PERFORMANCE EVALUATION OF TURBO CODE ENDCODING AND DECODING

Yahya Jasim Harbi
College Of Engineering
University Of Kufa
Najaf , Iraq

ABSTRACT:

Channel coding is very important for secure communication. The most widely used communication coding is turbo coding technique, which is very secure technique. We deal in this paper with turbo code which is new class of convolutional codes. The performance of turbo code reaches the Shanon limits in terms of bit error rate BER. To build the turbo code encoder we used two parallel concatinations of two Recursive Systematic Convolutional codes (RSC) and the associated decoder which is using the feedback decoding rule. At low signal-to-noise ratio (SNR), bit error rates (BER) of Turbo code were obtained by simulation.

Key words: Turbo codes, PCCCs, Frame size, Encoder memory size, Iterative decoding

أداء الجفرات التوربينية في التجفير وإزالته

الخلاصة

تشفير القنوات مهم جدا للحصول على اتصالات امنة. ومن اكثر طرق التشفير شيوعا هي استخدام الجفرات التوربينية والتي تعتبر من اكثر التقنيات امنا. قمنا بهذا البحث بدراسة الجفرات التوربينية والتي تعتبر من التقنيات الحديثة بهذا المجال. اداء الجفرات التوربينية يقترب من حدود شانون بمعدل خطأ البت (BER). لبناء مجفر التيربو استخدمنا اثنين من (RSC) متوازية التراص مع كاسر الشفرة باستخدام طريقة التغذية العكسية لكاسر الجفرة. باستخدام المحاكات حصلنا على معدل خطأ البت (BER) في المستويات المنخفضة من نسبة الاشارة الى الضوضاء (SNR).

List of Acronyms

AWGN: Additive White Gaussian Noise (channel)
BER: Bit Error Rate
ICC: International Conference on Communication
LLR: Log-Likelihood Ratio
MAP: Maximum A Posteriori
PCCCs: Parallel Concatenation Convolutional Codes
RSC: Recursive Systematic Convolutional (code)

.

1. INTRODUCTION:

The basis of informational revolution and its transformation to the power of the universe is dependent on the communications technology, its speed and security. The quality of information from source to destination should be the best. Channel coding is a useful tool in the design of reliable digital communication. Channel coding is necessary to improve error performance for digital communication system by mapping input sequences into code sequences which inserts redundancy and memory to the transmission. Information theory states that arbitrarily small error rates are achievable provided the rate of transmission is less than the capacity of the channel.

The gap between practical coding system and Shannon's theoretical limit closed even further in June 1993 at the International Conference on communications (ICC). At this conference two papers were presented that discussed a new class of codes and the associates decoding technique [1]. A very powerful channel coding scheme was developed by Berrou, Galvieux and Thitmajshima, which used ideas related to both block and trellis codes. The encoding scheme uses simple convolutional codes separated by interleaving stages to produce generally low rate block codes. The decoding is done by decoding the convolutional encoder separately using Log MAP Algorithm and sharing bit reliability information in an iterative manner. This coding scheme is called Turbo Code and is found capable of achieving near Shannon capacity performance, the theoretical limit [2].

This paper deals with achieved maximal information transfer over a limited-bandwidth communication link in the presence of data-corrupting noise. And the expected results that is when the numbers of decoder iterations increases, the performance of the code also increases, resulting in lower BER.

2. ENCODING OF TURBO CODE:

A turbo code is the Parallel Concatenation Convolutional Codes (PCCCs) of two or more component codes. In its original form, the constituent codes were from a subclass of Convolutional codes known as recursive systematic Convolutional (RSC) codes [1]. A generalized turbo encoder is known in the Fig.1. In the figure, a data block u, which is k bits long enters the coders.

The PAD block appends n - k tail bits to the data block, which yields the sequence x_0. This n bit sequence is then fed in parallel into M sets of interleavers αi and encoders ENCi. Each interleaver scrambles the x_0 sequence in a pseudo-random fashion and feeds its output into a constituent encoder. Each of the M constituent encoders presents a parity sequence xi at its output. The information sequence x_0 together with the M parity sequences is concatenated to form the code word [3].

From the above discussion, it is clear that the Turbo Code encoder consists of two main parts:

1. A set of classical convolutional encoders.

A set of interleavers. The basic rule of interleaver is to spread the residual error block of rectangular form which makes the decoding process stronger.

2-1 Encoder:

The common practice of encoder is to use Recursive Systematic Convolutional (RSC) encoders. By using a convolution encoder, it is possible for the decoder to utilize a modified version of the Viterbi algorithm. Recursive encoder is used as nonrecursive encoder will result in output codes with poor distance properties [4].

Two identical rate $r = 1/2$ RSC encoders work on the input data in parallel as shown in Fig.2. As shown in the figure, the input data is interleaved before being fed into the lower encoder. Because the encoder is systematic (one of the outputs is the input itself) and receive the same input (although in different order), the systematic output of the lower encoder completely redundant and dose not need to be transmitted. The overall code rate of the parallel concatenated code is $r = 1/3$, although

higher code rates can be obtained by puncturing (selectively removing outputs from the transmission stream) the parity output with a multiplexer (MUX) circuit[5][6].

2-2 Interleaver

A burst of errors is defined as a sequence of bit errors. The method of interleaver has proved to increase the reliability in the burst error channel. In Turbo code, the structure of interleaver has been carefully chosen. It allows that input sequences for which one encoder produces low weight codewords will usually cause the other encoder to produce high weight codewords. Although the constituent codes are individually weak, the combination is surprisingly powerful [7][8].

2-3 Puncture The Output:

It is a common practice to puncture the output of the encoder in order to increase the code rate to 1/2. For a rate 1/2 punctured turbo code, the first output stream is the input stream itself (plus the necessary padding), while the seconding output stream is generated by multiplexing the M non-systematic output of the RSC encoders[9].

2-4 An Example Of Encoding Of Turbo Code:

Because turbo codes are linear block codes, the encoding operation can be viewed as the modulo-2 matrix multiplication of an information vector with a generator matrix. Here the encoding a sequence by Turbo Code with the help of matrix representation is demonstrated.

If the code generator matrix g = [1 1 1 ; 1 0 1], the RSC encoder will become as shown in Fig.3.

The encoder has constraint length of 3 with memory of 2 only. The encoding algorithm is similar to that of convolutional encoding. In order to generate a block code using a parallel concatenation of convolutional encoders, it is desirable for encoders to start and end in the all-zero state. To ensure this will happen, extra bits are needs to "clear" the memory of encoder. Thus, the number of extra bits needed equals to the memory of encoder. In this example, 2 extra bits are needed.

For an input u = [1 0 1],

Input u_k	State D1	State D2	Feedback r	Output x_1
1	0	0	1	1
0	1	0	1	1
1	1	1	1	0
0	1	1	0	1
1	0	1	0	1
	0	0		

Therefore, output code c = [1 1 0 1 1]

If a pseudo-random interleaver α with the mapping table as follow:

l	α(l)
1	2
2	5
3	4
4	1
5	3

The interleave function α (l) means the lth bit will take the α (l)th bit of the original code.
The above table can also be represent by a interleaver matrix a = [2 5 4 1 3].With input u = [1 0 1 (0 1)], interleaver output = [0 1 0 1 1].

Then the turbo encoder become as shown in Fig.(4):

2. DECODING OF TURBO CODE:

It is proposed that an iterative decoding scheme should be used. The decoding algorithm is similar to Viterbi algorithm in the sense that it produces soft outputs. While the Viterbi algorithm outputs either 0 or 1 for each estimated bit, the turbo code decoding algorithm outputs a continuous value of each bit estimate. While the goal of the Viterbi decoder is to minimize the code word error by finding a maximum likelihood estimate of transmitted code word, the soft output decoding attempts to minimize bit error by estimating the posterior probabilities of individual bits of the code word. This decoding algorithm is called Software Decision Viterbi Decoding [10][12].

The turbo decoder consists of M elementary decoders, one for each encoder in turbo encoding part. Each elementary decoder uses the Software Decision Viterbi Decoding to produce a software decision for each received bit. After an iteration of the decoding process, every elementary decoder shares its soft decision output with the other M - 1 elementary decoders.

It is clear that as the number of these iterations approaches infinity, the estimate at the output of decoder will approach the maximum a posteriori (MAP) solution.
Fig.(5) described the Turbo code decoder.

A log ratio of the posteriori probability of u_k conditioned on the received signal y is defined as

$$L(u_k) \triangleq \log\left[\frac{P(u_k = 1/y_1^N)}{P(u_k = 0/y_1^N)}\right] \tag{1}$$

The decoding decision of $\tilde{u}_k$ is made based on the sign of L(u_k), i.e.,

$$\tilde{u}_k = sign[L(u_k)]. \tag{2}$$

L(u_k) is computed by three terms which are L_apriori, L_channel , and $L^e(u_k)$. L_apriori is a priori information based on the input bit u_k at time k. It is provided by the previous decoder. L_channel is the received systematic bit at time k, referring to Appendix A for details.

$$L(u_k) = \left[L^e(u_k) + Lc \cdot y_k^{1,s}\right] + \log\frac{\sum_{u^+}\tilde{\alpha}_{k-1}(s') \cdot \tilde{\beta}_k(s) \cdot \gamma_k^e(s',s)}{\sum_{u^-}\tilde{\alpha}_{k-1}(s') \cdot \tilde{\beta}_k(s) \cdot \gamma_k^e(s',s)}$$

$$= L\_apriori + L\_channel + L^e(u_k) \tag{3}$$

where L_aprior and L_channel denote $L^e(u_k)$ and $Lc \cdot y_k^{1,s}$ respectively. $\sum_{u^+}()$ is the summation over all the possible transition branch pair (s_{k-1}, s_k) at time k given input u_k=1 and $\sum_{u^-}()$ is the summation over all the possible transition branch pair (s_{k-1}, s_k) at time k given input u_k=0. Lc is the channel reliable factor; its computation is given as the following,

$$Lc = \frac{4 \cdot A \cdot SNR\_b}{p} \tag{4}$$

where A=1 for AWGN channel, SNR_b is the uncoded bit-energy-to-noise-ratio ($\dfrac{E_b}{N_0}$), p denotes

$1/r_c$, r_c is code rate of the Turbo encoder.

$L^e(u_k)$ is an extrinsic information based on all parity and systematic information except the systematic value at time k. It can be passed on to a subsequent decoder. It is computed using the following equations:

$$L^e(u_k) \triangleq \log \frac{\sum_{u^+} \widetilde{\alpha}_{k-1}(s') \cdot \gamma_k^e(s',s) \cdot \widetilde{\beta}_k(s)}{\sum_{u^-} \widetilde{\alpha}_{k-1}(s') \gamma_k^e(s',s) \cdot \widetilde{\beta}_k(s)} \qquad (5)$$

Where

$$\gamma^e(s',s) = \exp\left[\sum_{i=2}^{q}\left(Lc \cdot \frac{1}{2} \cdot y_k^{i,p} \cdot c_k^i\right)\right]. \qquad (6)$$

$\widetilde{\alpha}_k(s)$, $\widetilde{\beta}_{k-1}(s')$ can be computed recursively with initial conditions described below:

$$\widetilde{\alpha}_k(s) = \frac{\sum_{s'} \widetilde{\alpha}_{k-1}(s') \cdot \gamma_k(s',s)}{\sum_{s}\sum_{s'} \widetilde{\alpha}_{k-1}(s') \cdot \gamma_k(s',s)} \; ,$$

$$\widetilde{\alpha}_0(s) = \begin{cases} 1 & \text{if } s = 1 \\ 0 & otherwise \end{cases}. \qquad (7)$$

$$\widetilde{\beta}_{k-1}(s') = \frac{\sum_{s} \widetilde{\beta}_k(s) \cdot \gamma_k(s',s)}{\sum_{s}\sum_{s'} \widetilde{\alpha}_{k-2}(s') \cdot \gamma_{k-1}(s',s)} \; ,$$

$$\widetilde{\beta}_N = \begin{cases} 1 & \text{if } s = 1 \\ 0 & \text{otherwise} \end{cases}. \qquad (8)$$

$$\gamma_k(s',s) \propto \exp\left[\frac{1}{2} \cdot L^e(u_k) \cdot u_k + Lc \cdot \frac{1}{2} \cdot y_k^{1,s} \cdot c_k^1\right]\exp\left[\sum_{i=2}^{q}\left(Lc \cdot \frac{1}{2} \cdot y_k^{i,p} \cdot c_k^i\right)\right] \qquad (9)$$

For example, at any given iteration, decoder 1 $L_1(u_k)$ is computed as

$$L_1(u_k) = Lc \cdot y_k^{1,s} + L_{21}^e(u_k) + L_{12}^e(u_k) \qquad (10)$$

$\widetilde{u}_k = \text{sign}[L_1(u_k)]$

where $L_1(u_k)$ is given in equation (3). $L_{21}^e(u_k)$ is extrinsic information for decoder 1 derived from decoder 2, and $L_{12}^e(u_k)$ is the third term in equation (3) which is used as the extrinsic information

for decoder 2 derived from decoder 1. The decoders are sharing the information with each other. The value of $L_1(u_k)$ decides the degree of the reliability of $\tilde{u}_k$.

The performance of Turbo Code rate ½ RSC component code for various numbers of decoder iterations is shown in Figure 6. In this figure, it can be seen that as the numbers of decoder iterations increases, the performance of the code also increases, resulting in lower BER. Figure 7 and Figure 8 show the simulated performance results for two specific cases. In Figure 7, the Turbo code with rate 1/3 and in figure 8 the rate is ½ with the same constraint length and the numbers of decoding iterations, the difference between the simulated performance results is negligible for high bit energy to noise power spectral density E_b/N_o (SNR>2), but for lower E_b/N_o (SNR<2), this difference becomes greater.

4. CONCLUSIONS

Turbo coding technique has great potentials in achieving communications at very low values of E_b/N_o for AWGN channel. The performance of Turbo codes depends on the interleaver design which is embedded in the Parallel encoder. To get lower bit error rate we need large interleaver size which leads to longer decoding delay. General analytical upper bounds and design rules for concatenated codes with interleavers over AWGN channel were presented. Mutual information between transmitted systematic bits and extrinsic output of constituent decoders was found to be a useful measure for gaining insight into the convergence behavior of iterative decoding.

5. REFERENCES:

[1] C. Berrou, A. Glavieux, and P. Thitimasjshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes (1)," in Proc., IEEE Int. Conf. on Commun., pp. 1064-70, May 1993.

[2] M. C. Valenti, "Iterative Detection and Decoding for Wireless Communications", Ph.D. thesis, Virginia Polytechnic Institute and State University, July 8, 1999.

[3] H. H. ABBAS, "Simulation of Turbo-Encoder-Decoder Performance. Ph.D. thesis, Baghdad University, Oct. 2001.

[4] S. Benedetto, D. Divsalar, D. Montorsi, and F. Pollara, "A Soft-Input Soft-Output Maximum A Posteriori (MAP) Module to Decode Parallel and Serial Concatenated Codes", TDA Progress Report 42-127, November 15, 1996.

[5] M. C. VALENTI, "Turbo Codes and Iterative processing", Proc. IEEE New Zealand wireless communication Symposium 98, Nov.1998.

[6] Kuhn and Desmond, "Near-Shannon-limit channel coding using woven convolutional codes", Melbourne University, Nov. 13, 2000.

[7] Jian Qi "Turbo Code in Is-2000 Code Division Multiple Access Communications under Fading" M.Sc. thesis, Wichita State University, 1999.

[8] S. Benedetto and G. Montorsi, "Generalized concatenated codes with interleavers," in Proc., Int. Symp. on Turbo Codes and Related Topics, (Brest, France), pp. 32-9, Sept. 1997.

[9] Y. YASUDA, K. KASHIKI, and Y.HIRATA, "High-Rate punctured Convolutional Codes for Soft Decision Viterbi Decoding," IEEE Trans. Commun. Vol. Com.-32, pp.315-319, March 1984.

[10] J. Cheng, "Iterative Decoding", Ph. D, Thesis, California Iinstitute of Technology, Pasadena California, 1997.

[11] S. A. Barbulescu, "Iterative Decoding of Turbo Codes and Other Concatenated Codes", Univ. of South Australia, Feb. 1996.

[12] Marius OLTEAN, Maria KOVACI, Horia BALTA Andrei and CAMPEANU, "Multy Binary Turbo Coded WOFDM Performance in Flat Rayleigh Fading Channels", ACTA TECHNICA NAPOCENSIS Electronics and Telecommunications, Volume 49, Number 3, 2008.
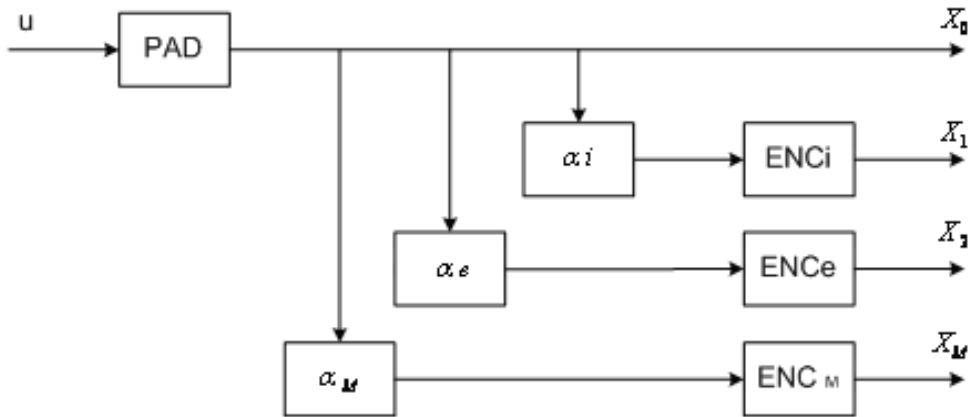

Fig.1. A general turbo encoder [7]
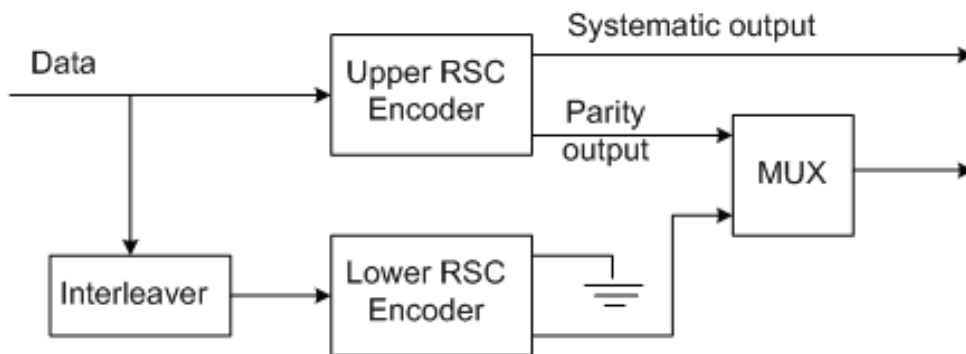
Fig.2. Turbo encoder [5]
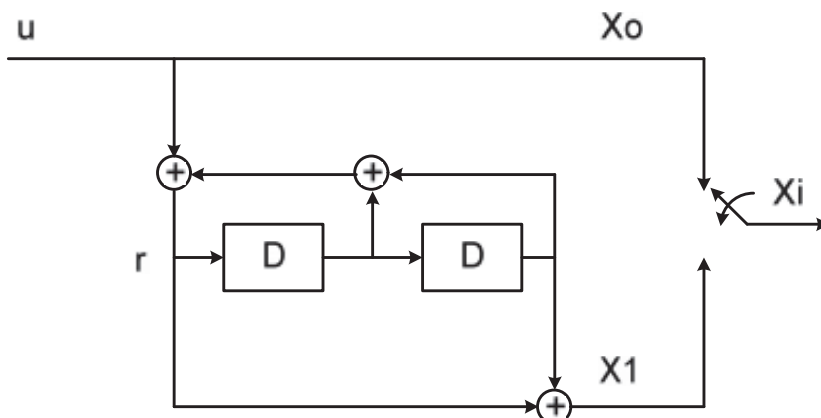
Fig.3. Example rate ½, constraint length Kc=3 recursive systematic convolutional (RSC) encoder
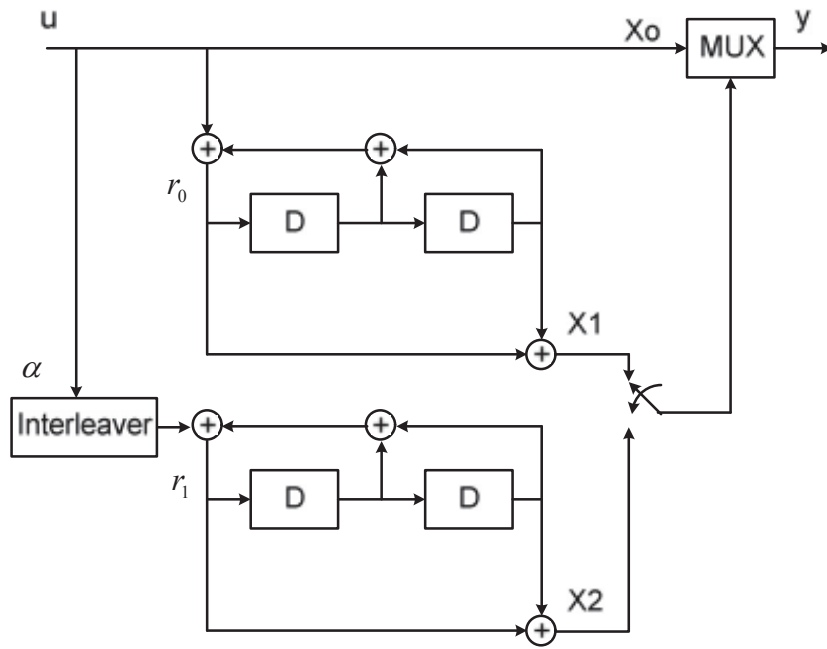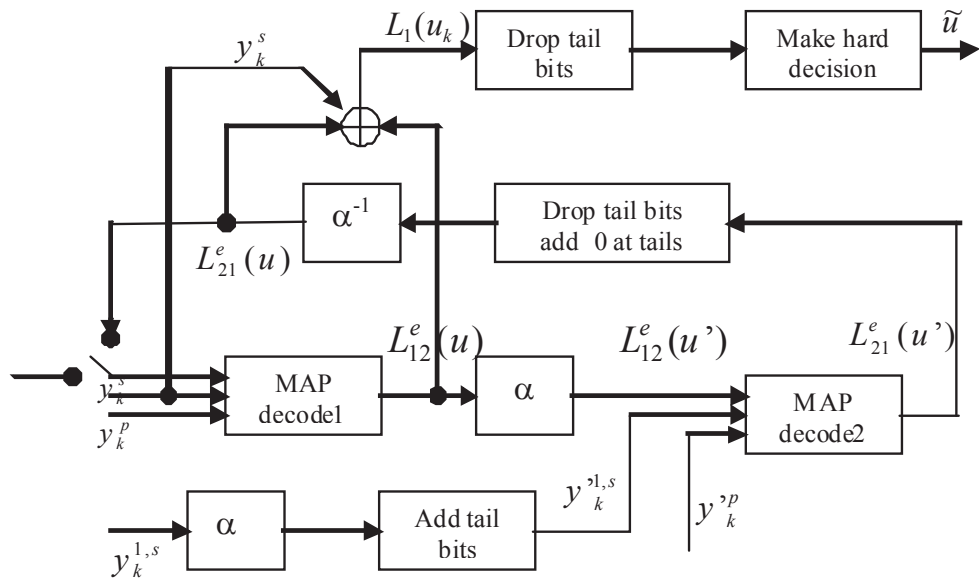
Fig.4. Example turbo encoder.

α is the notation for interleaver,
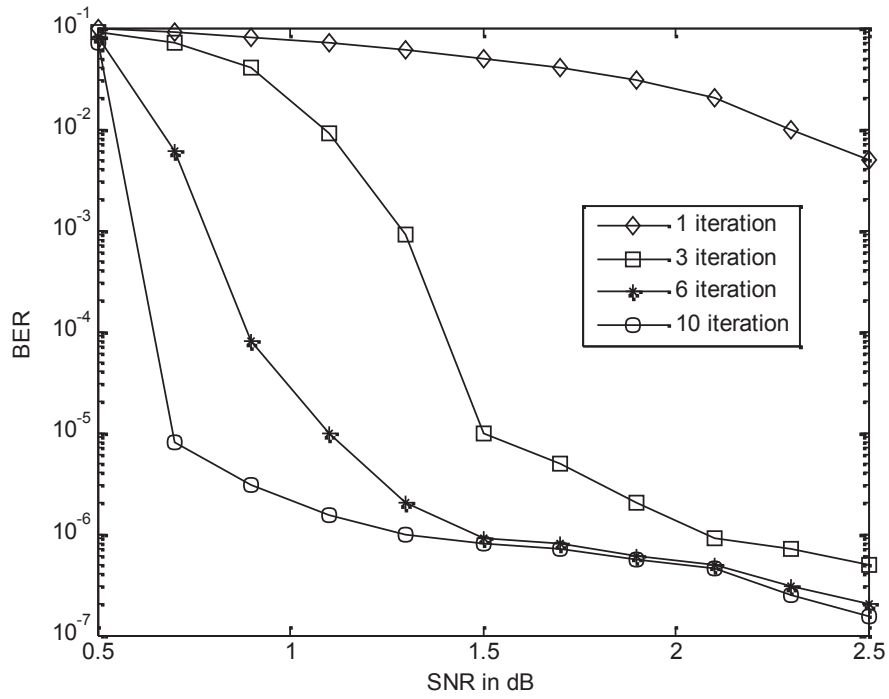α^{-1} is the notation for de-interleaver.

Fig.5. Block diagram of Turbo decoder

Fig.6. Simulated performance of a rate r=1/2, constraint length K_c=5, turbo code for various numbers of decoder iterations. The size of interleaver is L=65,536 and an AWGN channel is assumed.
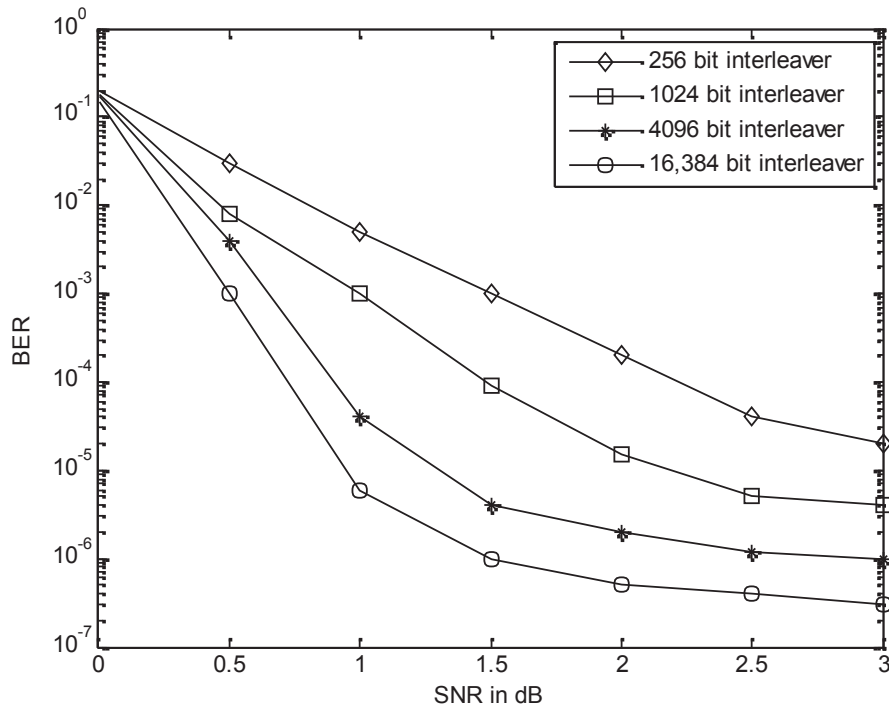

Fig.7. Bit Error Rate vs. E_b/N_o as parameterized by frame size for turbo code with rate 1/3, constraint length 3, and 8 iterations of improved decoding over an AWGN channel.
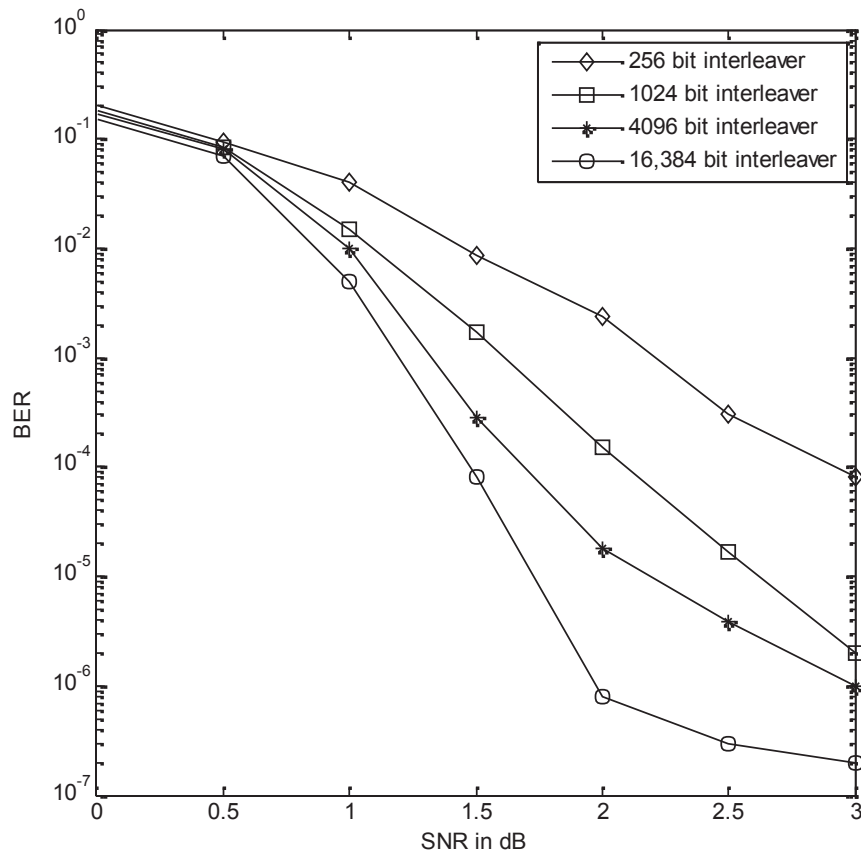
Fig.8. Bit Error Rate vs. E_b/N_o as parameterized by frame size for turbo code with rate 1/2, constraint length 3, and 8 iterations of improved decoding over an AWGN channel.