Journal of Computer and Communications, 2016, 4, 98-107
http://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

Convergence Analysis of Iterative Threshold Decoding Process

Mohammed Lahmer[1], Mostafa Belkasmi[2], Abdallah Rhattoy[1]

[1]Department of Computer Science, High School of Technologies, Meknes, Morocco
[2]Department of Network and Telecoms, ENSIAS, Rabat, Morocco
Email: Mohammed.lahmer@gmail.com

How to cite this paper: Lahmer, M., Belkasmi, M. and Rhattoy, A. (2016) Convergence Analysis of Iterative Threshold Decoding Process. *Journal of Computer and Communications*, **4**, 98-107.
http://dx.doi.org/10.4236/jcc.2016.414008

Received: August 26, 2016
Accepted: November 25, 2016
Published: November 28, 2016

Abstract

Today the error correcting codes are present in all the telecom standards, in particular the low density parity check (LDPC) codes. The choice of a good code for a given network is essentially linked to the decoding performance obtained by the bit error rate (BER) curves. This approach requires a significant simulation time proportional to the length of the code, to overcome this problem Exit chart was introduced, as a fast technique to predict the performance of a particular class of codes called Turbo codes. In this paper, we success to apply Exit chart to analyze convergence behavior of iterative threshold decoding of one step majority logic decodable (OSMLD) codes. The iterative decoding process uses a soft-input soft-output threshold decoding algorithm as component decoder. Simulation results for iterative decoding of simple and concatenated codes transmitted over a Gaussian channel have shown that the thresholds obtained are a good indicator of the Bit Error Rate (BER) curves.

Keywords

OSMLD, EXIT, BER, Mutual Information, Iterative Threshold Decoding Extrinsic Information, LLR

1. Introduction

Since its introduction, the iterative threshold decoding algorithm (ITDA) [1] of one step majority logic decodable codes has gained great interest [2] [3] [4] and this is for several reasons. In fact, OSMLD codes are a subclass of LDPC codes currently present in most telecoms standards, but can be decoded iteratively by ITDA which is less complex than belief propagation algorithm with approximatively the same performances. Getting the performance of these codes is carried out by simulation to evaluate the evolution of the BER compared to the SNR. However, this operation is very expensive in terms of time, especially for long codes. To address this problem, different techniques have been proposed in the literature to analyze the iterative decoding of concatenated codes [5] [6] [7]. The most used method is based on mutual information known as the Exit (Extrinsic Information Transfer) Chart. Indeed, it has been shown by simulation (case of a frame of large size) [8] that the model used by the extrinsic exchanged between the decoders is Gaussian. Moreover, the study of a single decoder is sufficient to predict the iterative decoding performances.

All work on the Exit chart have focused around the iterative decoding of convolution codes, however, in this paper, we propose to apply it to analyze the convergence behavior of the iterative decoding threshold of one step majority decodable codes. Our work is distinguished by the fact that the codes used are structural block codes completely different from that of convolutional codes and that are decoded by our iterative decoding algorithm [1] [2].

The organization of this article is as follows: Section 2 describes the iterative threshold decoding process. In Section 3, we begin with the presentation of the principle of calculating the Exit Chart and end with behavior analysis of convergence of ITDA when applied in both simple and concatenated OSMLD codes. Section 4 provides some conclusions.

2. Iterative Threshold Decoding Process

In this section a brief description of our proposed iterative decoding algorithm [1] will be given. We will focus in particular on variables that will serve us for the calculation in the Exit chart.

2.1. Soft-In Soft-Out Threshold Decoder

First, Let us consider a transmission of code word C (n, k, d) of binary symbols {0, 1} using a BPSK modulation over AWGN channel, the decoder soft output for the j^{th} bit position of a given soft input R ($r_1, r_2 \cdots r_n$) is defined as:

$$LLR_j = \underbrace{\frac{4E_s}{N_0} * r_j}_{\text{Soft in}} + \underbrace{Le_j}_{\text{Extrinsic Information}} \tag{1}$$

$$Le_j = \sum_{i=1}^{J} \ln\left[\frac{1+\prod_{k=1}^{n_i}\tanh\left(L_{ik}/2\right)}{1-\prod_{k=1}^{n_i}\tanh\left(L_{ik}/2\right)}\right] \tag{2}$$

where J is the number of orthogonal parity check equations in code C, n_i represents the number of elements in the i^{th} orthogonal equation, ik represents the k^{th} element of the i^{th} parity check equation orthogonal to the j^{th} bit and $L_{ik} = \frac{4E_s}{N_0}\left|r_{ik}\right|$.

The computation of extrinsic information can be approximated by replacing the formula (2) by:

$$Le_j = \min\left\{L_{ik}\right\}_{k=1\cdots n_i} \tag{3}$$

Which is a good approximation since it requires only signal adders and comparators and can be most easily implemented without undue deterioration in error performance. The algorithmic structure of the SISO threshold decoding can be summarized as follows:

For each bit $J = 1, \cdots, n$
Compute the extrinsic information Le_j **as (3)**
Compute the Soft-input
Compute de Soft-output: $LLR_j = (4Es/N0)\ r_j + Le_j$
Correct de bit r_j

Table 1 shows some examples of OSMLD codes that can be decoded by theses codes are also known as self orthogonal codes. In this table we used the abbreviations DSC for Difference Set Cyclic codes, EG for Euclidean Geometry codes and BCH for Bose Chaudhuri and Hocquenghem codes. For an extensive description of projective geometry codes and Euclidean geometry can be found in [9].

2.2. Iterative Threshold Decoding Process

The Iterative decoding process (see Figure 1) can be described as follows: In the first iteration, the decoder only uses the channel output as input, and generates extrinsic information for each symbol. In subsequent iterations, a combination of extrinsic information and channel output is used as input.

The soft input respectively the soft output of the q^{th} iteration is given by:

$$R(q) = R + \alpha(q) L_e(q) \tag{4}$$

$$LLR(q) = \frac{4E_s}{N_0} R(q) + L_e(q+1) \tag{5}$$

where R represent the received data, $L_e(q)$ is the extrinsic information computed by the previous iteration. In our procedure we use a fixed value $1/J$ for the pparameter $\alpha(q)$ and this for all iterations. The value chosen for $\alpha(q)$ reacts as if one took an average of all J estimators which contribute in the computation of Le.

Our iterative decoding process can also be applied to the concatenated codes; in this case R is a matrix whose rows are constructed from the C^1 code and columns from C^2

Table 1. Set of OSMLD Codes.

n	*k*	*J*	d_{min}	*Rate*	*code*
7	3	3	4	0.42	DSC
21	11	5	6	0.52	DSC
63	37	8	9	0.58	EG
73	45	9	10	0.61	DSC
255	175	16	17	0.68	EG
273	191	17	18	0.69	DSC
1057	813	33	34	0.76	DSC
4161	3431	65	66	0.82	DSC

code. Extrinsic information computed by the SISO decoder of code C^1 is given as a priori information to the second SISO decoder of code C^2. **Table 2** shows some code built from the concatenation of OSMLD codes.

We wish to emphasize that our algorithm can be applied to simple majority decoding codes, product codes (PC) or parallel (PCB) [2], serial concatenated codes [4] and quasi-cyclic codes [3]. Indeed in [10] we showed that our algorithm has a good compromise between complexity and performance compared to the belief propagation algorithm (BP). For example, for the code DSC (273, 191) our algorithm is 5 times faster than BP. **Figure 2** depicts the performance of iterative decoding of (1057, 813) DSC code with rate 0.69 on AWGN channel. We can see that the performance improves with each iteration and no improvement after 15th iteration, at this stage we can said that the decoding process has converged. The number of iterations for convergence as well as the SNR at which begins the enhancement (Point of separation of the different curves) will be predicted by the Exit chart.

3. Convergence Behavior Analysis

3.1. Exit Chart Computation

EXIT chart [5] [6] [7] [8] has emerged as a useful tool for analyzing convergence behavior of iterative decoding of both Turbo codes and LDPC codes. Theses diagrams propose a graphic visualization of the evolution of the quality of the extrinsic information exchanged between two SISO decoders (Turbo codes) or between variable nodes and check nodes in case of LDPC codes. We succeed to apply EXIT charts technique to our

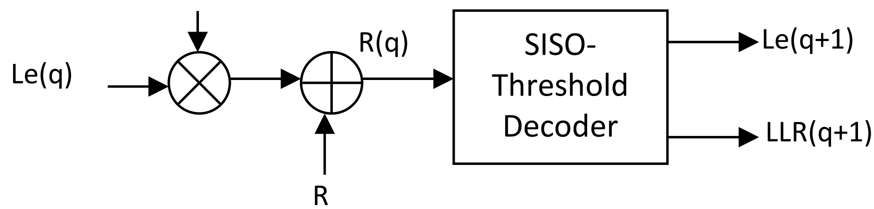

Figure 1. The block diagram of the q^{th} iteration.

Table 2. Set of concatenated codes.

Concatenated Code	Code (C^1)	Code (C^2)	Rate
PC (441, 121)	DSC (21, 11)	DSC (21, 21)	0.27
PC (1533, 495)	DSC (21, 11)	DSC (73, 45)	0.32
PC (4095, 1337)	BCH (15, 7)	DSC (273, 191)	0.32
PC (3969, 1369)	EG (63, 37)	EG (63, 37)	0,34
PC (5329, 2025)	DSC (73, 45)	DSC (73, 45)	0.37
PC (19929, 8595)	DSC (73, 45)	DSC (273, 191)	0.43
PCB (17633, 8595)	DSC (73, 45)	DSC (273, 191)	0.48
PC (74529, 36481)	DSC (273, 191)	DSC (273, 191)	0.49

iterative decoding process while only one SISO decoder is used. The evolution of extrinsic information at the output of our SISO decoder is visualized during iterations. Figure 3 represents the simulation model to be implemented for calculation of EXIT chart. Let X denote the binary input to an additive white Gaussian noise AWGN channel, $X = \{\pm 1\}$, and L_{CH} its corresponding output expressed as a log-likelihood ratio *LLR*. The soft-input/soft-output (SISO) decoder receives as its input both L_{CH} and a priori information L_A, which is also expressed as a *LLR*.

For purposes of the analysis, it is assumed that the sequences $\{L_A\}$ and $\{L_{CH}\}$ at the input to the decoder are infinitely long. One can easily show that $\{L_{CH}\}$ Gaussian (conditioned on X) with parameters $(2/\sigma^2, 4/\sigma^2)$ where $\sigma^2 = N_0/2Es$. Under the assumption that the a priori *LLR* $\{L_A\}$ follows a Gaussian distribution we have $L_A = \mu X + \eta$ where $\mu = \sigma_A^2/2$ and η are a Gaussian variable of zero mean and variance σ_A^2. The mutual information $I_A = I(X; A)$ between X and A which can be expressed as:

$$I_A(\sigma_A) = 1 - \int_{-\infty}^{+\infty} \frac{e^{-\left(\left(\xi - \cdot \sigma_A^2/2\right)^2\right)/2\sigma_A^2}}{\sqrt{2\pi}\sigma_A} \mathrm{Log}_2\left(1 + e^{-\xi}\right) d\xi \tag{6}$$

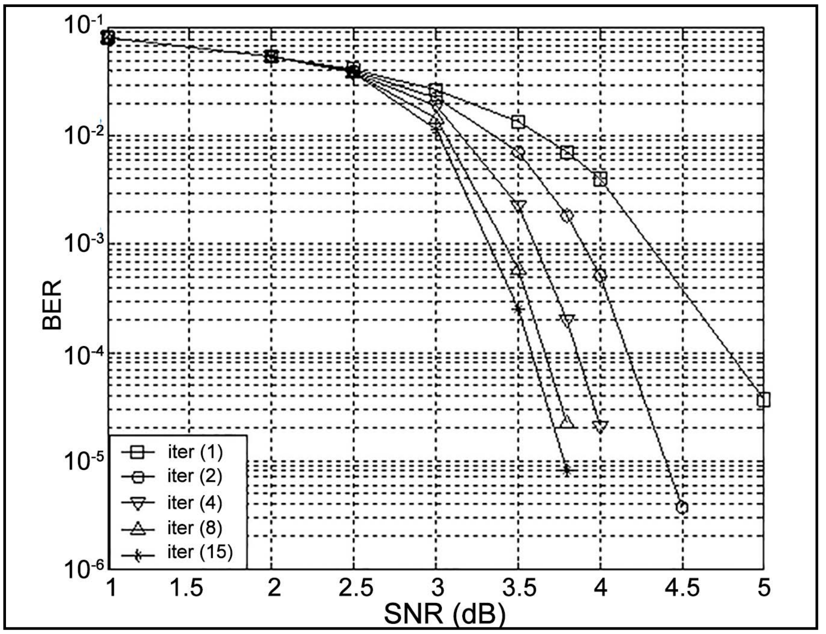

Figure 2. BER performance of (1057, 813) code on AWGN channel.

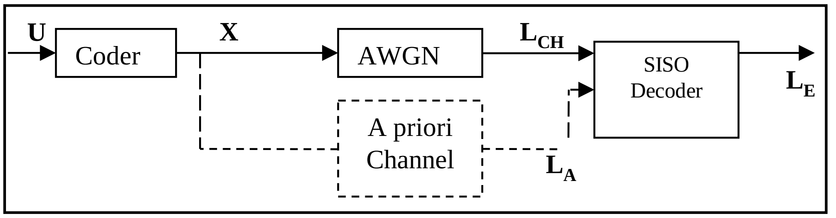

Figure 3. Exit chart computing model.

We put

$$\sigma_A = J^{-1}(I_A) \tag{7}$$

where $\lim_{\sigma\to 0} J = 0$ and $\lim_{\sigma\to 0} J = 1$. The function J is monotonically increasing and can be calculated numerically only once because J and J^{-1} are the same for all codes.

On the other hand the mutual information between the extrinsic *LLR* of the I_E and X, $I_E = I(X; E)$ can be expressed as:

On the other hand the mutual information between the extrinsic *LLR* of the I_E and X, $I_E = I(X; E)$ can be expressed as:

$$I_E = 1 - \int_{-\infty}^{+\infty} p_E(\xi / X = 1) \log\left(1 + e^{-\xi}\right) d\xi \tag{8}$$

It is impossible to express this relation in analytical form; the distributions p_E is most conveniently determined by means of Monte Carlo simulation. Viewing I_E as a function of I_A and the signal to noise ratio SNR value, the extrinsic information transfer characteristics is defined as:

$$I_E = T(I_A, \text{SNR}) \tag{9}$$

For a concatenated constructed from C^1 (n1, k1) and C^2 (n2, k2) and a fixed SNR, the different steps of calculation can be summarized as follow:

Initialize a word w of size (k1 × k2) with bit 0
Initialize the maximum of iteration at Max
Modulate w with Binary phase shift key (BPSK)
Transmit de code word over an AWGN channel
Do
Generate L_A for columns (AWGN/σA) and compute I_A
Decode Lines of the received word and compute I_E
σA = σA + (σA/Max)
Iteration = Iteration + 1
While (iteration! = Max)

3.2. Convergence Analysis of ITD

In order to guarantee a good approximation of the graphical output, it is convenient to work with blocks of size of the order of 10^5. However, the approximation of the distribution of L_E by a Gaussian variable is checked in the case of finite length codes (**Figure 4**). This assumption will allow us to validate curves Exits reports rated to the BER curves.

Figure 4 shows the evolution of the distribution of the extrinsic information at the output of the SISO threshold decoder of PC code (74529, 36481). We simulated the transmission of a frame without any approximation.

Transfer characteristics $I_E = T(I_A, SNR)$ for different SNR are shown in **Figure 5**. The a priori input I_A is on the abscissa, the extrinsic output IE is on the ordinate. The SISO threshold decoding algorithm is applied to (819, 447) OSMLD code of rate 0.54. The numbers on different curves are the *SNR* values in dB. The obtained diagrams look like those of Turbo code.

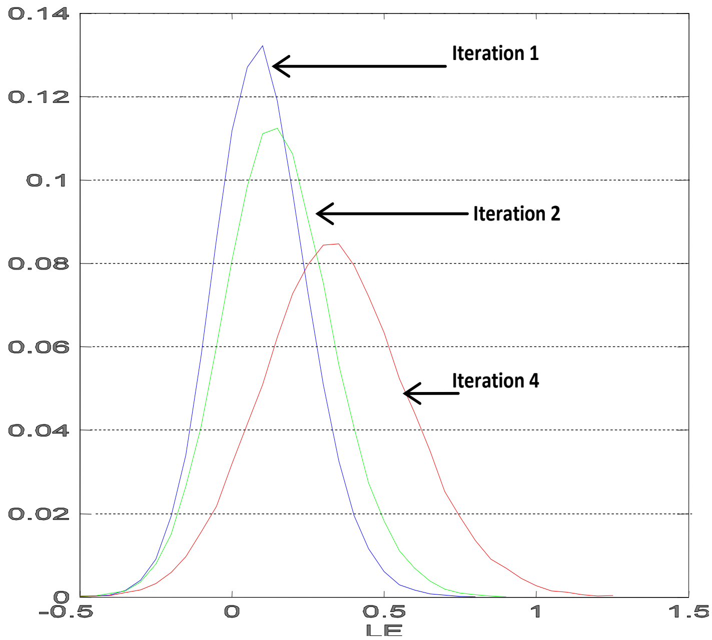

Figure 4. Evolution of the distribution of the extrinsic L_E based on iterations.

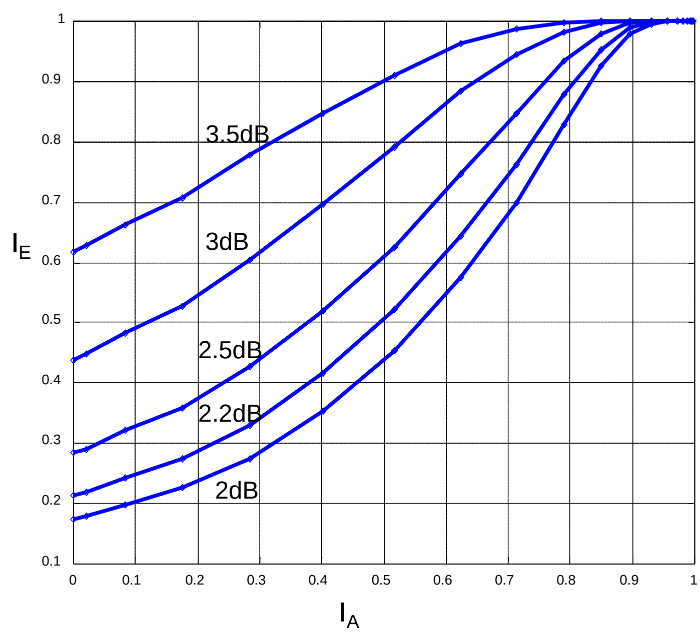

Figure 5. Extrinsic information transfer characteristics of SISO decoder for code (819, 447) on AWGN channel.

To view the convergence behavior of the iterative decoding, we draw in the same figure (see **Figure 6**) the exit charts of the two decoders. In the case of a symmetrical concatenated code the curve of the second decoder is obtained by inverting the symmetry axis of the I_E and I_A.

Figure 6 depicts the Exit chart of the concatenated code PC (74529, 36481) constructed from DSC (273, 191) for *SNR* 2.5, 3, 3.15 and 3.5 dB. The expected decoding trajectory is the "stairs"-like and can be explained as follows: Let q be the iteration index. For $q = 0$ the decoding process starts at the origin with zero a priori knowledge $I_{A,0} = 0$. At iteration q, the extrinsic output $I_{E,q} = T(I_{A,q})$ is forwarded as a priori knowledge for the next decoder $I_{A,q+1} = I_{E,q}$, and so on, until $I_A = I_E = 1$ after about 16 iterations. Each step of the staircase curve represents one iteration of the decoding process.

Figure 7 shows the Exit chart curve for iterative decoding of asymmetric concatenated code constructed from DSC (273, 191) and DSC (73, 45) at SNR 2.8 dB. In this case and contrary to the result obtained in the figure 6 we calculated and plotted the transfer curves for each Component elementary decoder. As we can see both transfer curves enters a tunnel at $SNR = 2$ dB away from the x = y − axis.

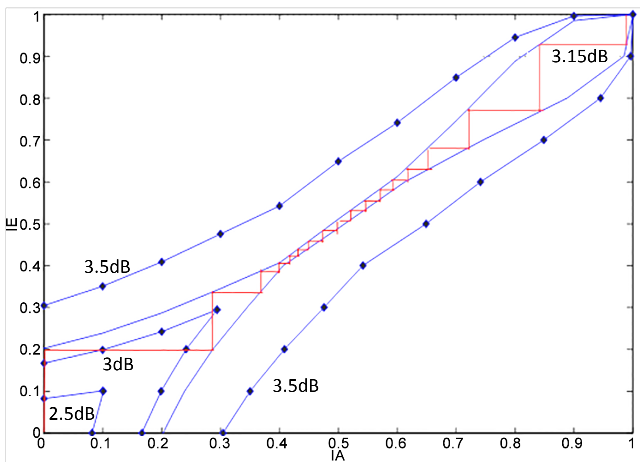

Figure 6. Exit charts of PC code (74529, 36481) and decoding trajectory on a Gaussian channel.

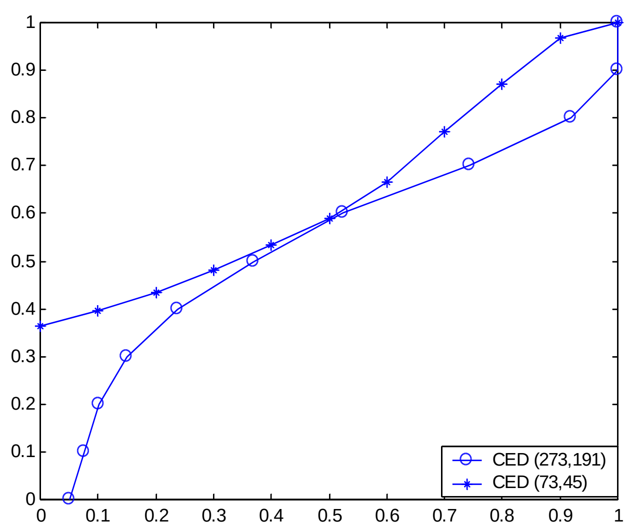

Figure 7. Exit charts of asymmetric concatenated code (19929, 8595) at SNR = 2.8 dB.

BER chart vs. Exit chart

Two main regions can be defined from the curve of BER (see Figure 8). The first is called "pinch-off Region" (for *SNR* < 2.9 dB) whose BER remains constant and iteration has no effect. The second is called "waterfall Region" (*SNR* > 2.9 dB) here, the BER decreases excessively when the rate increases *SNR*. These three regions can be illustrated using the exit chart (see Figure 6).

"Pinch-off Region" the two decoders transfer curves cross at small values of mutual information (no convergence).

"Waterfall Region": the decoding trajectory enters a tunnel area [5] near the bisector with a slower convergence rate.

For *SNR* = 3.15 dB we can observe that the decoding trajectory enters an area of tunnel close to the x = y − axis with a deceleration characteristic of convergence's speed. The bit error rate (BER) versus SNR performance on AWGN channel is displayed in Figure 8. As we can observe, the waterfall region appears to take shape at *SNR* > 2.9 dB, which is 0.25 dB away from Exit chart threshold. This uncertainty of estimate is due mainly to the fact that we use a code with finite length.

4. Conclusion

In this work, we applied Exit chart as a semi analytic method to predict convergence behavior of our iterative threshold decoding for simple and concatenated OSMLD codes with medium length and a gap of 0.3 dB was found compared to BER chart. This uncertainty of estimate is due mainly to the fact that we use a code with finite length. The great interest of Exit chart is mainly due to the time we gain compared to the simulation based on BER. Indeed if we assume that θ is the decoding time of one frame

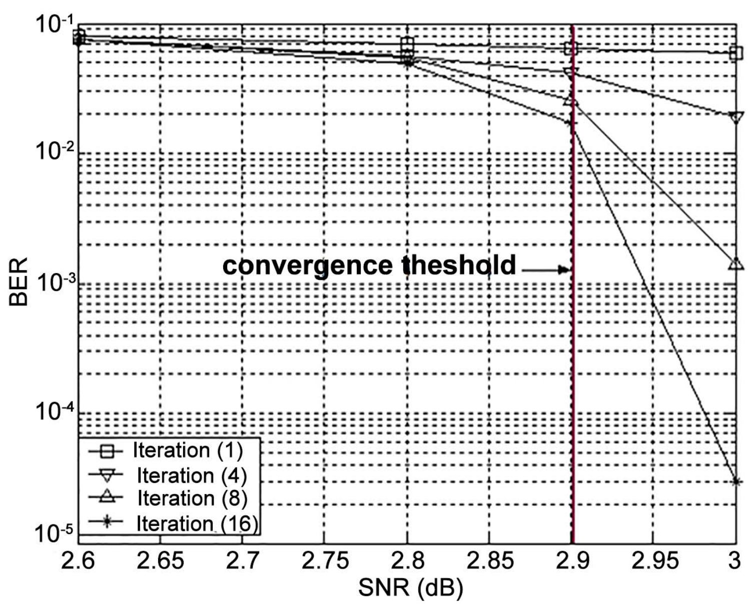

Figure 8. BER performances of ITD of PC code (74529, 36481) on a Gaussian channel.

and frame it takes to generate the BER is 1000, then the gain is of the order of $1000 \times \theta$. As future work, we plan to apply the Exit on a concatenation of OSMLD codes with a convolutional code of rate 1, but also we believe that the Exit Chart can be applied to other non-linear iterative process such as that used in learning neural network.

References

[1] Belkasmi, M., Lahmer, M. and Benchrifa, M. (2006) Iterative Threshold Decoding of Parallel Concatenated Block Codes. *Turbo Coding* 2006 *Conference*, Munich, 4-7 April 2006.

[2] Lahmer, M. and Belkasmi, M. (2010) A New Iterative Threshold Decoding Algorithm For One Step Majority Logic Decodable Block Codes. *International Journal of Computer Applications*, **7**, No. 7. https:/doi.org/10.5120/1265-1697

[3] Rkizat, K., Yatribi, A., Lahmer, M. and Belkasmi, M. (2016) Iterative Threshold Decoding of High Rates Quasi-Cyclic OSMLD Codes. *International Journal of Advanced Computer Science and Applications*, **7**, No. 4. https:/doi.org/10.14569/IJACSA.2016.070468

[4] Ayoub, F., Farchane, A., Askali, M., Belkasmi, M. and Himmi, M. (2016) Serially Concatenated OSMLD Codes Design and Iterative Decoding. *Applied Mathematical Sciences*, **10**, 2179-2188.

[5] ten Brink, S. (1999) Convergence of Iterative Decoding. *Electronics Letters*, **35**, 806-808. https:/doi.org/10.1049/el:19990555

[6] Richardson, T. and Urbanke, R. (2000) Thresholds for Turbo Codes. *Proceedings of ISIT* 2000, Sorrento, Italy, June 2000, 317. https:/doi.org/10.1109/isit.2000.866615

[7] Richardson, T.J. and Urbanke, R.L. (2001) The Capacity of Low-Density Parity-Check Codes under Message-Passing Decoding. *IEEE Transactions on Information Theory*, **47**, 599-618. https:/doi.org/10.1109/18.910577

[8] ten Brink, S. (2001) Convergence Behavior of Iteratively Decoded Parallel Concatenated Codes. *IEEE Transactions on Communications*, **49**, 1727-1737. https:/doi.org/10.1109/26.957394

[9] Lin, S. and Costello, D.J. (1983) Error Control Coding, Fundamentals and Applications. Prentice-Hall, Englewood Cliffs, NJ.

[10] Lahmer, M. and Belkasmi, M. (2007) Iterative Threshold Decoding of One Step Majority Logic Decodable block Codes. *ISSPIT Conference*, Cairo, Egypt, 15-18 December 2007, 668-673. https:/doi.org/10.1109/isspit.2007.4458188