Modified Maximum A Posteriori Algorithm For Iterative Decoding of Turbo codes

Prof M. Srinivasa Rao[#1], Dr P. Rajesh Kumar[*2], K. Anitha[#3], Addanki Srinu[*4]

[#1]Professor, Department of ECE , Sir C R Reddy Engineering College, Eluru, AP, INDIA

[*2] Professor, Department of ECE , PVP Siddhartha institute of technology, Vijayawada, AP, INDIA

[#3] Associate Professor, Department of ECE , Sir C R Reddy Engineering College, Eluru, AP, INDIA

[*4] Post graduate student, Department of ECE , Sir C R Reddy Engineering College, Eluru, AP, INDIA

[1]msrao55@yahoo.com [2]rkpanakala@yahoo.com [3]komma_anitha22@yahoo.co.in

[4]addanki6@gmail.com

Abstract **:**

Turbo codes are one of the most powerful error correcting codes. What makes these codes so powerful is the use of the so-called iterative decoding or turbo decoding. An iterative decoding process is an iterative learning process for a complex system where the objective is to provide a good suboptimal estimate of a desired signal. Iterative decoding is used when the true optimal estimation is impossible due to prohibitive computational complexities. This paper extends the mathematical derivation of the original MAP algorithm and shows log likelihood values can be computed differently. The proposed algorithm results in savings in the required memory size and leads to a power efficient implementation of MAP algorithm in channel coding.

Key words: Turbo codes, Iterative decoding, Map algorithm, Memory savings

I. INTRODUCTION

Recently, the digital communications has been further strengthened by important developments in at least two specific areas.

One of the area is digital signal decoding and detection. When applied to turbo codes and low-density parity – check (LDPC) codes, iterative algorithms are capable of a approaching Shannon's channel capacity bound within a very small margin [1], [2].

This paper is concerned with dynamic analysis of iterative decoding for turbo codes. Our aim is to show how to analyze an iterative decoding process using a system theory based approach more specifically.

II. TURBO ENCODING

Ever since Shannon published his famous channel coding theorem [14] in 1948, the advances in the field of communications theory can in one way be viewed as a painstaking pursue for discovering practical coding and decoding algorithms which enable us to approach the Shannon capacity limit.

A turbo encoder is illustrated in Figure 1. It is a binary code, consisting of two Recursive systematic encoders G1(z) and G2(z), an interleaver. The two constituent encoders are typically the same and will be denoted by G(z). The two constituent encoders share the same systematic sequence xs.

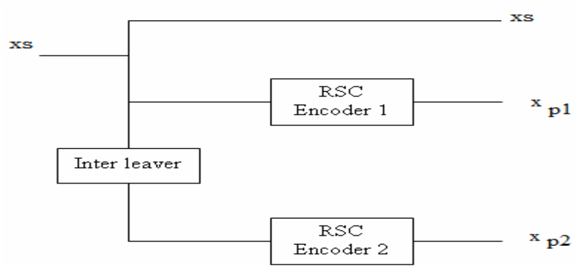


Figure 1

III. MAP DECODING

The turbo decoding algorithm is depicted in Figure 2. The input data to the turbo decoder are ys, y_{p1} and y_{p2}, which are the noisy version of xs x_{p1} *and* x_{p2} coming from demodulation.

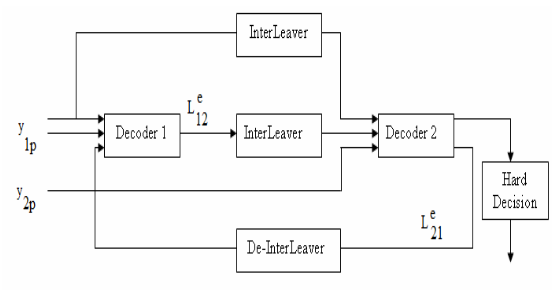


Figure 2

Using this method the likelihoods of different bits are computed and passed to the second decoder. The second decoder computes the likelihood ratios and passed the first decoder. The process is repeated until the likelihoods suggest high probability of correct decoding for each bit. Le12 and Le21 represent the information passed between the two decoders

IV. MODIFIED MAP ALGORITHM

The Map decoding algorithm is a recursive technique that computes the Log-Likelihood Ratio (LLR) of each bit based on the entire observed data block of length N.

$$y_k = (Ys.k, Y_{1p}.k); y_i^{\ j} = (y_1, y_2 ..., y_j); y = y_1^{\ n}$$

Where *Ys.k* is the k-th element of Ys, and Y_{1p}.k is similarly defined . A posteriori probability (APP) ratio of u, as defined below;

$$R(uk) = \frac{P(uk = +1 \mid y)}{P(uk = -1 \mid y)} \tag{1}$$

Where u_k is the k-th symbol of u.
The log-likelihood ratio (LLR) is given as

$$L(uk) = \log\left(\frac{P(uk = +1 \mid y)}{P(uk = -1 \mid y)}\right) \tag{2}$$

$$P(uk = +1 \mid y) + P(uk = -1 \mid y) = 1$$

Denoting the set of all possible states by S and the state at the k-th symbol by S_k. Now using the Bayes rule, we have

$$L(uk)=\log\left(\frac{\sum s+P(s_{k-1}=s^1,s_k=s,y)/P(y)}{\sum sP(s_{k-1}=s^1,s_k=s,y)/P(y)}\right) \tag{3}$$

It is clear from above that P(y) can be cancelled and we only need to find a way for computing $P(S_{k-1} = s^1, S_k=s, y)$, or $P(s^1,s,y)$ for short. By breaking Y into (y_{k-1} , y_k , y_{k+1}) and applying the Bayes rule again.

We can write

$$p(s^1,s,y)=\alpha_{k-1}(s^1).y_k(s^1,s).\beta_k(s) \tag{4}$$

$$\alpha_{k-1}(s^1)=p(s_{k-1}=s^1,y^1{}_{k-1}) \tag{5}$$

$$\gamma_k(s^1,s)=p(s_k=s,y_k\mid s_{k-1}=s^1 \tag{6}$$

$$\beta_k(s)=p(y^n{}_{k+1}\mid s_k=s) \tag{7}$$

These terms can be computed recursively using the Bayes rule again. More precisely,

$$\alpha k(s)=\sum_{s^1\in S}\alpha k-1(s^1)\gamma k(s^1,s) \tag{8}$$

With the initial conditions

$\alpha_0(s = 0) = 1;$

$\alpha_0\ (s \neq 0) = 0;$

Where s=0 is the known initial state for the code. Similarly,

$$\beta k-1(s)=\sum_{s\in S}\beta k(s)\gamma k(s^1,s) \tag{9}$$

With the terminating conditions

$\beta n\ (s = 0) = 1;\ \beta n\ (s \neq 0) = 0$

If s=0 is the known terminating state for the code. If the code is not terminated, $\beta n(s)$ is usually set equally.

It remains to compute $\gamma k(s^1,s)$, for which we have

$$\begin{aligned}\gamma(s^1,s)&=P(s\mid s^1)P(y_k\mid s^1,s)\\&=Pa(u_k)P(y_k\mid u_k,x_{p1.k})\end{aligned} \tag{10}$$

Where the values of u_k and $x_{pl.k}$ correspond to the transition from s^1 to s. The term $Pa(u_k)$ is the a priori probability of u_k which is related to the extrinsic information $L^e{}_{21}$ as follows :

$$\overline{L}^e{}_{21}.k=\log\left(\frac{P_a(uk=+1)}{P_a(uk=-1)}\right) \tag{11}$$

i.e., the *k-th* element of $L^e{}_{21}$ is the log a priori probability ratio for u_k.

For example, if the channel is an additive Gaussian white noise (AGWN) channel with noise variance σ^2, then $y_{s.k}$ and $y_{p1.k}$ are independent and we have

$$\begin{aligned}P(y_k/u_k,x_{p1}.k)&=P(y_s.k/u_k)P(y_{p1}.k/x_{p1}.k)\\&=C\exp\left[\frac{(y_{s.k}-u_k)^2}{2\sigma^2}\right]\exp\left[\frac{(y_{p1.k}-x_{p1.k})^2}{2\sigma^2}\right]\end{aligned} \tag{12}$$

With a constant C which does not affect L(uk). If we want to produce a hard estimate of u_k, we simply take

$$U_k=sign(L(u_k) \tag{13}$$

If we want to compute the extrinsic information L^e_{12} for further iterations, we simply subtract the input extrinsic information from $L(u_k)$, i.e.,

$$L^{e}_{12.k} = L(uk) - L^{e}_{21.k} \tag{14}$$

V. SIMULATION RESULTS

For Different cases the Simulation results are :

The figures (3, 4, 5 and 6) shows BER curves for the amount of data of 500, signal to noise ratio as up to 4 db, number of iterations are 5 and the interleaver lengths are taken as 50,100,150 and 200 respectively . Here interleaver lengths are only changed .

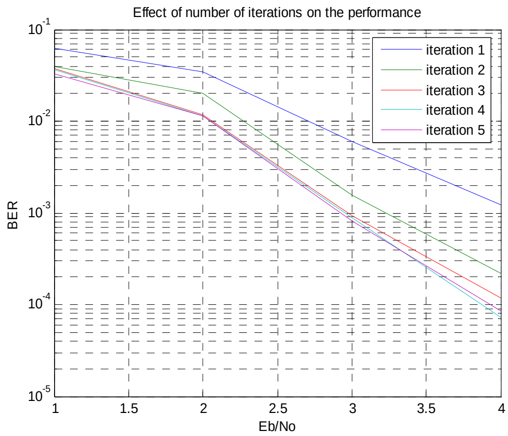


Figure 3

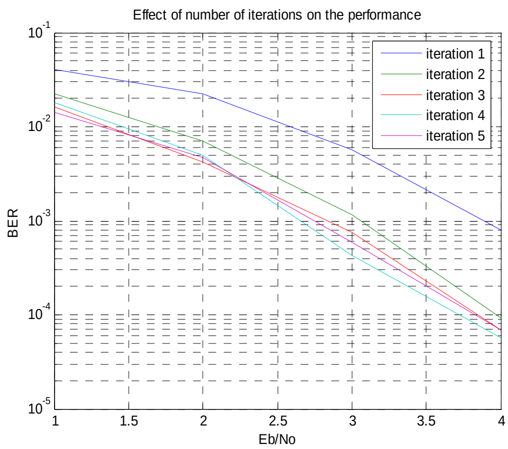


Figure 4

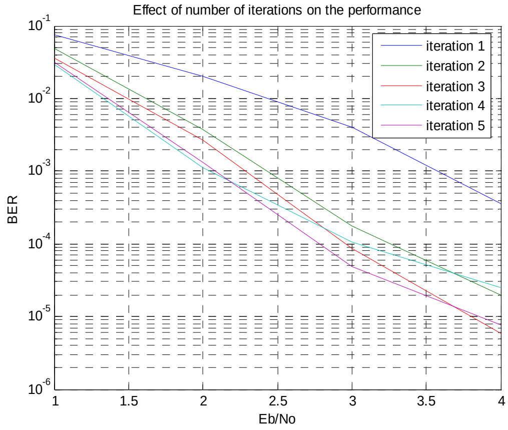


Figure 5

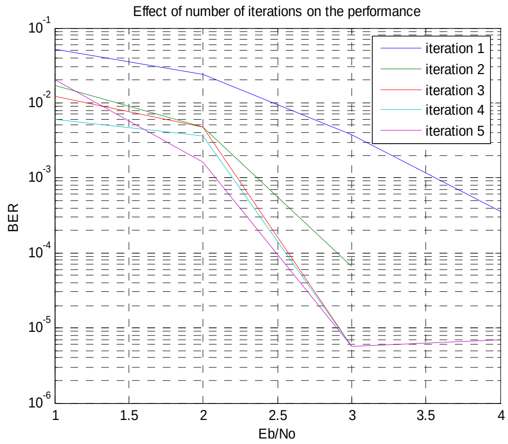
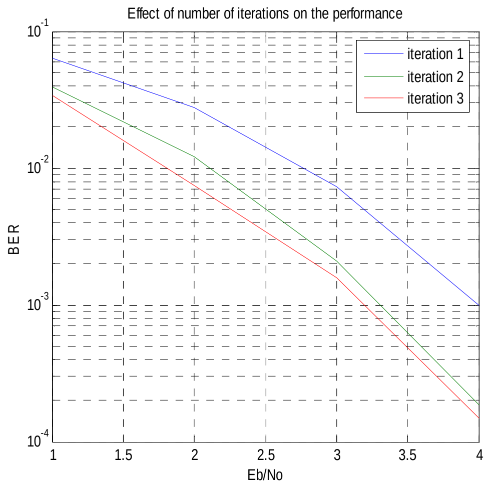


Figure 6

The below figures (7, 8, 9 and 10) shows BER curves for taking amount of data as 500 bits, signal to noise ratio as up to 4 db, the interleaver length is 50 and the number of iterations are taken as 3,4,5 and 6 respectively. Here the number of iterations only changed



.

Figure 7

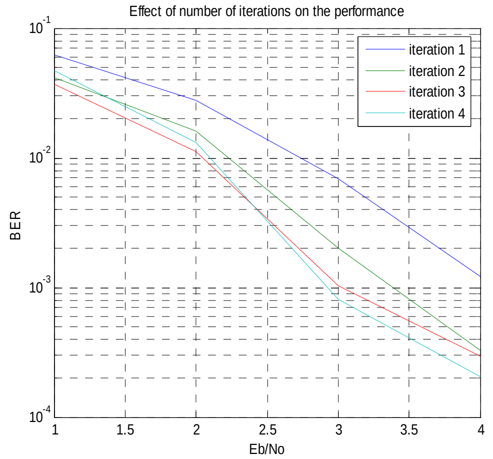


Figure 8

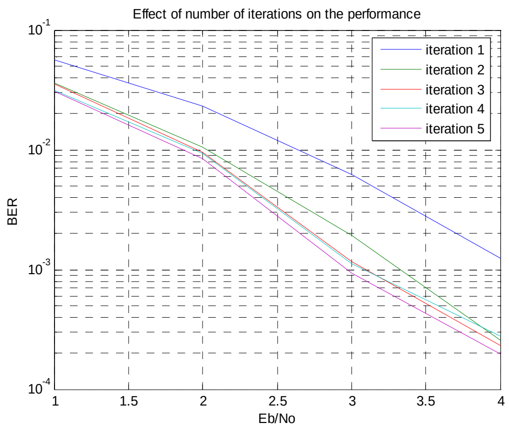


Figure 9

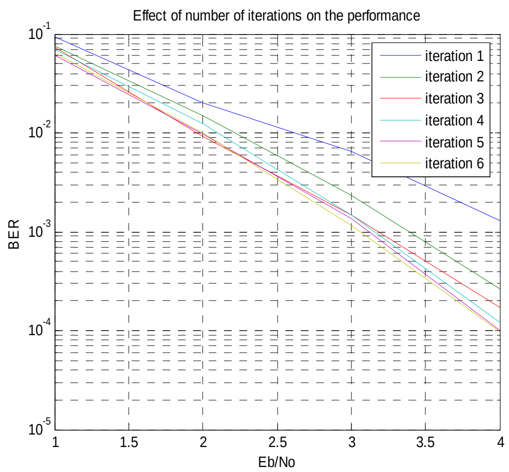


Figure 10

VI. CONCLUSION

From the simulation results we can notice two things.

1. The bit error probability decreases as the iterations goes up. This means as the iterations increase the reliability of the decisions taken increases.

2. The interleaver length also affects the performance. As the interleaver length increases the bit error probability decreases.

As BCJR algorithm is very complex, we are trying to modify the algorithm to save memory and to reduce complexity. The basic idea is as follows.

This paper extends the mathematical derivation of the original MAP algorithm and shows that the log likelihood values can be computed using only partial state metric values. By processing N stages in a trellis concurrently, the proposed algorithm results in savings in the required memory size and leads to a power efficient implementation of the MAP algorithm in channel decoding.

REFERENCES

[1] C. Berrou and A. Glavieux, "*Nea Optimum Error Correcting Coding and Decoding; Turbo Codes*," IEEE Trans. Comm., pp. 1261-1271, Oct 1996.
[2] S. Benedetto, G. Montorsi, D. Divsalar and F. Pollara , "*Serial Concatenation of Interleaved Codes: Performance Analysis, Design and Iterative Decoding*", TDA Progress Rep. 42-126, Apr-June 1996, Jet Rropulsion lab., Pasadena, CA, pp.1-26, Aug.15, 1996.
[3] L.Bahl, J.Cocke ,F.Jelinek, and J.Raviv, "*Optimum Decoding of linear codes for minimzing Symbol Error Rate*",IEEE Trans. On Inf. Theory, vol.IT-20, pp.284-287, Mar1974.
[4] P.Robertson, E.Villebrum, and P.Hoeher, "*A Comparision of Optimal and Sub-Optimal MAP Decoding algorithms Operating in the Log-domain*." International Conference on Communications, pp.1009-1013, June 1995
[5] S. Benedetto, G. Montorsi, D. Divsalar, F. Pollara ,"*Soft -Output decoding Algoitham in Iteratie Decoding of Turbo*
[6] *Codes*," TDA progress Report42-124, pp.63-87, February15,1996 [6]. J. Hagenauer, "*The turbo principle :Tutorial introduction and state of the art*," Proc. 1st Internet Symp. Turbo Codes, pp. 1-12, 1997.
[7] S. Brink, "Designing Iterative Decoding Schemes with the Extrinsic Information Transfer Chart," AEU Int. J. Electron. Commun., vol. 54, no.6, pp.389-398, 2000.
[8] S. Brink, J.Speidel, and R. Yan, "*Iterative demapping and decoding for multilevel modulation*," in Proc. Globecom'98, vol.1, pp.579-584, 1998.
[9] Forney,G.D :Concatenated Codes. Cambridge (Mass, USA) : MIT Press, 1966.
[10] T. Mittelholzer, X.Lin, J. Massey, "*Multilevel Turbo Coding for M-ary Quadrature and Amplitude Modulation*". Int. Symp. On Turbo Codes, Brest, September, 1997.
[11] Li, X; Ritcey, J. A: *Bit – Interleaved coded modulation with iterative decoding*. Proc. ICC (1999), 858-863.
[12] Cover, T. M. Thomas, J. A . *Elements of Information theory*. New York: Wiley, 1991.
[13] W.S.Wong and .W.Brockett, "*Sysytems with finite communication bandwidth constraints II: stabilization with limited information feedback*." IEEE Trans. Automatic Control, vol.44, no.5,pp.1049-1053,May 1999..
[14] D.J.C. MacKay and R. M.Neal, "*Near Shannon Limit performance of low- density arity- check codes*." Elecron. Lett., vol.32, pp.1645-1646. Aug.1996.