the approximation, consider a random LDPC ensemble. Note that, for a random ensemble and for α's close to 1 the Gaussian approximation of input messages is still valid. We use this fact to get a simple estimation of $\gamma^*(\alpha)$. The estimation procedure is the following:

1. Fix α_{init} and a SNR range $[SNR_{min}, SNR_{max}]$. Fix the ensemble parameters $\lambda(x)$ and $\rho(x)$.
2. Put $SNR = 0.5(SNR_{min} + SNR_{max})$.
3. Generate two equal blocks of input samples, for $\alpha = \alpha$ and $\alpha = 1$, and for given SNR and apply the Gaussian approximation. At this step, we approximate the pdf of message Log-Likelihood Ratios (LLRs), which is a mixture of two Gaussians $\mathcal{N}(2SNR, 4SNR)$ and $\mathcal{N}(\in\mathcal{SNR}\alpha, \triangle\mathcal{SNR}\alpha)$, by the equivalent Gaussian distribution.
4. Compute the approximation of EXIT curves for variable and check nodes as described in [2] and check whether they do cross. If yes, put $SNR_{min} = SNR$. Put $SNR_{max} = SNR$ if no.
5. Reiterate. Proceed until $SNR_{min} = SNR_{max}$. Then $SNR^* = SNR_{max}$.

The program which performs such an estimation is provided.

5 Discussion and Further Work

5.1 Outline

In the first part of the project, a class of full-diversity root-LDPC codes over $GF(q)$ has been defined. A fundamental relation between the iterative decoding threshold of a random or root-LDPC ensemble and the fading coefficients α_1 and α_2 has been approximated numerically. The approximation holds both for binary and non-binary code ensembles and allows to obtain the whole code outage boundary simply by simulating from two to four its points and by doing the curve fitting. An analytical derivation of the iterative decoding threshold versus fading is in progress. Moreover, an algorithm to perform the density evolution in the ergodic case has been implemented.

In that way, the points 1a and 2a of the procedure to follow have been fully accomplished, and points 1b and 2b are in progress.

5.2 Provided Software

Together with the report, the following is also provided:

- A toy Matlab example to be used for the curve fitting to approximate the code outage boundary (both for random and root-LDPC codes). Recommendations how to perform the curve fitting with the EzyFit toolbox are given in Appendix A;
- Matlab code of the iterative threshold approximation of a random LDPC ensemble based on the Gaussian approximation. The transmission over two fading channels is implemented.

5.3 Future Work

The future work will be concentrated around three following issues:

Analytical derivation of $\gamma^*(\alpha)$
The goal of this stage is justify the approximation $\hat{\gamma}(\alpha)$, obtained numerically, and to derive the parameters of the approximation through ensemble parameters, namely degree distributions λ and ρ and the ergodic iterative threshold. This work is already in progress.

Density evolution for arbitrary non-binary ensemble and α
An investigation on a more efficient implementation of density evolution for non-binary codes, without Gaussian approximation, is needed to be done. This stage will help to obtain more precise numerical values of iterative decoding threshold for given values of fading coefficients.

Optimization of parameters of non-binary root-LDPC codes
Finally, an optimization of the code parameters and the choice of the best code from the ensemble is to be performed.

References

[1] I. Andriyanova, E. Biglieri, and J. Boutros. Code outage boundaries for random and root LDPC codes over nonergodic block-fading channels. in preparation.

[2] A. Bennatan and D. Burshtein. Design and analysis of nonbinary ldpc codes for arbitrary discrete-memoryless channels. *IEEE Trans. Inform. Theory*, 52(2):549–583, February 2006.

[3] J. Boutros. Diversity and coding gain evolution in graph codes. In *ITA'09*, San-Diego, USA, February 2009.

[4] J. Boutros, Albert Guillen i Fabregas, Ezio Biglieri, and Gilles Zemor. Low-density parity-check codes for nonergodic block-fading channels. submitted to IEEE Trans. Inform. Theory, 2007.

[5] J. Boutros, Albert Guillen i F'abregas, and Emilio Calvanese Strinati. Analysis of coding on non-ergodic block-fading channels. In *Allerton'05*, September 2005.

[6] A. Kavcic, Xiao Ma, and M. Mitzenmacher. Binary intersymbol interference channels: Gallager codes, density evolution, and code performance bounds. *IEEE Trans. Inform. Theory*, 49(7):1636–1652, July 2003.

[7] Ge Li, Ivan J. Fair, and Witold A. Krzymien. Density evolution for non-binary ldpc codes under gaussian approximation. *IEEE Trans. Inform. Theory*, 55(3):997–1015, March 2009.

[8] T. Richardson and R. Urbanke. Multi-edge LDPC codes. submitted to IEEE Trans. Inform. Theory, 2004.

[9] T. Richardson and R. Urbanke. Modern coding theory. Cambridge University Press, 2008.

[10] L. Sassatelli. *Codes LDPC multi-binaires hybrides et mthodes de decodage iteratif.* PhD thesis, Universite de Cergy-Pontoise, October 2008.

A Curve Fitting with EzyFit

- *What fading coefficients to simulate :*
  It is recommended to include in the simulation points the threshold of the ergodic transmission channel, when $\alpha_1 = \alpha_2$, and a point for which $\alpha_1 << \alpha_2$. The former case directly gives us a good approximation of K_a:

$K_a = 10^{0.1\gamma^*_{(\alpha_1,\alpha_2=\alpha_1)}}$, γ^* is given in dB. The latter case ensures a better precision in estimation of K_b, τ_a and τ_b.
Also, the convenient input format is $\gamma^*(\alpha) = \gamma^*(1, \frac{\alpha 1}{\alpha_2})$ (see Section 3.1), so it will be more precise if α_1 is already fixed to 2 in simulations.

- *EzyFit toolbox installation :* The free curve fitting toolbox for Matlab called EzyFit which is to be downloaded from *http://www.fast.u-psud.fr/ezyfit/* by following given instructions. It is a very simple procedure taking no more than one minute.

- *Using EzyFit :* A toy Matlab program is provided. The input format is the vector of α's and the vector of corresponding γ^*'s in dB. The curve fitting procedure is the following. Plot the simulated curve. In the EzyFit menu, choose "Edit User Fits" and add two of them, random LDPC and root LDPC, as follows:

$$\text{r}andomLDPC: \quad 10\log_{10}\left(10^{0.1\gamma^*(1)}e^{-a\alpha}\left(1+\frac{1}{\alpha^2}\right)+\frac{b(1-e^{-c\alpha})}{\alpha}\right)$$

$$\text{r}ootLDPC: \quad 10\log_{10}\left(\frac{10^{0.1\gamma^*(1)}(1-e^{-a\alpha})+be^{-c\alpha}}{\alpha}\right)$$

Choose "Show Fits" and then the fitting formula you want to use, in order to see a result and to estimate $a,\ b,\ c$.