Performance of Turbo Code with Different Parameters

Samir Jasim
College of Engineering, University of Babylon
dr_s_j_almuraab@yahoo.com
Ansam Abbas
College of Engineering, University of Babylon
'ansamabbas76@gmail.com

Abstract

Turbo codes are one of error correction coding where the errors which may be added into the transmission data through a communication channel can be detected and corrected, these codes provided for long codewords with decoding complexity. Turbo code is one of the concatenated codes connected in serial or in parallel for transmission data with great throughput and achieve near Shannon limit.

This paper presents the performance of turbo code with different parameters such as (number of iteration, type of decoding techniques, length of code, rate, generator polynomial and type of channel) get the Bit Error Rate (BER) for each case, then compare the results to specify the parameters which give the optimum performance of this code. The system is simulated by using MATLAB R2016b program.

Keywords: Turbo code, Bit Error Rate (BER), recursive systematic convolutional (RSC), Binary Phase Shift Keying (BPSK), log likelihood ratio (LLR) and Universal Mobile Telecommunications System (UMTS).

الخلاصة

تشفير التربو هو احد تشفيرات تصحيح الاخطاء، حيث الاخطاء التي تضاف الى البيانات نتيجة لارسالها وانتقالها خلال قناة اتصال ممكن ان تكتشف وبالتالي امكانية تصحيحها. تشفير التربو يستخدم مع الكلمات المشفرة الطويلة ولكن بتعقيد عالي. وهو احد انواع الترميزات المتسلسلة الربط يربط بشكل توالي او توازي يرسل البيانات بانتاجية عالية وانه تقترب من حدود شانون .

في هذه البحث تقديم اداء تشفير التربو بتغيير عدة عوامل مثل (عدد المحاولات، انواع تقنيات فك التشفير ،طول الكود، المعدل، مولد الكودات وقناة الاتصال) نحصل فيها على الاداء وBER في كل حالة وبالتالي مقارنة النتائج لتحديد العوامل التي تعطي الاداء الامثل لهذا الترميز. تم تطبيق النظام في برنامج الماتلاب R2016b .

الكلمات المفتاحية: تشفير التربو،نسبة الخطأ bit، كود التلفيفي المتماثل، محول ثنائي الطور، معدل نسبة الاحتمال، النظام العالمي للاتصالات المتنقلةUMTS).

1. Introduction

Turbo codes are introduced by Berrou, Glavieux and Thitimajshima in 1993. These codes are designed usually from two simple convolutional codes connected Serially (SCCC) or Parallel (PCCC) separated by an interleaver (Anum, 2012).

The structure of turbo code is using two encoders normally identical and the interleaver between them that reads the bits in pseudo-random order (Gooru and Rajaram, 2013). Turbo codes are chosen for the telemetry coding standard for high data rate transmission in Universal Mobile Telecommunications System (UMTS) third generation mobile communication standards with constraint length of $k = 4$ and 8 states exist in the trellis (Jagadeesh and Chaitali, 2004).

2. Turbo Codes Structure

Turbo codes are especially attractive for mobile communication systems and have been adopted of several channel coding standard. The superior performance of these codes is due to the combination of parallel concatenation code, recursive encoders, pseudo-random interleaver and iterative process of decoding, where decoders operate in a soft -input soft-output mode. Therefore, these codes are used in many applications such as: mobile radio, satellite, image processing, digital video, terrestrial telephony, etc. (Rupinder and Sarpreet ,2013).

2.1.Turbo Encoder

The fundamental of turbo encoder is using two identical Recursive Systematic Convolutional (RSC) code arranged in parallel form separated by an interleaver. The nature of the interleaver in turbo code is pseudo-random in order to minimize the correlation between the outputs of encoders that makes the best results, it's matrix forms with rows and columns, depending on the block size of the code(Christian *et.al.*,2009). The structure of turbo encoder shown in Figure (1).

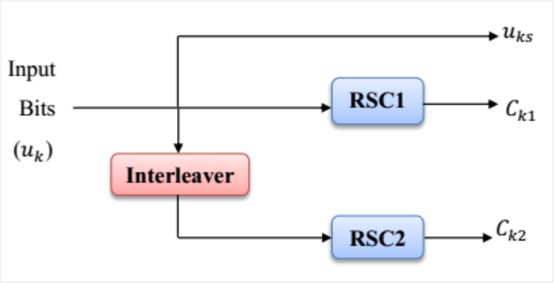

Fig. 1 : The structure of turbo encoder

Interleaver/deinterleaver are used and play an important role in the performance of turbo codes, where the interleaver helps to increase the minimum distance and break the low weight of the input sequence by spread out the burst errors by maps the sequence of bits to another sequence of bits. When the length of the interleaver is very large the excellent performance of Turbo codes is achieved (Hamid *et.al.,*2001).

According to the structure of turbo encoder the rate is $\mathrm{R} = 1/3$ in order to increase the rate puncturing technique will be used to obtain high rate $\mathrm{R} = 1/2$. Puncturing is operating on the parity bits only but the systematic bits are not punctured.(Todd, 2005).

2.2. Turbo decoder

Turbo decoders consist of a pair of convolutional decoders which cooperatively iteratively exchanging soft-decision information, the information can be passed from one decoder to the other, where each decoder takes the information correspond to the systematic, parity bits from the encoder and a priori information from the other decoder and the resulting output generated by the decoder should be soft decisions or estimates. The passing of information between the first and second decoder continues until a given number of iterations is reached. With each iteration the estimates of the information bits improve and they usually converge to a correct estimate of the message by increasing the number of iterations. However, this improvement does not increase linearly, in practice, it is enough to utilize a small number of iterations to achieve acceptable performance(Jorge and Patrick, 2006). Figure(2) illustrates the structure of turbo decoder.

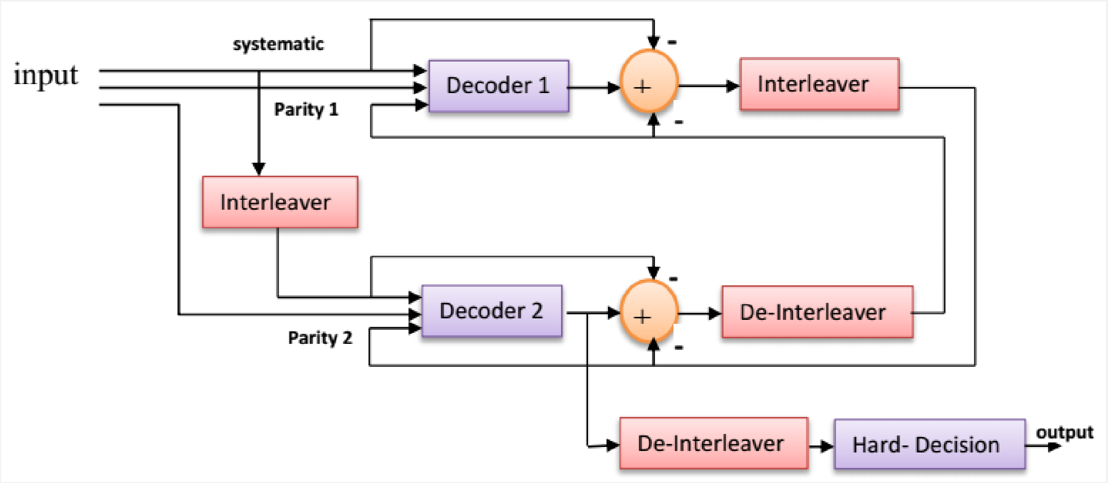

Fig. 2 : The structure of turbo decoder

The decoder produces a soft-decision to each message bits in logarithmic form known as a log likelihood ratio (LLR), in Equation(1) (Jorge and Patrick, 2006).

$$L_a(u_k) = Ln\frac{P(u_k = +1)}{P(u_k = -1)} \qquad (1)$$

where: u_k is data bits, $L_a(u_k)$ is the Log Likelihood Ratio (*LLR)* of data bit.

At the end of this process a hard decision is carried out at the second decoder to convert the final signal to 1's and 0's and compare it with the original message (Ibrahim and Mehmet,2005).

2.3. Turbo decoding techniques

Turbo decoding techniques are mainly divided into two types: the first one is Soft Output Viterbi Algorithm (SOVA) and the other is log-MAP and MAX-log-MAP evolved from Maximum A posteriori Probability (MAP). In the SOVA algorithm chooses the branch with the highest probability in the trellis and discards the other, but in the MAP algorithm takes all paths in the trellis and calculates the probabilities of each point (Jagadeesh and Chaitali, 2004).

The implementation of the MAP technique is complex due to numerical representation of probabilities, the large numbers of multiplication and division operations and non-linear functions. Therefore, the approximations of the MAP algorithms derived in log domain named Log-MAP and Max-Log MAP (Hamid *et.al.*,2001].

2.3.1. SOVA Algorithm

Soft output Viterbi Algorithm SOVA operates similarly to the Viterbi Algorithm but with two essential modifications that allow it to be used as a component decoder for turbo codes. First, SOVA is used a modified path metric that takes account of a prior probabilities of input symbols. Second, SOVA is modified to produce a soft output that indicate the reliability of the decision (Todd,2005).

The $L(u_k/y_k)$ is represented the soft output of SOVA component decoder and it is decomposed in three terms in Equation(2) :

$$L(u_k/y_k) = L(u_k) + L_c y_{ks} + L_e(u_k) \qquad (2)$$

Where: $L(u_k)$ is the a priori information, $L_c y_{ks}$ is the value of the weighted received systematic channel, L_c is denoted as the channel reliability value and calculated in Equation(3):

$$L_c = \frac{E_b}{2\sigma^2}4a \qquad (3)$$

where E_b = energy/bit , σ^2 = $noise$ variance and a = fading amplitude and $\mathrm{L_e(u_k)}$ is the extrinsic value that produced by present SOVA component decoder.

The path metrics can be calculated by the cross correlation to the received, according to the Equation(4):

$$M(S_k^s) = M(S_{k-1}^s) + \frac{1}{2}u_k L(u_k) + \frac{L_c}{2}\sum_{l=1}^{n} y_{kl}\, x_{kl} \qquad (4)$$

Where : $M(S_{k-1}^s)$ the prevised path metric, n the length of codewords, x_{kl} the codewords after BPSK modulation.

The extrinsic information that produced by first SOVA component decoder illustrates in Equation(5) .

$$L_{e1}(u_k) = L_1(u_k/y_k) - \ L_1(u_k) - L_c y_{ks} \qquad (5)$$

The extrinsic information that produced by second SOVA component decoder illustrates in Equation(6) .

$$L_{e2}(u_k) = L_2(u_k/y_k) - \ L_2(u_k) - \pi(L_c y_{ks}\ ) \qquad (6)$$

The $\mathrm{L_{e2}(u_k)}$ is deinterleaved, then return to feed back to first SOVA as a-priori information to the next iteration. This process repeated in each iteration and finally stop after specified number of iteration, where by increasing the number of iteration better results can be achieved (Ang and Lim, 2009).

At the last iteration hard decision can be calculated from the second decoder after de-interleaving.

2.3.2. Log-MAP Algorithm

Although log-MAP has low complexity compared to MAP, but it still requires a large number of computations [Yi Bo-nian,2013].

The LLR expression of Log-MAP algorithm can be expressed as Equation(7) (Ang and Lim, 2009):

$$L(\hat{u}_k) = log\frac{\sum_{u_k=+1}^{(s',s)} \exp(\alpha_{k-1}^{LM}(s')) \cdot \exp(\gamma_k^{LM}(s',s)) \cdot \exp(\beta_k^{LM}(s))}{\sum_{u_k=-1}^{(s',s)} \exp(\alpha_{k-1}^{LM}(s')) \cdot \exp(\gamma_k^{LM}(s',s)) \cdot \exp(\beta_k^{LM}(s))}$$

$$= log\left(\sum_{u_{k=+1}}^{(s',s)} \exp(\alpha_{k-1}^{LM}(s') + \gamma_k^{LM}(s',s) + \beta_k^{LM}(s))\right)$$

$$- log\left(\sum_{u_{k=-1}}^{(s',s)} \exp\left(\alpha_{k-1}^{LM}(s') + \gamma_k^{LM}(s',s) + \beta_k^{LM}(s)\right)\right) \qquad (7)$$

Where: $\alpha_{k-1}^{LM}(s')$ forward recursion, $\beta_k^{LM}(s)$ backward recursion and $\gamma_k^{LM}(s',s)$ transitional probability in logarithmic domain.

2.3.3. Max-log-MAP Algorithm

The Max-log-MAP is a simplification of the log-MAP algorithm, it includes one forward and one backward recursion through the received soft input data and calculates a number of different metrics [Yi Bo-nian,2013].

The Max-log-MAP algorithm operates to reduce the complexity of the log-MAP by performing the expression to the forward and backward recursion, Equations (8,9 and 10) illustrate that (Eid, 2013).

$$\alpha_k^{MLM}(s) = \max\left(\left[\gamma_{u_{k=+1}}^{LM}(s',s) + \alpha_{k-1}^{LM}(s')\right], \left[\gamma_{u_{k=-1}}^{LM}(s',s) + \alpha_{k-1}^{LM}(s')\right]\right) \qquad (8)$$

$$\beta_k^{MLM}(s') = max\left(\left[\gamma_{u_k=+1}^{LM}(s',s) + \beta_k^{LM}(s')\right], \left[\gamma_{u_k=-1}^{LM}(s',s) + \beta_k^{LM}(s')\right]\right) \qquad (9)$$

Then LLR for Max-log-MAP can be expressed as follows :

$$L(\hat{u}_k) = \max_{s',s(u_k=+1)}([\gamma_k^{LM}(s',s) + \alpha_{k-1}^{MLM}(s) + \beta_k^{MLM}(s')], [\gamma_k^{LM}(s',s) + \alpha_k^{MLM}(s) + \beta_k^{MLM}(s')]) - \max_{s',s(u_k=-1)}([\gamma_k^{LM}(s',s) + \alpha_{k-1}^{MLM}(s) + \beta_k^{MLM}(s')], [\gamma_k^{LM}(s',s) + \alpha_k^{MLM}(s) + \beta_k^{MLM}(s')]) \quad (10)$$

where: k is the time instant, s_k is the current state at time instant k, $\alpha_k(s)$ is the forward state metric at time instant k, $\beta_k(s)$ is the backward state metric at time instant k and γ_k: the branch state metric

By applying this approximation the degradation will occur in the performance of Max-log-MAP compare with the log-MAP (Eid, 2013) .

3. Results and Discussion

This section contains the results which were obtained from the Simulation of turbo code on MATLAB program. In this work many parameters are varying to observe the performance in each case such as: number of iteration, type of decoding techniques, length of code, rate, generator polynomial and type of channel. Figure (3) illustrates the flow chat of the proposed system.

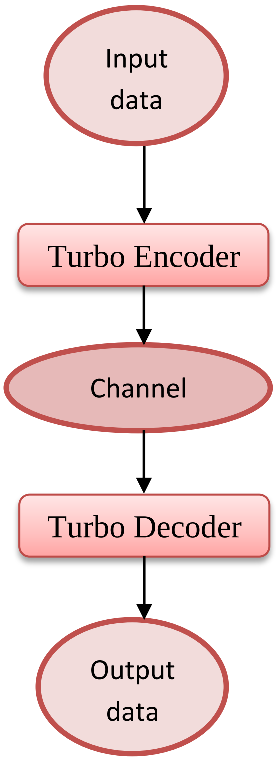

Fig. (3) Flow Chat of the proposed system

In Table (1), all parameters of turbo code Simulation, get the results for each case then compared between them.

Table (1): Parameters of turbo code Implementation

1	Modulation	Binary Phase Shift Keying (BPSK)
2	Interleaver type	UMTS
3	Number of iterations	1-8
4	Decoding Techniques	SOVA, log-MAP, MAX-log-MAP
5	Code Lengths	256,512, 1024, 4096
6	Rate	$R_1 = 1/3$, $R_2 = 1/2$
7	Generator Polynomials	$g_1(13{,}15)_8$, $g_2(7{,}5)_8$, $g_3(37{,}21)_8$
8	Channels	AWGN, Rayleigh fading

In order to illustrate the effect of the number of iteration, Figure (4) illustrates that in Log-MAP decoding technique with 8 iterations, 256 bit/frame,1024 frame and $R = 1/2$, as shown the BER is improving repaid for the first few iteration then the improvement is less until 6-8 iteration where the performance remains constant.

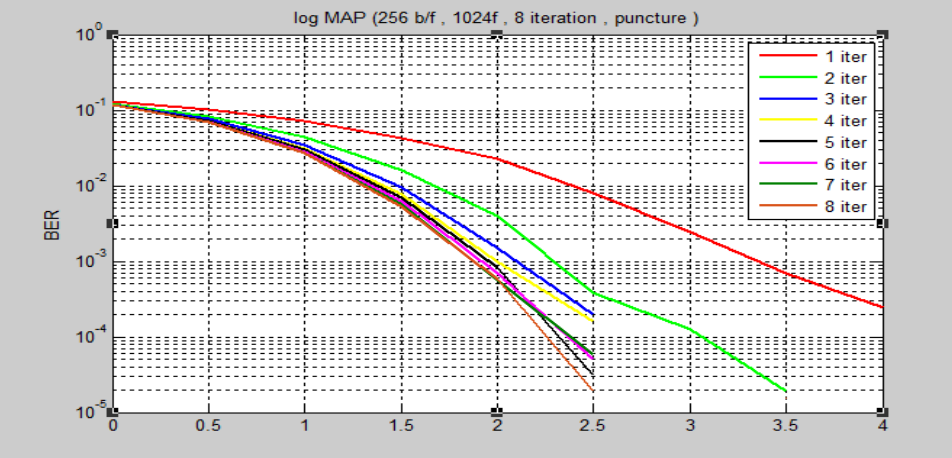

Fig. 4 : performance of turbo code with 8 iteration

Table(2) illustrates the value of BER at $Eb/No = 2\ dB$ for each iteration.

Table (2): BER at different number of iteration

$Eb/No\,(dB)$	iteration	BER
2	1	2.5×10^{-2}
2	2	4×10^{-3}
2	3	1.75×10^{-3}
2	4	1×10^{-3}
2	5	8×10^{-4}
2	6	6.5×10^{-4}
2	7	5.75×10^{-4}
2	8	5.75×10^{-4}

In Figure (5), three types of decoding techniques are introduced (SOVA, log-MAP and MAX-log-MAP) these techniques are applied with 500 bit/frame, 1024 frame, 4 iteration and Rate $R = 1/3$, by increasing the Eb/No the performance is improved, log–MAP technique is best than SOVA and MAX-log-MAP for the same conditions of the system.

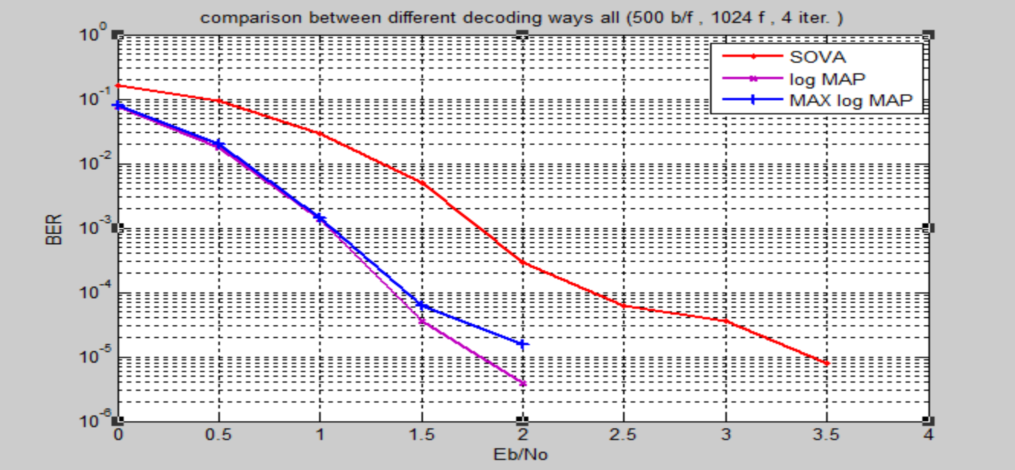

Fig.5 : Turbo code in three decoding techniques

In Figure (6), The effect of the number of code length applied in log-MAP technique with different lengths 256, 512, 1024 and 4096 bit/frame all them with 1024 frame, 4 iteration and $R = 1/3$. Table (3) illustrates how the BER improve by increasing the code length at $Eb/No = 1\ dB$ for each length .

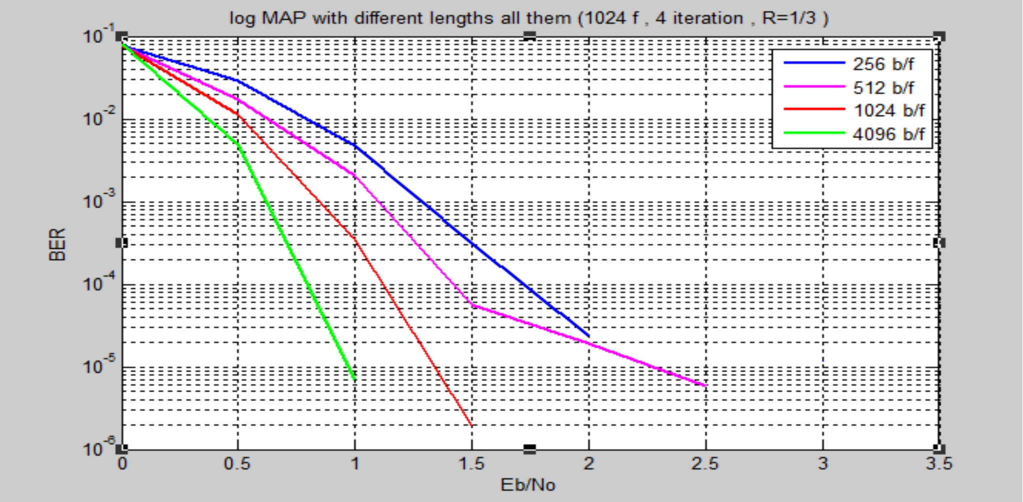

Fig. 6 : log-MAP technique with different code lengths

Table (3): BER at different code length in log-MAP technique

Eb/No (dB)	Code length	BER
1	256	4.75×10^{-3}
1	512	2×10^{-3}
1	1024	3.5×10^{-4}
1	4096	7.5×10^{-6}

By using puncturing technique the rate of the system can be incearsed but the BER is worse as shown in Figure (7), this applied in log-MAP technique with frame size 256 b/f, 1024 f and 4 iteration.

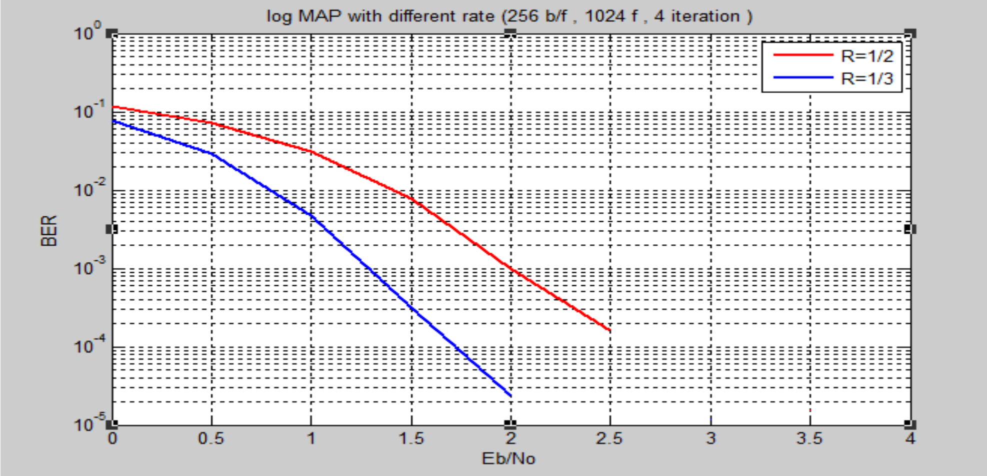

Fig. 7: log-MAP techinque with two different Rate

For turbo code in UMTS the standard generator polynomial is $g = (13{,}15)_8$, Figure (8) illustrates another generators polynomials applied in the system $g_1 = (7{,}5)_8$ *and* $g_2 = (37{,}21)_8$,in log-MAP technique with frame size 512 b/f, 1024f, 4 iteration and R=1/3.

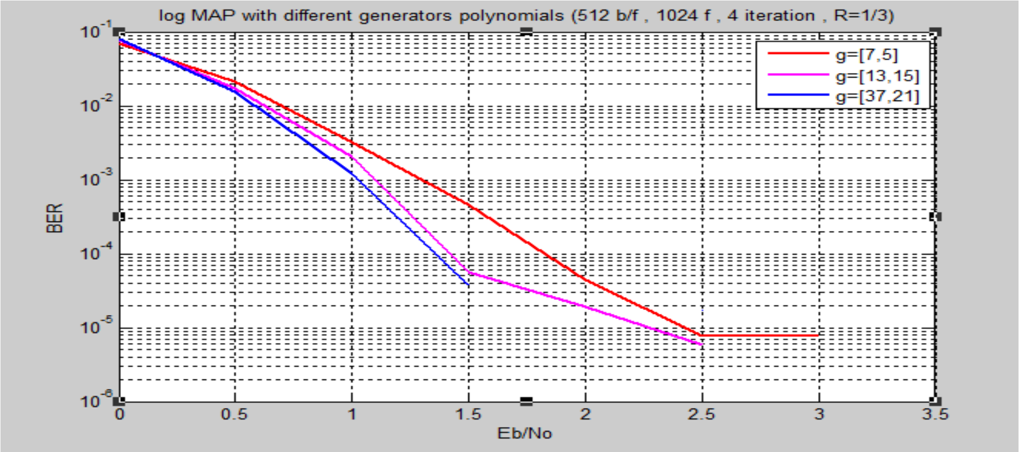

Fig. 8 : Turbo code with different generators polynomials in log-MAP technique

All changes in the parameters that introduced in Figures above applied in AWGN channel, in Figure(9) the performance of turbo code in Rayleigh fading channel compare with AWGN channel, applied in log-MAP technique with frame size 512 b/f, 1024 f, 3 iteration and R = $1/3$. For an AWGN channel the fading amplitude a=1 whereas for a Rayleigh fading channel a is a random variable, as shown when using AWGN channel the performance is best than the Rayleigh fading channel.

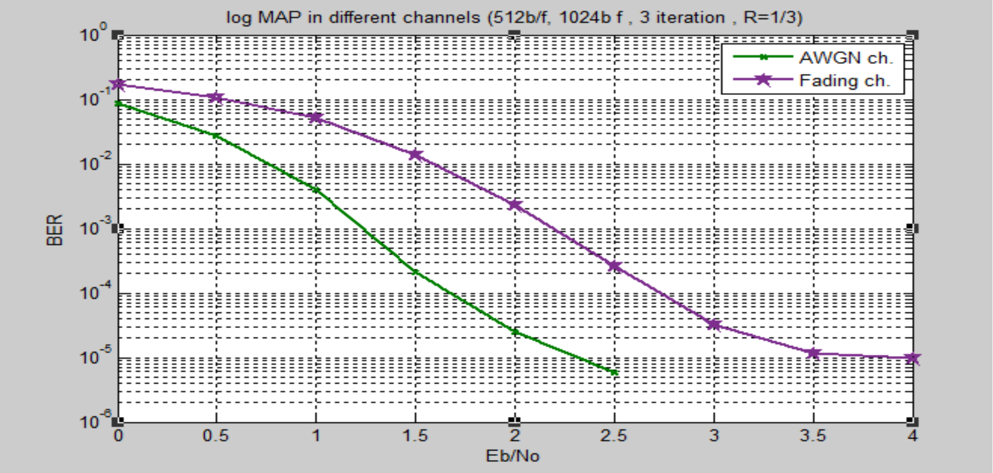

Fig.9 : log-MAP technique in different channels

4. Conclusion

In this paper, the Simulation of Turbo Code is in MATLAB R2016b program. In this work the modulation is BPSK and the interleaver with the structure of UMTS interleaver, from the results which can be concluded that, the performance of turbo code is improvement by increasing the number of iteration. For the same conditions of the system the BER in log–MAP is best than SOVA and MAX-log-MAP. The increasing of code length is improved the performance. Puncturing technique is degradation the performance. The performance of turbo code is changing and depending on the generator polynomial of the system. Finally observed the performance is worse when using Rayleigh fading channel compare with the AWGN.

References :

Ang, Lay Hong and Lim, Wee Guan, 2009, " SOVA Based LTE Turbo Decoders",Master Thesis, Lund University.

Anum Imran, 2012," Software implementation and performance of UMTS Turbo decoder ", M.Sc. Thesis, Tampere University.

Christian Benkeser, Andreas Burg, Teo Cupaiuolo and Qiuting Huang, 2009,"Design and Optimization of an HSDPA Turbo Decoder ASIC ", Journal Of Solid-State Circuits, VOL.X,NO. X, January.

Eid Mohamed A. A., 2013, "Design and Implementation for multi-standard Turbo Decoder using a reconfigurable ASIP ", Master The Cairo University.

Gooru Santosh and DR.S.Rajaram, 2013, "Design and Implementation of turbo coder for LTE on FPGA", International Journal of Electronics Signals and Systems (IJESS) , ISSN: 2231-5969,Vol-3,Iss-2,2013.

Hamid R. Sadjadpour, Neil J. A. Sloane, Masoud Salehi, and Gabriele Nebe, 2001, “Interleaver Design for Turbo Codes,” IEEE Journal on Selected Area In Communications, vol.19, No.5,May.

Ibrahim S. Raad and Mehmet Yakan,2005, " Implementation of a turbo codes test bed in the Simulink environment ", International Symposium on Signal Processing and Its Applications (pp. 847-850). Piscataway:IEEE.

Jagadeesh Kaza and Chaitali Chakrabarti, 2004," Design and Implementation of Low-Energy Turbo Decoders", IEEE Transactions On Very Large Scale Integration (VLSI) Systems, VOL. 12. NO. 9, September.

Jorge Castiñeira Moreira and Patrick Guy Farrell, 2006," Essential of Error Control Coding " , Wiley.

Rupinder Kaur and Sarpreet Singh, 2013 " Techniques for Turbo Decoding Using Parallel Processing, Comparative Analysis " , Volume 3 , Issue 4 , April 2013.

Todd K. Moon, Wiley , 2005," Error Correction Coding: Mathematical Methods and Algorithms ".

Yi Bo-nian,2013, "Turbo Code Design and Implementation of High-Speed Parallel Decoder", Telkomnika, Vol. 11, No. 4, April 2013,pp. 2116-2123.