Exit Chart for Iterative Decoding of Product and Concatenated Block Codes

Abderrazak Farchane[1], Mostafa Belkasmi[2] and Ahmed Azouaoui[3]

[1]University Mohamed V SOUISSI, ENSIAS
Rabat, 10000, MOROCCO

[2]University Mohamed V SOUISSI, ENSIAS
Rabat, 10000, MOROCCO

[3]University Mohamed V SOUISSI, ENSIAS
Rabat, 10000, MOROCCO

Abstract

In this paper, we study the convergence behaviour of a new SISO decoder that we have proposed recently in [1]. The exchange of extrinsic information is used to trace the decoding trajectory in the extrinsic information transfer Chart. That allows foreseeing the turbo cliff position. The influence of concatenated scheme as well as different constituent codes on the convergence behaviour is investigated for product and generalized serially and parallel concatenated codes. Simulation result shows that the thresholds obtained by EXIT chart and BER curves are almost the same.

Keywords: *EXIT chart, SISO decoder, iterative decoding, product codes, BCH codes, GSCB codes, GPCB codes.*

1. Introduction

Stephan Ten Brink [2, 3] introduced a useful tool to analyse the convergence behaviour of iterative decoders. This tool is known as EXIT chart (The extrinsic information transfer chart). It allows us analysing the behaviour of iterative decoders that exchange soft information. This process of information exchange is described by a characteristic transfer diagram. The later shows the evolution of the extrinsic mutual information as a function of the a priori mutual information. The EXIT allows determining the $\frac{E_b}{N_0}$ beyond of that we can have a correct decoding. The EXIT chart can be used to design good codes that converge rapidly. It has been developed originally to analyse turbo and turbo-like codes based on convolutional codes. Subsequently, EXIT chart has been extended to LDPC codes. Nevertheless, this technique is not directly applicable to Chase-Pyndiah's iterative decoder [4].

In [1], we have developed a new iterative decoder. It works for product and generalised concatenated block codes. This decoder uses a discrete parameter α. The later will be substituted by another continuous one. It is calculated using interpolation. The continuous parameter allows us applying the EXIT chart technique in order to study the convergence behaviour of product and generalised concatenated block codes. The simulation result shows that the thresholds obtained by the EXIT chart and BER chart are almost the same.
The remaining of this paper is organised as follows: the section two describes our decoder. We present the principle of EXIT chart in the section three. In section four, we introduce the parameter α for EXIT chart. Section five present the EXIT chart of our decoder. Finally, we conclude this paper in section six.

2. Description of our decoder

The decoding of product codes is done by decoding the rows, then the columns of the code matrix. Like turbo codes, it is possible to decode product codes using an iterative process. A reliability must be associated to each symbol. We consider a transmission that use BPSK modulation coded by a block code, with code rate $\frac{k_i}{n_i}$ (i=1 or 2). The input of the decoder, when the channel is perturbed by a white Gaussian noise, is equal to $Y = X + n$, where Y is the observed vector, The binary random variable denotes the transmitted bits with realizations $x \in \{\pm 1\}$; for brevity of notation, we will not distinguish between X and x in the following (only where

needed for clarification) and n is the white noise whose components n_j has zero average and variance σ^2.
The reliability of component using the log-likelihood ratio (LLR) of the received sequence is defined by:

$$r_j = \ln\left(\frac{P_r\left[x=+1/y_j\right]}{P_r\left[x=-1/y_j\right]}\right) = \frac{2}{\sigma^2}y_j \qquad (1)$$

Decoding of rows (or columns) is realized using a list decoding algorithm that lets us to determine the most likelihood codewords. Then, among those codewords, we select the closest codeword to the received sequence R in term of euclidean distance, where $R = (r_1, r_j, ...r_n)$ is the reliability of the received sequence.
The decoder affects a weighting to each component of the decided codeword ,d_j, in order to measure the reliability of each decision. This reliability is evaluated by the logarithm of the likelihood ratio associated to a decision d_j at the output of the decoder and is defined by

$$\Lambda_j = \ln\left(\frac{P_r\left[d_j=+1/R\right]}{P_r\left[d_j=-1/R\right]}\right) \qquad (2)$$

The sign of Λ_j gives the decision d_j and the absolute value of Λ_j is the measure of the reliability of this decision. When the signal to-noise ratio is sufficiently big and the noise is Gaussian, the LLR of (2) can be simplified to this form:

$$\Lambda_j = \frac{1}{2\sigma^2}\left(\|R - C^{min(-1)}\|^2 - \|R - C^{min(+1)}\|^2\right) \quad (3)$$

where the two codewords $C^{min(+1)}$ and $C^{min(-1)}$ having the minimum distance from R and belonging respectively to S_j^{+1} and S_j^{-1}. By introducing the components of the vector R and if we suppose that σ is constant, we can normalize Λ_j with respect to the constant $\frac{2}{\sigma^2}$. We can write the LLR in following form [4]:

$$\begin{aligned}\acute{\Lambda}_j &= \frac{\sigma^2}{2}\Lambda_j \\ &= \frac{1}{4}\left(\|R - C^{min(-1)}\|^2 - \|R - C^{min(+1)}\|^2\right) \qquad (4) \\ &= r_j + e_j\end{aligned}$$

The LLR of a bit is equal to the sum of the reliability of simple r_j in the input of the decoder and a quantity e_j is independent to the reliability of simple r_j. The quantity e_j is analogue to the extrinsic information for the convolutional turbo codes.
In order to determinate the simplified expression of the LLR of a bit in the output of the decoder, it is necessary to determinate the two codewords $C^{min(+1)}$ and $C^{min(-1)}$ having the minimum distance from R and having an opposite sign in position j. For this, we use a list decoding algorithm (like Chase algorithm [5]). It allows us to determine a sub set of codewords among which we can find the searched two codewords.
Sometimes we can not find the two codewords in sub set determined by the list decoding (Chase algorithm for example). This means that all codewords have the same decision on the j^{th} element, d_j of the vector D. They vote for the same candidate. In this case the decision confirms the input decoder. Consequently, the reliability of the decision must be increased while the sign of decision, d_j , is given by the decoder. We propose a formula that can allows computing the reliability of the decision by taking into account the reliability of the decoder input, the sign of the decision. The $\acute{\Lambda}_j$ of the j^{th} element of the decision is given by the following formula:

$$\acute{\Lambda}_j = \frac{1}{2}\sigma_R d_j + |r_j| d_j \qquad (5)$$

where σ_R represents the standard deviation of the decoder input.
The extrinsic information $e_j(p)$ is exploited to modify the input of the next decoder. The reliability of the p^{th} decoder is given by the following formula:

$$r_j(p) = r_j + \alpha(p)e_j(p) \qquad (6)$$

The parameter $\alpha(p)$ is used to control the reliability of the extrinsic information that is unreliable in the first few iterations. However, it became reliable with the iterations.
Using the equations 2 and 4, the extrinsic information e_j is evaluated in the p^{th} decoder by the formula:

$$\begin{aligned}e_j(p) = \frac{1}{4}&\left(\|R(p) - C^c(p)\|^2 - \|R(p) - D(p)\|^2\right) \\ &\times d_j(p) - r_j(p) \qquad (7)\end{aligned}$$

where $r_j(p)$ represents the j^{th} component of the vector $R(p)$ at the input of the p^{th} decoder. $D(p)$ is the decided codeword at the output of the p^{th} decoder. $C^c(p)$ is the competitor codeword such that the symbol at the j^{th} position is opposite to d_j ($c_j^c = -d_j$).
The figure 1 shows the decoding scheme of the product and generalized concatenated codes.

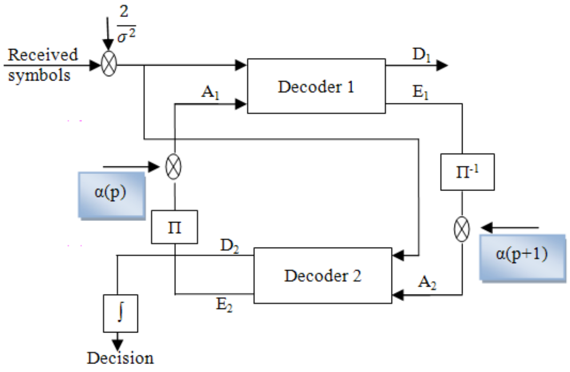

Figure 1: Product and generalized concatenated codes scheme [1]

3. Principle of EXIT chart

EXIT chart is a semi-analytical tools originally developed by S.T. Brink. It allows us to visualise the decoding trajectory of iterative decoders. Its transfer characteristic curves are obtained by plotting the mutual information between the input and the output of the constituent decoders. The iterative decoding of concatenated codes is done by two soft constituent decoders. The mutual information at the output of the first decoder is re-injected in the input of the second decoder. By the same manner the output of the second decoder is connected to the input of the first. It is possible to obtain independently the transfer characteristic of the two decoders. Finally, by combining the results in the same figure we obtain the EXIT chart of the iterative decoder. This technique has the advantage that it doesn't demand time as many as the BER chart. Indeed, by simulation, it has been demonstrated that the a priori returned by the decoder is Gaussian distribution. Consequently, it is unnecessary to consider the whole scheme but we take only one constituent decoder whose a priori is generated by using a Gaussian distribution.

3.1. LLR of the AWGN channel

It is suitable to use the notations of the figure 1. Let A_i denote the a priori information, where $i = 1, or\ 2$, E_i denote the extrinsic information, D_i denote the output of the decoder, and the soft input is denoted by y. For the AWGN channel, the input and output variables are considered to be random variables that are related by

$$Y = X + n \tag{8}$$

where X is a random variable representing the bits x of the message. The bit x can take two values $x = +1\ or\ x = -1$. Let Y be a random variable representing the channel observation, n be a random variable denoting the noise of variance $\sigma^2 = \frac{N_0}{2}$. The logarithm likely-hood ratio, LLR, of the channel is denoted by the following equation:

$$L_c = \log\left(\frac{P(y|x=+1)}{P(y|x=-1)}\right) = \frac{2}{\sigma^2}y \tag{9}$$

where

$$P(y|x) = \frac{1}{\sqrt{2\pi}\sigma}e^{\frac{-(y-x)^2}{2\sigma^2}} \tag{10}$$

the expression (9) allows to determine the random variable Y that represent the received sequence:

$$L_c = R = \frac{2}{\sigma^2}y = \frac{2}{\sigma^2}(x+n) \tag{11}$$

This can also be written as

$$R = \mu_R x + n_R \tag{12}$$

where

$$\mu_R = \frac{2}{\sigma^2} \tag{13}$$

and n_R is a random variable of zero mean and variance:

$$\sigma_R^2 = \frac{4}{\sigma^2} \tag{14}$$

The mean value and the variance of this random variable are related by

$$\mu_R = \frac{\sigma_R^2}{2} \tag{15}$$

This relationship will be useful in the construction of the EXIT chart.

We have assumed that the observation and the a priori have a Gaussian distribution. Consequently, even if the processing done by the decoder isn't linear it's natural to consider that probability of the extrinsic at the output of the constituent decoder is also Gaussian.

3.2. EXIT chart model and mutual information for consistent Gaussian channel

It is useless to constitute the whole decoding scheme. But, we consider the functioning of one constituent decoder or each constituent decoder solely. The a priori is artificially generated. The figure 2 shows the model used to obtain the EXIT chart.

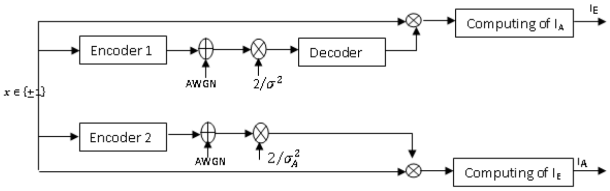

Figure 2: Calculating EXIT chart model.

The prediction of convergence behaviour of iterative decoders can be done by determining the relation between the input and the output of the constituent decoders. For this we use some observation obtained by simulation [6, 7].

1. The a priori A remains uncorrelated with the channel observation for several iterations and wide size interleavers.

2. The probability density function of the extrinsic can be approached by a Gaussian distribution.

These observations are done for the PCC codes in [6], and we have verified it, in section 5, for the GPCB codes using the proposed decoder.

These two observations allow modelling the a priori by:

$$A = \mu_A x + n_A \tag{16}$$

where n_A is a Gaussian random variable of zero mean and variance σ_A^2. The mean μ_A verifies the following relation:

$$\mu_A = \frac{\sigma_A^2}{2} \tag{17}$$

The conditional probability density function of the a priori A is given by:

$$P_A(\xi|X=x) = \frac{1}{\sqrt{2\pi}\sigma_A}e^{-\frac{(\xi-\frac{\sigma_A^2}{2}x)^2}{2\sigma_A^2}} \tag{18}$$

The a priori mutual information $I_A = I(X;A)$, between the variable A and the variable X that represents the systematic bits is used to quantifies the a priori information. The mutual information can be calculated as follow [8, 9]:

$$I_A = \frac{1}{2} \sum_{x=-1,+1} \int_{-\infty}^{+\infty} P_A(\xi|X=x) \times \log_2 \left(\frac{2P_A(\xi|X=x)}{P_A(\xi|X=-1) + P_A(\xi|X=+1)} \right) d\xi \quad (19)$$

where $0 \leq I_A \leq 1$
The expression (19) can be combined with (18) in order to develop the expression of the mutual information as follows:

$$I_A = 1 - \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_A} e^{-\frac{(\xi - \frac{\sigma_A^2}{2} x)^2}{2\sigma_A^2}} \log_2(1 + e^{-\xi}) d\xi \quad (20)$$

For a given value of σ_A, we can compute numerically or by simulation the a priori mutual information, I_A. Figure 3 represents the curve of mutual information I_A as a function of σ_A by using simulation. We can approximate (19) by [10]:

$$I_A \approx 1 - \frac{1}{N} \sum_{i=1}^{i=N} \log_2(1 + e^{-xL_{A_i}}) \quad (21)$$

where N is the size of the transmitted message.

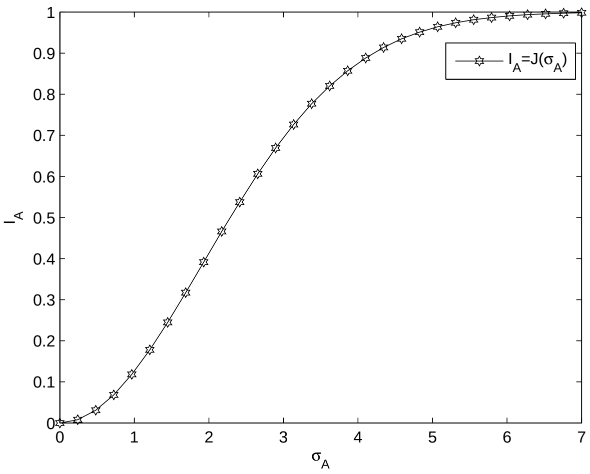

Figure 3: The a priori mutual information, I_A, as a function of σ_A.

Let $J(\sigma_A)$ be the abbreviation of $I_A(\sigma_A)$, we have the following properties.

- $0 \leq J(\sigma_A) \leq 1$
- $\lim_{\sigma_A \to 0} J(\sigma_A) = 0$
- $\lim_{\sigma_A \to +\infty} J(\sigma_A) = 1$

the function $J(\sigma_A)$ is monotonous. So $J(\sigma_A)$ is reversible. Let J^{-1} be the inverse function defined by:

$$\sigma_A = J^{-1}(I_A) \quad (22)$$

Mutual information is used to quantify extrinsic information $I_E = I(X; E)$.

$$I_E = \frac{1}{2} \sum_{x=-1,+1} \int_{-\infty}^{+\infty} P_E(\xi|X=x) \times \log_2 \left(\frac{2P_E(\xi|X=x)}{P_E(\xi|X=-1) + P_E(\xi|X=+1)} \right) d\xi \quad (23)$$

4. The Parameter α for EXIT chart

In the decoding scheme of product codes [4, 11], the a priori information is scaled by a parameter α. The value of the later parameter increases with the iterations. During the first iterations, α limits the effect of the extrinsic information because it is less reliable. Nevertheless, α increases in proportion as the extrinsic information gets improved. Therefore, the values of α have to be harmonized with extrinsic information exchanged between the two constituent decoders.
The EXIT Chart allows us to study the convergence of iterative decoders, by studying only one constituent decoder or each constituent decoder separately. This study is based on the exchange of the extrinsic information between constituent decoders.
In the works [2, 3, 6], the authors study the product codes and concatenated codes. Their decoding scheme consists to inject the extrinsic information without scaling it at the input of the next decoder. However, our decoding scheme [1] and also other schemes in [4, 11] consist to scale extrinsic information by a scaling factor α. It is difficult to study the convergence of product and generalised concatenated block codes using EXIT Chart technique because we haven't the notion of iterations there. Nevertheless, we change the value of the a priori mutual information and follow the evolution of the extrinsic mutual information. Both the a priori and extrinsic information must be scaled by the factor α. So, how can we choose the value of the parameter α that goes with a priori and extrinsic information?
In order to resolve this problem, we have to find a function allowing to calculate the parameter α whatever the a priori information. One solution is to interpolate the discrete value of α by a continual function. This needs to express this parameter as a function of another, like the standard deviation or the mutual information. Once we have a couple of points, the standard deviation and it's corresponding value of α we can determine the interpolation function that interpolates these points. Finally, the interpolation function permits to calculate the value of the parameter α that goes well with the a priori and extrinsic information.
To interpolate the parameter α, we fix the code, for example $BCH(63, 51)^2$, and we take the SNR that gives $BER = 10^{-5}$. The value of the standard deviation of the extrinsic information is used to interpolate α are shown in table 1 : We use the interpolation by pieces to interpolate α. The curve of the figure 4 represents the function that interpolates α as a function of the standard deviation σ_E of extrinsic information.

Table 1: Values of α and its corresponding standard deviation of the extrinsic information

α	0.0	0.13	0.15	0.18	0.2	0.25
σ_E	0.0	3.34	3.61	3.75	3.9	4.04
α	0.3	0.35	0.4	0.45	0.5	0.55
σ_E	4.3	4.65	5.13	5.72	6.34	6.95
α	0.6	0.65	0.7	0.72	0.75	0.77
σ_E	7.59	8.3	9.12	10.08	10.91	11.82
α	0.8	0.82	0.85	0.87		
σ_E	12.70	13.75	14.83	16.14		

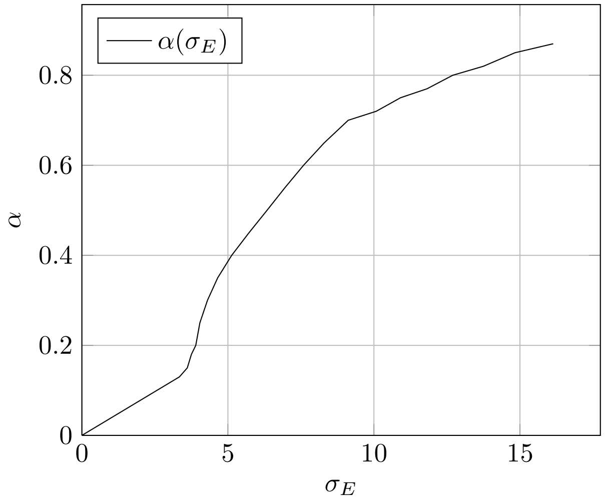

Figure 4: Parameter α as a function of σ_E of extrinsic information for the product code $BCH(63,51)^2$.

To verify the effectiveness of this way (interpolation), we evaluate the performance of the code $BCH(63,51)^2$ using interpolated α. The figure 5 shows the performance of the product code $BCH(63,51)^2$. According to this figure and the result obtained in [1] we observe that the discrete and interpolated parameters are equivalent. Thus, the EXIT chart for product and generalised concatenated block codes will be done by using interpolated α.

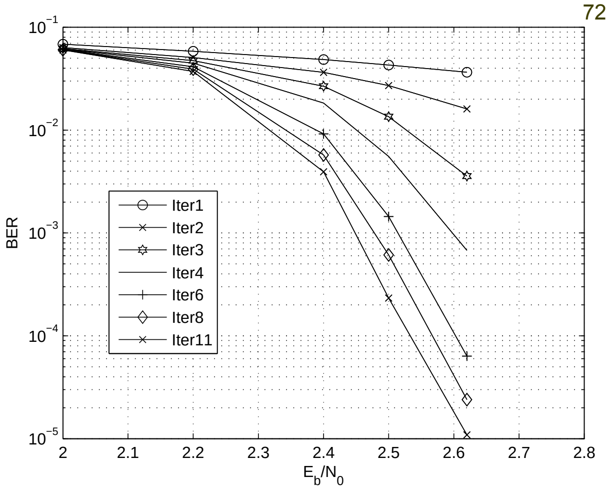

Figure 5: Performance of the decoder in figure 1 for the product code $BCH(63,51)^2$ over AWGN channel, with interpolated α.

5. EXIT chart for our decoder

Monte Carlo simulation allows us to affirm that the values of the a priori information A are independent and uncorrelated with the observations. On the other hand, the probability density function, or more exactly, the histogram of the values of the extrinsic information is Gaussian distribution. It is also true that these values become the a priori values for the next iteration, so that a similar conclusion can be stated for the probability density function of the a priori information. That is to say, the extrinsic and the a priori can be approached by a Gaussian distribution.

The figure 6 presents the non-normalized histogram of the extrinsic information generated by the decoder in [1] for the code $GPCB-BCH(141,113)$ with $M = 500$ [12]. A zero codeword is transmitted. The curves are plotted for different SNR: $\frac{E_b}{N_0} = 3,\ 4\ et\ 5dB$. Increasing of the SNR renders the histogram of the decoder of figure 1 moves towards the left. According to this figure, the histogram can be approached by Gaussian distribution. To make sure that the distribution of the a priori and extrinsic information are Gaussian distribution, we feed the decoder of the figure 2 by a message of length 500 × 113. This message will be coded using the code $GPCB-BCH(141,113)$ with $M = 500$. Then we feed the decoder by a priori information of variance $\sigma_A = 2\ et\ 4.5$. This information is generated according to the equation (16) and (17), with the parameter $\frac{E_b}{N_0} = 1.5dB$. The histograms of the a priori and extrinsic are represented by the figure 7. According to this figure, we remark that the increasing of σ_A lets the histograms move toward the right or the left, depending on the transmitted messages(null vector or full 1 vector).

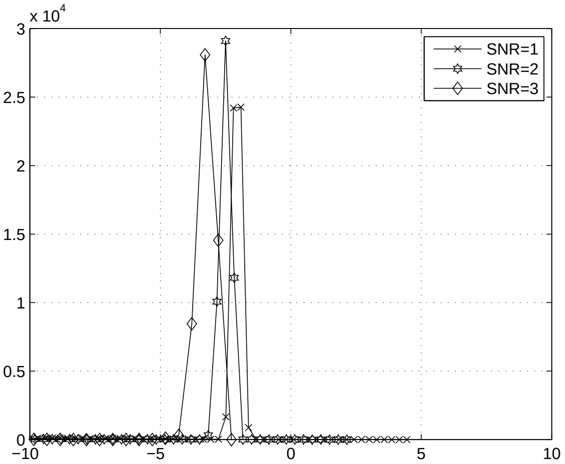

Figure 6: Non-normalized histogram of extrinsic information E at the output of the decoder for $GPCB-BCH(141,113)$ code, with interleaver size equal to 500 × 113.

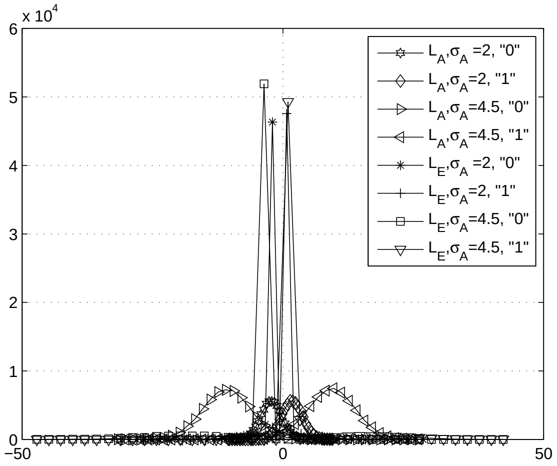

Figure 7: Non-normalized histogram of the a priori information applied to the decoder in figure 1, and the resultant non-normalized histogram of the extrinsic information for the code $GPCB - BCH(141, 113)$ with $M = 500$, over AWGN channel.

5.1. Convergence behaviour analysis of product codes

The EXIT Chart describes the relation between the mutual information of the a priori, I_A, and extrinsic, I_E. Both the a priori and the extrinsic information are measured by using the mutual information between these quantities and the information in the systematic or message bits. The transfer diagram can be obtained by computing, for a given value of I_A and a fixed value of the parameter $\frac{E_b}{N_0}$, the corresponding value of I_E. This computing assumes that the a priori have been generated using the relations (16) and (17), where the probability density function is described by (18). The a priori information A is applied to the constituent decoder, of the figure 2, with a codeword affected by a Gaussian noise according to the value of $\frac{E_b}{N_0}$. The decoder generates extrinsic information E that can be quantified by I_E. The value of the later variable can be obtained by the Monte Carlo method using the relation(23).

The figure 8 shows the transfer function $T(I_A)$, for a set of values of the parameter $\frac{E_b}{N_0}$, for the code $BCH(63, 51)^2$ of rate 0.65. The a priori I_A represents the abscissa axes, and the extrinsic I_E represents the ordinate axes. According to this figure, we remark that the increasing of the value of $\frac{E_b}{N_0}$ involves the increasing of the extrinsic mutual information I_E. The parameter α used to obtain the transfer characteristics is interpolated α. The interpolation of the parameter α, by a continuous function, allows to choose the appropriate value that can goes with the a priori.

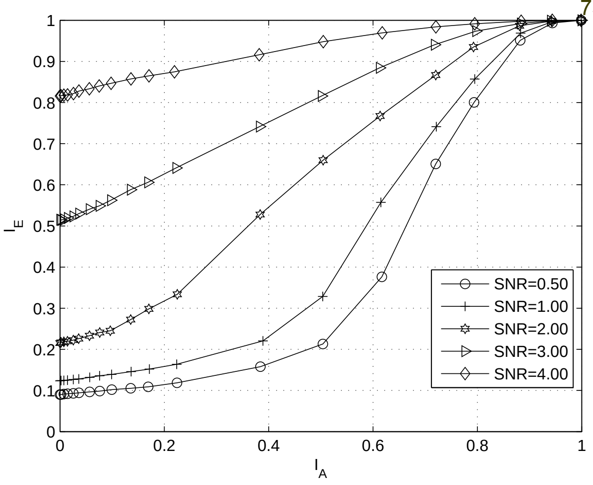

Figure 8: Transfer characteristics of the product code $BCH(63, 51, 5)^2$, with code length 63×63

5.2. Trajectory of iterative decoding

The trajectory of iterative decoding is obtained as follows: Let n denotes the index of iterations and E_b/N_0 a fixed signal to noise ratio. At $n = 0$ the decoder starts with a zero a priori, $I_{A_1,0} = 0$. At the iteration n the extrinsic of the first decoder $I_{E_1,n} = T_1(I_{A_1,n})$ is submitted to the second decoder as a priori $I_{A_2,n} = I_{E_1,n}$. The extrinsic information of the second decoder $I_{E_2,n} = T_2(I_{A_2,n})$ is re-injected in the first decoder as a priori $I_{A_1,n+1} = I_{E_2,n}$. The iterations are stopped when $I_{E_2,n+1} = I_{E_2,n}$, this corresponds to the intersection of the two EXIT charts. The process of iterative decoding designs theoretically a trajectory by projection over the transfer characteristic of the decoder.

To describe the nature of iterative decoding, the characteristic curves of the two iterative decoders must be plotted in the same figure. However, the axes of the transfer characteristic of the second decoder are swapped. The exchange of the mutual information can be seen as a decoding trajectory. The later can be obtained by designing the zigzag over the EXIT chart, it must coincides with the one obtained by simulation.

The figure 9 depicts the EXIT chart of the product code $BCH(63, 51)^2$ for the SNR 0.5, 1.2, 2.0, 2.3, 2.5, and 3.0 dB. Note that in the graphical representations the decoder characteristics are only plotted up to their first intersection. An opening for the trajectory at 2.3dB can clearly be seen, that corresponds to the turbo cliff position in BER chart of the figure 5.

The main contribution of the EXIT chart to the analysis of iterative decoding is the advantage that only simulation on individual decoder is needed to obtain the desired transfer characteristics. No BER simulation of the iterative decoding scheme itself is required. Moreover, for turbo code with the same constituent decoder, transfer characteristics of only one constituent decoder are sufficient to predict the performance of iterative decoding. This further speed up the evaluation of new concatenated codes.

The figure 10 represents the iterative decoding trajectory

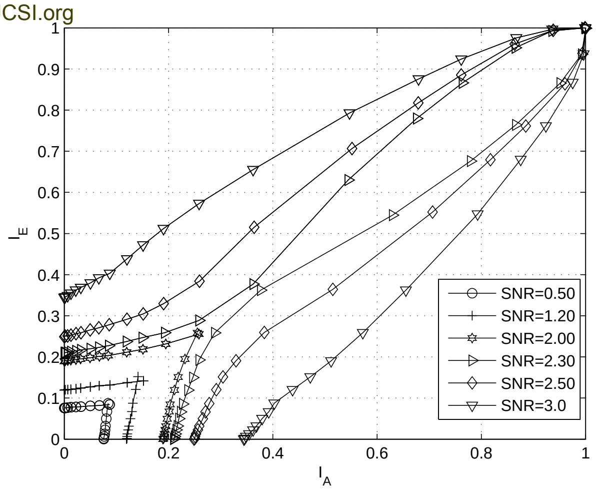

Figure 9: EXIT Chart of our decoder for the product code $BCH(63, 51)^2$, with code length 63×63.

drawn as a zigzag over EXIT chart at the $SNR = 2.5dB$. After three passes through the decoder, increasing correlations of extrinsic information starts to show up and lets the trajectory deviate from its expected zigzag-path.

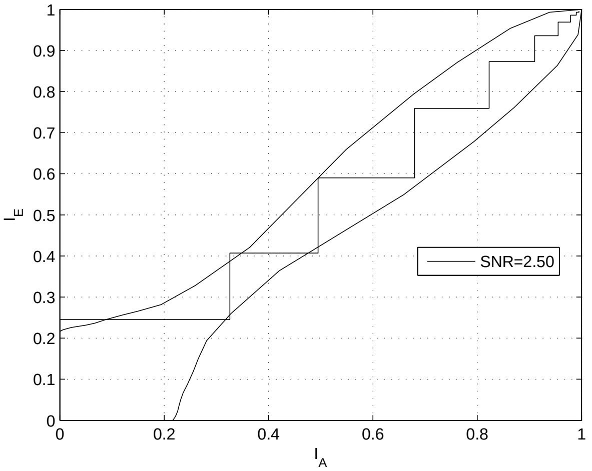

Figure 10: Simulated trajectory of iterative decoding of the product code $BCH(63, 51, 5)^2$ at $SNR = 2.5dB$, with code length 63×63.

The figure 11 shows the EXIT chart of the product code $BCH(127, 113)^2$ for the SNR 1.0, 1.5, 2.2, 2.5, 2.9, 3.0, 3.5, 4.0 and 4.5 dB. For $SNR = 3.0dB$, the decoding trajectory enters in tunnel region close to the bisector. The threshold obtained by the EXIT chart is $0.1dB$ away from the starting point of the waterfall region of the BER curves as shown in the figure 12. The difference between the estimated threshold obtained by EXIT chart and that obtained by BER chart is due to the fact that the size of codeword is finite, and due to the fact of computing α using interpolation.

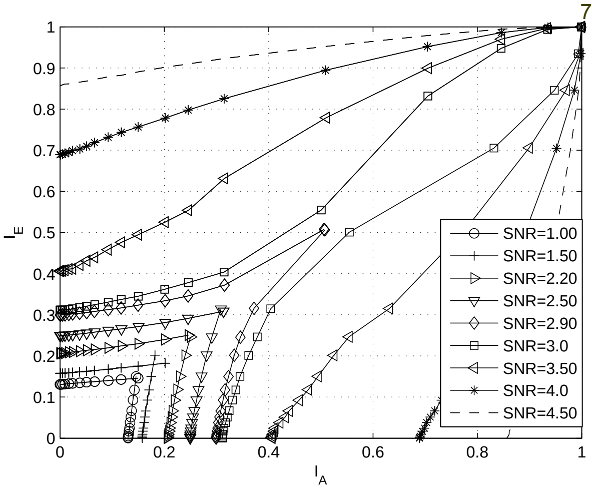

Figure 11: EXIT Chart of our decoder for the product code $BCH(127, 113)^2$, with code length 127×127.

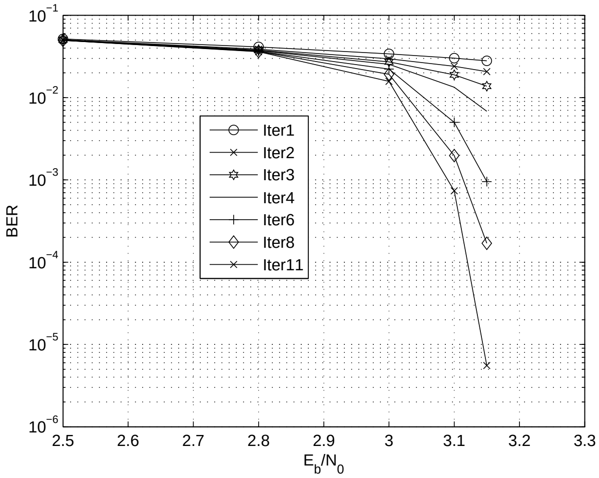

Figure 12: Performance of our decoder for the product code $BCH(127, 113)^2$.

The figure 13 represents the transfer diagram of the product code $BCH(255, 247)^2$ for the SNR 1, 1.5, 3.0, 4.3, 4.6 and 5.0 dB. For $SNR = 4.3dB$, the decoding trajectory enters in tunnel region close to the bisector. The threshold obtained by the EXIT chart is $0.2dB$ away from the starting point of the waterfall region of the BER curves of the figure 14. The difference between the estimated threshold obtained by EXIT and that obtained by BER curves can be explained by the fact that the size of codeword is finite and approximation done by interpolation.

The figure 15 displays the transfer diagram of the product code $BCH(511, 493)^2$ for the SNR 2.0, 3.5, 4.3, 4.5, and 4.7 dB. For $SNR = 4.2dB$, the decoding trajectory enters in tunnel region close to the bisector. The threshold obtained by the EXIT chart is $0.2dB$ away from the starting point of the waterfall region of the BER curves of the figure 16. The difference between the estimated threshold obtained by EXIT and that obtained by BER curves can be explained

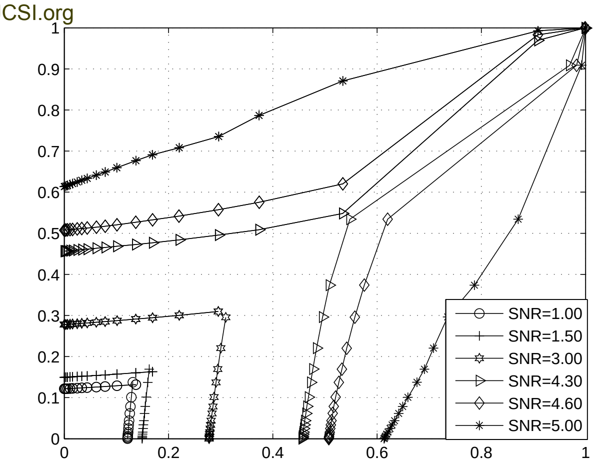

Figure 13: EXIT Chart of our decoder for the product code $BCH(255,247)^2$, with code length 255 × 255.

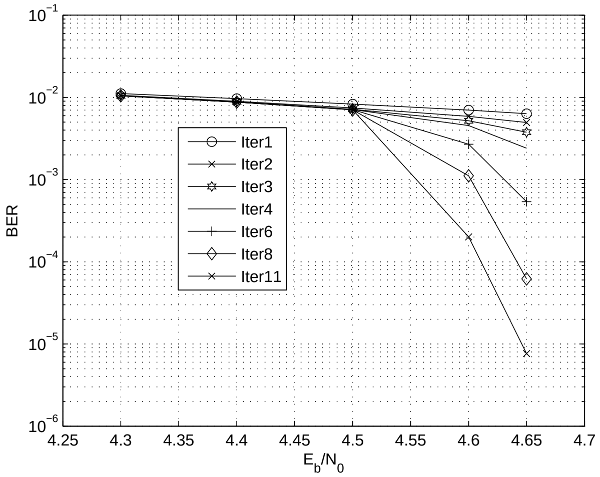

Figure 14: Performance of our decoder for the product code $BCH(255,247)^2$

by the fact that the size of codeword is finite and approximation done by interpolation.

5.3. Convergence behaviour analysis of GPCB codes

The figure 17 shows the transfer diagram of the parallel concatenated code $GPCB-BCH(75,51)$ for the SNR 1.0, 1.5, 2.0, 2.45 and 3.0 dB. For $SNR=2.45dB$, the decoding trajectory enters in tunnel region close to the bisector. The threshold obtained by the EXIT chart is $0.15dB$ away from the starting point of the waterfall region of the BER curves of the figure 18. The difference between the estimated threshold obtained by EXIT and that obtained by BER curves can be explained by the fact that the size of codeword is finite and approximation done by interpolation.

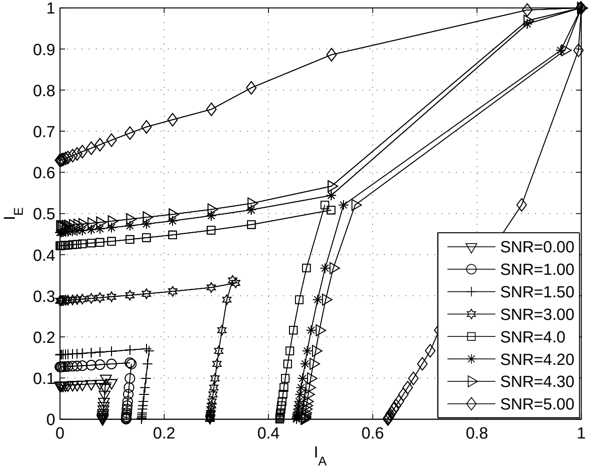

Figure 15: EXIT Chart of our decoder for the product code $BCH(511,493)^2$, with code length 511 × 511.

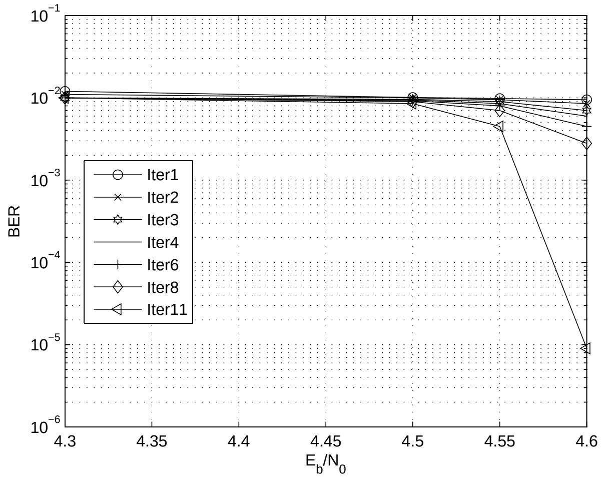

Figure 16: Performance of our decoder for the product code $BCH(511,493)^2$

The figure 19 displays the transfer diagram of the parallel concatenated code $GPCB-BCH(75,51)$ for the SNR 1.0, 2.0, 2.8, 2.9, 3.0, 3.5 and 4.0 dB. For $SNR=2.9dB$, the decoding trajectory enters in tunnel region close to the bisector. The threshold obtained by the EXIT chart coincides with the threshold of the waterfall region of BER chart of the figure 20.

The figure 21 represents the trajectory of the iterative decoding designed over the EXIT chart of the $GPCB-BCH(75,51)$, for the $SNR=2.6dB$. The trajectory has just managed to sneak through the bottleneck. After three passes through the decoder, increasing correlations of extrinsic information starts to show up and lets the trajectory deviate from its expected zigzag-path.

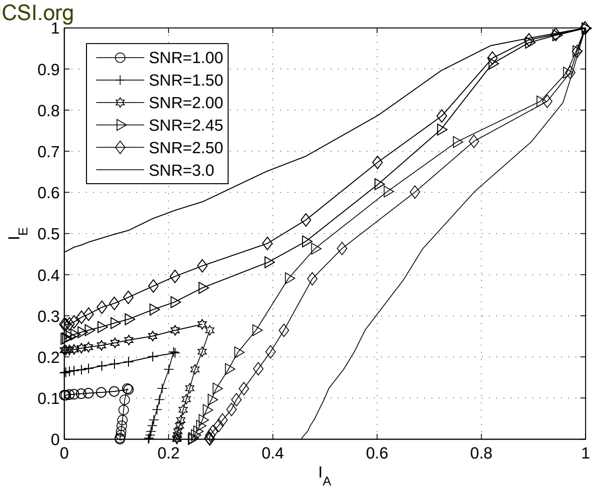

Figure 17: EXIT Chart of our decoder for the parallel concatenated codes $GPCB - BCH(75, 51)$, with M=100 over AWGN channel

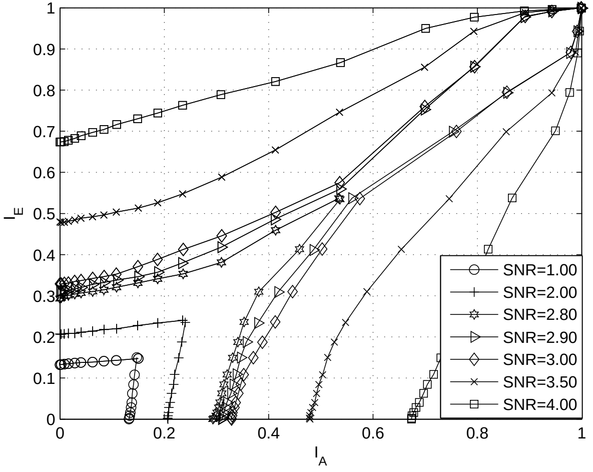

Figure 19: EXIT Chart of our decoder for the parallel concatenated codes $GPCB - BCH(141, 113)$, with M=100, over AWGN channel

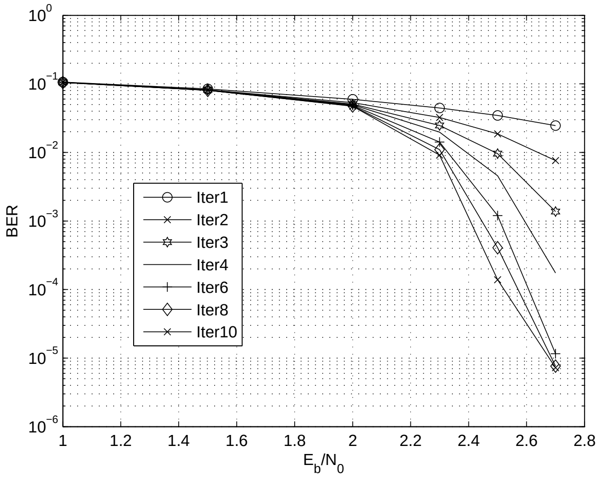

Figure 18: Performance of our decoder for the parallel concatenated code $GPCB - BCH(75, 51)$, M=100.

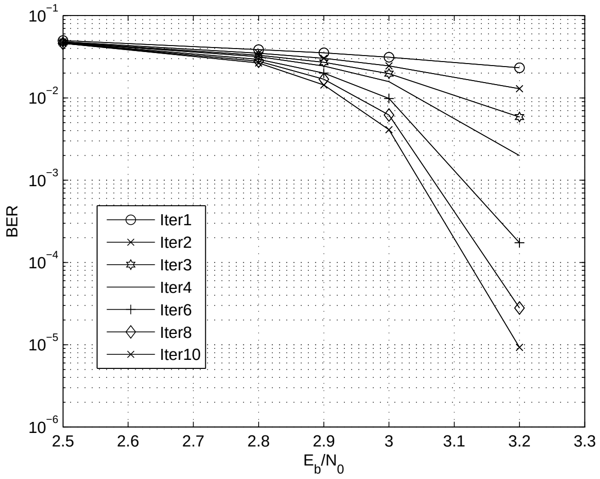

Figure 20: Performance of our decoder for the parallel concatenated code $GPCB - BCH(141, 113)$, with M=100.

5.4. Convergence behaviour analysis of GSCB codes

We have seen that for symmetric product codes and symmetric generalized parallel concatenated code the studying of unique decoder is sufficient to foresee the behaviour of the iterative decoding. Nevertheless, in the case when we have non symmetric elementary decoders the EXIT chart necessitates studying both constituent decoders independently. The figure 22 shows the transfer characteristic of the generalized serially concatenated code GSCB-BCH(63, 39) [13, 1]. According to this figure, the threshold is achieved at $SNR = 2.3dB$.

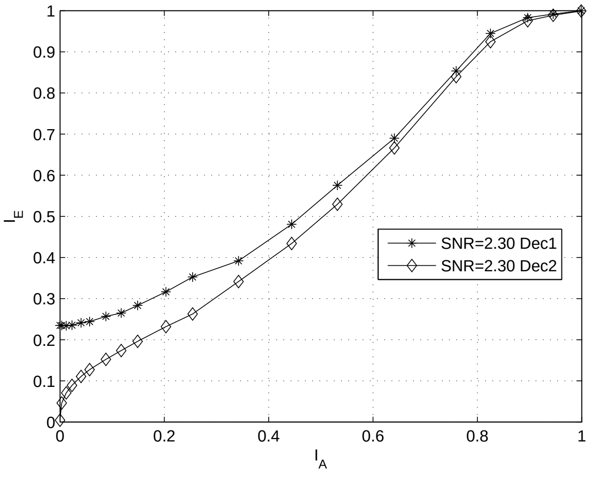

Figure 22: EXIT Chart of our decoder for the serially concatenated code $GSCB - BCH(63, 39)$, with M=100

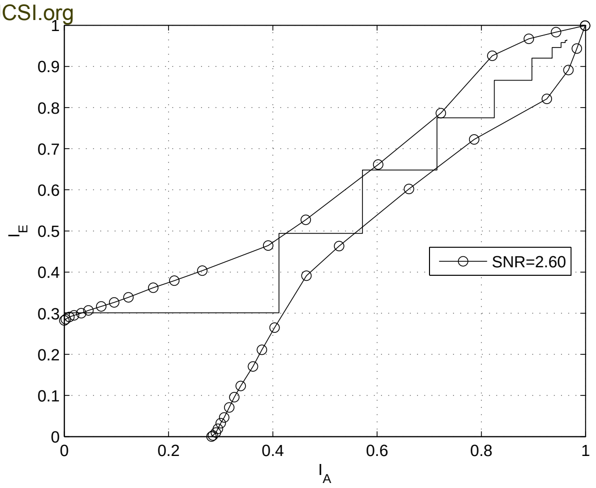

Figure 21: EXIT chart at $SNR = 2.6dB$ of parallel concatenated code $GPCB - BCH(75, 51)$, with M=100

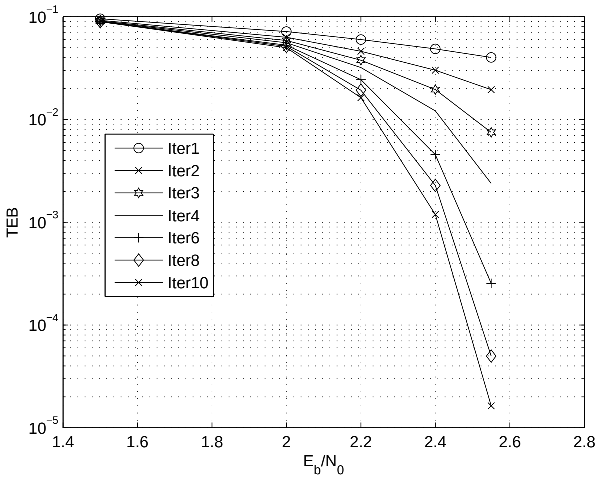

Figure 23: Performance of our decoder for the serially concatenated code $GSCB - BCH(63, 39)$, with $M = 100$.

6. Conclusion

In this work, we have analysed the convergence behaviour of iterative decoder of product and generalized concatenated block codes using the technique called EXIT chart.

According to the results obtained by the BER and EXIT charts, we remark that the thresholds obtained by the two techniques are almost the same. The difference between the thresholds of the EXIT and BER charts may be explained by the fact that the size of codeword is finite and approximation done by the interpolation of the parameter α. Finally, EXIT chart is an alternative technique for designing iterative decoders. This technique can be extended to other codes, like RS codes, either over AWGN or Rayleigh channel.

References

[1] A. Farchane and M. Belkasmi, “New efficient decoder for product and concatenated block codes,” *Journal of Telecommunication*, vol. 12, pp. 17–22, JANUARY 2012.

[2] S. Ten Brink, “Convergence behaviour of iteratively decoded parallel concatenated codes,” *IEEE Trans. Commun.*, vol. 49, pp. 1727–1737, October 2001.

[3] S. T. Brink, “Convergence of iterative decoding,” *Electron. Lett.*, vol. 35, May 1999.

[4] R. Pyndiah, A. Glavieux, A. Picart, and S. Jacq, “Near optimum decoding of product codes,” *GLOBECOM94*, November 1994.

[5] D. Chase, “Class of algorithms for decoding block codes with channel measurement information,” *IEEE Trans. Information theory*, vol. 13, pp. 170–182, Jan 1972.

[6] S. T. Brink, “Iterative decoding trajectories of parallel concatenated codes,” *Proceedings of 3^{rd} IEEE/ITG Conference on Source and Channel Coding, Munich, Germany,*, January 2000.

[7] S. T. Brink, *Design of Concatenated Coding Schemes based on Iterative Decoding Convergence*. PhD thesis, Institute of Telecommunications, University of Stuttgart, 2001.

[8] R. W. Hamming, “Coding and information theory,” *Prentice Hall, New Jersey*, 1986.

[9] N. F. Kiyani and J. H. Weber, “Exit chart analysis of iterative demodulation and decoding of MPSK constellations with signal space diversity,” *Journal of Communications*, vol. 3, pp. 43–50, JULY 2008.

[10] J. Hagenauer, “The exit chart- introduction to extrinsic information transfer in iterative processing,” *EUSIPCO*, 2004.

[11] A. Picart and R. Pyndiah, “Adapted iterative decoding of product codes,” *Global Telecommunications Conference GlobCom*, 1999.

[12] A. Farchane, M. Belkasmi, and S. Nouh, “Generalized parallel concatenated block codes based on BCH and RS codes: construction and Iterative decoding,” *Journal of Telecommunication*, vol. 12, pp. 1–9, JANUARY 2012.

[13] A. Farchane and M. Belkasmi, “Generalized serially concatenated codes: construction and iterative decoding,” *International Journal of Mathematical and Computer Sciences*, 2010.

Abderrazak Farchane received his license in Computer Science and Engineering in June-2001 and Master in Computer Science and telecommunication from University of Mohammed V - Agdal, Rabat, Morocco in 2003. recently, he obtained his PhD in Computer Science and Engineering at ENSIAS, Rabat, Morocco. His areas of interest are Information and Coding Theory.