ISSN 2250-3153

Analysis of Turbo codes in Frequency Selective Fading Channel

Suchita Chatterjee, Mangal Singh

Department of Electronics & Telecommunication Engineering,
Chhatrapati Shivaji Institute of technology, Durg (C.G.)
suchitachatterjee@csitdurg.in

Abstract- In this paper, we discuss the bit error performance of turbo codes in Rayleigh fading channel. Turbo encoded signal is transmitted over a Rayleigh fading channel with additive white Gaussian noise. Training sequence is sent prior to data transmission in order to calculate the channel coefficients. After computing channel coefficients, it is assumed that channel characteristics are constant over a small interval of time. Data signal is determined using these coefficients and then turbo decoded. It reduces the bit error probability and a more reliable output is obtained.

Index Terms- Turbo codes, turbo encoder, Rayleigh fading channel, turbo decoder, MAP algorithm, likelihood ratio.

I. INTRODUCTION

Key system parameters for a communication system are transmitted power and channel bandwidth. These parameters determine the bit energy to noise ratio E_b/N_0. Practical considerations place a limit on the value that can be assigned to E_b/N_0. For a fixed E_b/N_0, only practical option for changing data quality from problematic to acceptable is by error control coding. Shannon's work promised the existence of theoretical performance improvement of 11.2 dB over the performance of uncoded systems. Although Shannon proved the theoretical limit at which error-free communications could take place using error-correcting codes, all previous coding schemes have fallen far short of this limit. Turbo codes are one of the most powerful types of channel codes that were introduced in 1993, along with error correction capability and a practical decoding algorithm. The importance of turbo codes is that they enable reliable communications with power efficiency close to the theoretical limit predicted by Claude Shannon [1].

Wireless communication systems have to overcome the problems of data transmission over fading channels. In the proposed method, the input data bits are encoded by a rate R turbo encoder. The resulting bits are binary phase shift keying modulated and transmitted over a Rayleigh fading channel with additive white Gaussian noise. Training sequence is used to find the time-varying properties of the channel prior to data transmission. A synchronised version of the training sequence is generated at the receiver, where it is applied to the equaliser as the desired response. The adaptation is achieved by observing the error between the desired response and actual response at the filter output and using this error to estimate the tap weights of the filter. By using these coefficients, the data signal is determined assuming channel characteristics are constant over a period of time. Then it is turbo decoded to obtain more accurate results.

II. SYSTEM MODEL

The system analysed in this paper consists of following blocks: Turbo encoder, Rayleigh fading channel, channel estimator, turbo decoder.

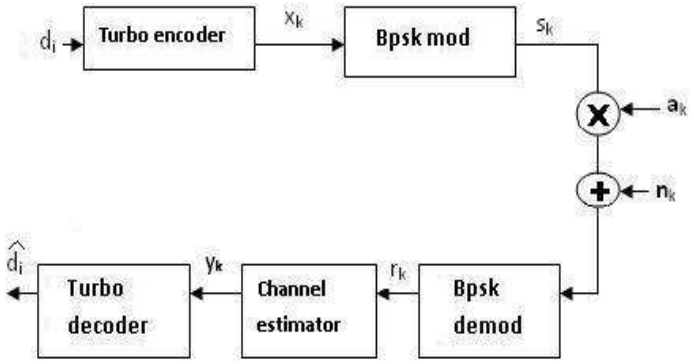

Figure 1: System Model

III. METHODOLOGY

A. Turbo encoding

The input data bits di, di € {0, 1} 1 <= i <=L are encoded by a rate R turbo encoder with an interleaver of size L. A turbo encoder consists of two identical recursive systematic convolutional encoders connected in parallel by an interleaver. Message bits are applied directly to encoder1 and reordered version of the same message bits is applied to encoder2. The more scrambled the information sequence for the second encoder, the more uncorrelated the information exchange between the decoders. The output of encoder consists of systematic bits (that is the original message bits) and two sets of parity-check bits (generated by the two encoders). The input data stream and the parity outputs of the two parallel encoders are then serialised into a single turbo codeword [2].

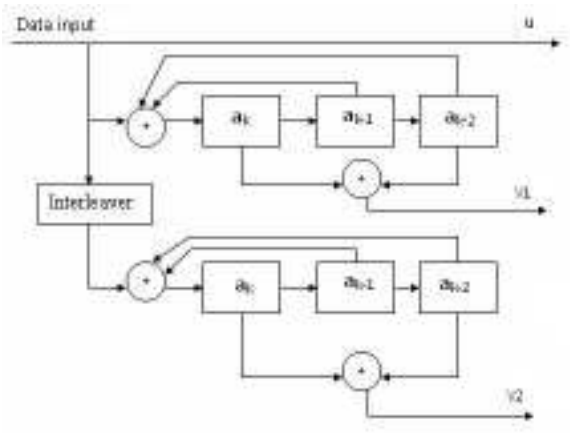

Fig. 2: Turbo Encoder

B. Interleaving

Interleaving is a process of rearranging the ordering of a data sequence in a one to one deterministic format. The inverse of this process is called de-interleaving which restores the received sequence to its original order. Interleaving is a practical technique to enhance the error correcting capability of coding. In turbo coding, interleaving is used before the information data is encoded by the second component encoder. The basic role of an interleaver is to construct a long block code from small memory convolutional codes, as long codes can approach the Shannon capacity limit. Secondly, it spreads out burst errors.

C. Fading channel

The encoded bits x_k , $1 \leq k \leq (L/R)$ are binary phase shift keying modulated, by doing discrete time base conversion to obtain s_k , $s_k \in \{-1, 1\}$.. Then it is transmitted over a Rayleigh fading channel with additive white Gaussian noise. Rayleigh fading is a reasonable model when there are many objects in the environment that scatter the radio signal before it arrives at the receiver. In Rayleigh fading model, the magnitude of a signal that passes through the transmission medium fades according to Rayleigh distribution. In this paper, multipath propagation of the signal is considered and applies different channel path delays. Then the signal is extremely faded. In Rayleigh distribution, the random variable S has a probability density function.

$$p_S(r) = \frac{2r}{\sigma^2} exp(\frac{-r^2}{\sigma^2}), r \geq 0 \quad (1)$$

where $\sigma^2 = E(S^2)$

D. Channel estimator

After BPSK demodulation, the received signal can be represented as

$$r_k = a_k s_k + n_k \quad (2)$$

where ak are Rayleigh distributed fading amplitudes and nk are samples of AWGN process.

A training sequence is transmitted through the Rayleigh fading channel. A synchronised version of the training sequence is generated at the receiver, where it is applied to the equaliser as the desired response. When the training process is completed, the adaptive equaliser is ready for normal data transmission. For a few consecutive frames, same channel coefficients are used by considering that channel characteristics are same for particular interval of time. Training sequence is repeated after a fixed time interval. Initially weight vector is computed as follows:
Received signal r(n) is fed to a Wiener filter of order N and with coefficients wi, i = 0; 1::::::N .
The output of the filter is denoted by y(n).

$$y(n) = \sum_{i=0}^{N} w_i r(n-i) \quad (3)$$

The Wiener filter is designed to minimise the mean square error

$$w_i = argminE\left\{e_1^2[n]\right\} \quad (4)$$

where e1(n) is the difference between filter output y(n) and reference input s(n). Filter coefficients are computed by solving Wiener Hopf equations, which has compact matrix formulation

$$\mathbf{T}\mathbf{w}_0 = \mathbf{P} \quad (5)$$

where T is the autocorrelation matrix of r(n) , P is the cross correlation vector of r(n) and s(n) and w0 is the coefficient vector.

$$\mathbf{w}_0 = \mathbf{T}^{-1}\mathbf{P} \quad (6)$$

Then the coefficient vector w0 is taken as the initial weight vector W(1). In Least Mean Square algorithm for adaptive equalisation, the error signal e(n) actuates the adjustments applied to individual tap weights of the equaliser as the signal proceeds from one iteration to the next. The (N + 1)_1 tap-weight vector,

$$W(n) = [w_0(n), w_1(n).....w_N(n)]^T \quad (7)$$

Firstly received training sequence is taken as the signal input

$$X(n) = [x(n), x(n-1), ...x(n-N)]^T \quad (8)$$

for n=1,2,.... compute filter output

$$y(n) = X^T(n)W(n) \quad (9)$$

W(n+1) is updated value of kth tap weight

$$W(n+1) = W(n) + \mu e(n)X(n) \quad (10)$$

W(n) is old value of kth tap weight, μ is step size parameter,

X(n) is input signal applied to kth tap weight, e(n) is error signal.
error signal

$$e(n) = d(n) - y(n) \qquad (11)$$

where d(n) is the desired response obtained from training sequence and y(n) is the actual response.

Take N as the filter order. Firstly, N samples of the input is taken. Using the filter coefficients obtained from Wiener Hopf equations, output y(n) is calculated. As the received sequence is complex valued, real part of y(n) is taken. Error between desired response and actual response is calculated. Updating of weight vector is continued until the error becomes 0.1. For the first N/2 samples, the step size μ is taken as 0.5 for faster convergence and for the remaining samples, μ is taken 0.02 to obtain correct weights. The same procedure is repeated for the remaining samples. The step size μ in the LMS algorithm governs the convergence rate, accuracy and stability of the adaptive filter.
For the few consequent received frames, the same weight vector is used to obtain the response. On comparing the received frames before and after equalisation, the number of errors is considerably reduced.

E. Turbo decoder

The process of turbo code decoding starts with the formation of a posteriori probabilities(APPs) for each data bit, which is followed by choosing the data bit value that corresponds to the maximum a posteriori (MAP) probability for that data bit [2]. Upon reception of a corrupted code bit sequence, the process of decision-making with APPs allows the MAP algorithm to determine the most likely information bit to have been transmitted at each bit time. There are two main algorithms to decode the data. One of the algorithms is called Soft Output Viterbi Algorithm(SOVA) and the other is Maximum Aposteriori Algorithm(MAP) [6].

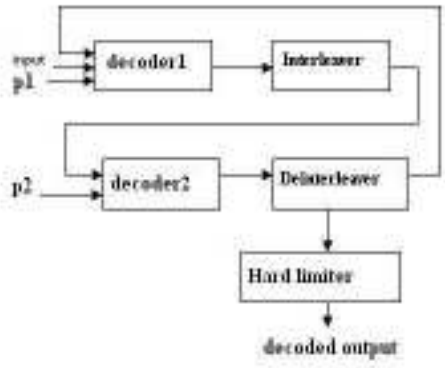

Fig. 3:Turbo decoder

The difference in MAP algorithm from the other algorithm is that while the former produces hard outputs, this one produces soft outputs. The MAP algorithm calculates the APPs for each code symbol produced by a Markov process given a noisy channel observation y. The APPs are P[dk = 1/y], the probability that the information bit is a 1 given the received vector y, an P[dk = 0/y], the probability that the information bit is a 0 given the received vector y. Once these probabilities are found, they are put into log-likelihood ratio (LLR) form in order to make a decision on a particular symbol. If the LLR L (d_k) is greater than zero, a binary 1 is chosen as the most likely transmitted symbol; conversely, if the LLR L (d_k) is less than zero, a binary 0 is chosen.

The iterative MAP turbo decoder consist of two component decoders serially concatenated via an interleaver. The first MAP decoder takes as input the received information sequence and the received parity sequence (p1) generated by the first encoder. The decoder then produces a soft output, which is interleaved and used to produce an improved estimate of the a priori probabilities of the information sequence for the second decoder. The inputs to the second MAP decoder are the interleaved received information sequence and the received parity sequence (p2) produced by the second encoder. The second MAP decoder also produces a soft output, which is used to improve the estimate of the a priori probabilities for the information sequence at the input of the first MAP decoder. The feedback loop is a distinguishing feature of this decoder. After a certain number of iterations, the soft outputs of both MAP decoders stop to produce further performance improvements. Then the last stage of decoding makes a hard decision after de-interleaving.

F. Bit error probability

$$P_b = B_d erfc\left(\sqrt{Rd_k(E_b/N_0)}\right) \qquad (12)$$

where B_d is bit errors/data bits, d_k is the weight of the kth codeword, R is the code rate, Eb/N0 is the bit energy to noise ratio.

IV. SIMULATION RESULTS

The performance of turbo coding system with channel estimation technique is evaluated through MATLAB simulations. The following results show the performance of turbo codes for different block sizes, channel path delays, constraint lengths and different frame sizes. The performance graph is plotted between bit energy to noise ratio and bit error rate.

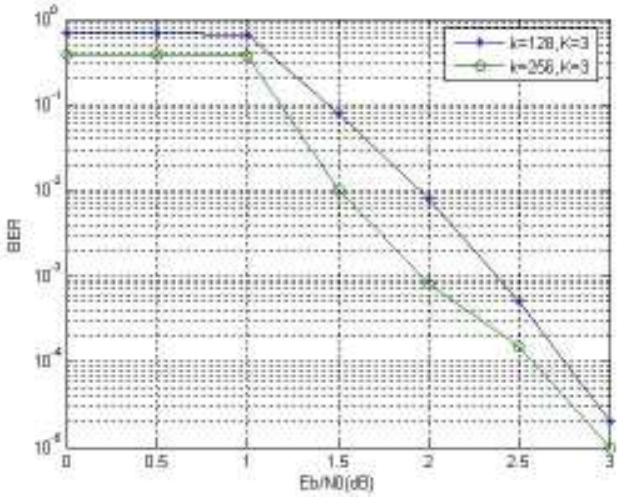

Fig.4: Simulation results for different block sizes and constraint length K=3

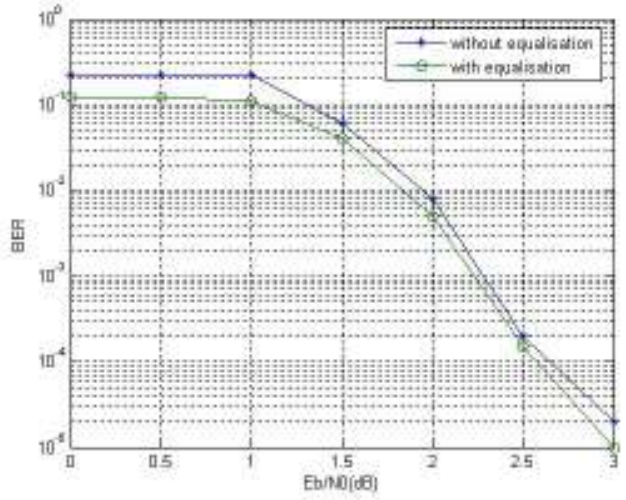

Fig. 5. Simulation results of turbo codes with and without equalisation

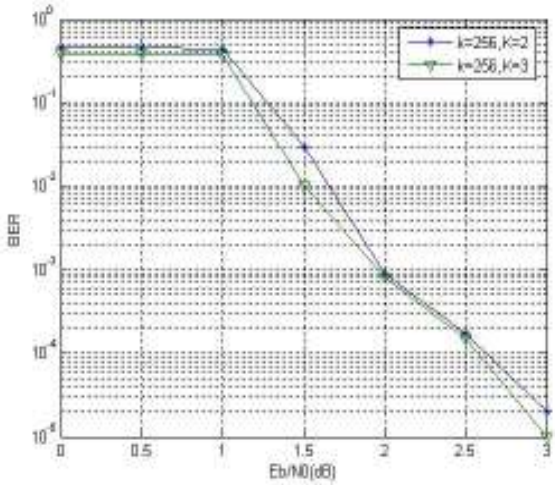

Fig. 6. Simulation results for same block size and different constraint lengths

V. CONCLUSION

This paper investigates the performance of turbo codes in Rayleigh fading channel. Turbo coding with channel estimation scheme improves the bit error performance. From simulation results, it is found that equalisation provides better results. Performance of turbo codes is better for large block sizes compared to small ones. Bit error rate of turbo codes with different path delays is compared. Future work can include channel estimation schemes for other channel models.

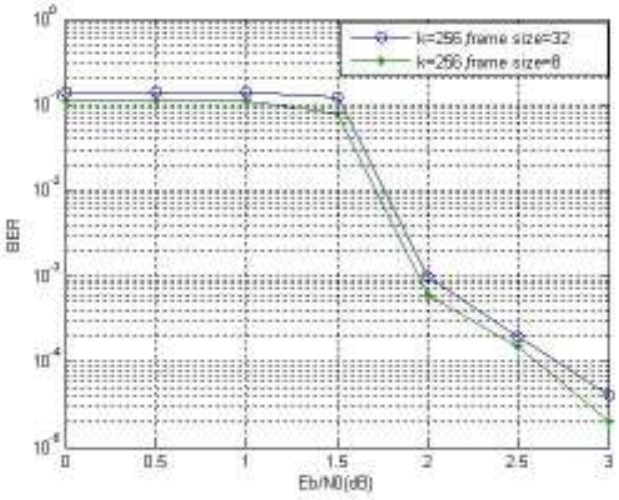

Fig. 7. Simulation results for same block size and different frame sizes

REFERENCES

[1] C. Berrou, A. Glavieux and P. Thitimajshima,"Near Shannon limit error correcting coding and decoding: Turbo-codes ", Proc. ICC'93, pp. 1094-1070, May 1993.

[2] B. Sklar,"A primer on turbo code concepts ", IEEE Commun. Mag., vol. 35, no. 12, pp. 94-102, Dec. 1997.

[3] B.Mielczarek and A.Svensson,"Modelling the influence of fading channel estimation errors on turbo code performance ", IEEE Trans. Commun., Oct. 2002.

[4] B. L. Yeap, T. H. Liew, J. Hamorsky and L. Harzo,"Comparitive study of turbo equalization schemes using convolutional, convolutional turbo and block-turbo codes ", IEEE Trans. Wireless Commun., Vol. 1, no. 2, pp. 266-273, Apr. 2002.

[5] C.Komninakis,R.D.Wesel,"Joint iterative channel estimation and decoding in flat correlated rayleigh fading, ", IEEE Journal on Selected Areas in Communications,vol.19,pp.1706-1717,Sept.2001.

[6] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv,"Optimal decoding of linear codes for minimizing symbol error rate ", IEEE Trans. Inf. Theory, vol. IT-20, no. 2, pp. 284-287, Mar. 1974.

[7] C. Berrou, A. Glavieux and P. Thitimajshima,"Near optimum error correcting coding and decoding:Turbo codes ", IEEE Trans. Commun., vol.44, pp. 1261-1271, Oct. 1996.