# ANALYSIS AND IMPLEMENTATION OF DECODING ALGORITHMS FOR TURBO CODES IN DIGITAL MAGNETIC RECORDING CHANNELS

Cătălin SANDU[1], Codruț ROTARU[2]

*Turbo codes decoding algorithms logic is analyzed in Digital Magnetic Recording (DMR) Partial Response (PR) channels environment. The performances of turbo decoding process are estimated, in the presence of appropriate DMR noise, generally a particulate noise, manifested as burst errors of various intensities and dimensions. Depending of decoding algorithms, turbo code performance (Bit Error Rate BER) is estimated, in presence of noise, over variations with its parameters. Optimal code length and structures design recommendations are obtained, related to required recording densities and Signal Noise Ratio (SNR) ranges.*

**Keywords**: turbo, magnetic channel, coding, decoding, noise

## 1. Introduction

A Turbo Code is a parallel concatenation of two or more convolutional codes with an interleaver between.

The Interleaver's role is to rearrange the input sequence, in other words it spreads the error. The basic coding schema for a Turbo Code with a total rate of 1/3, is presented below in Fig. 1.

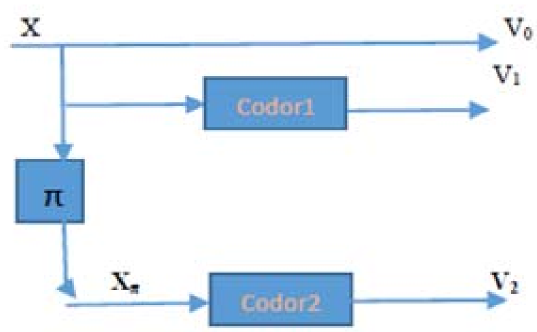


Fig.1 (Turbo Encoder)

[1] Prof., Dept. of Electronics Applied Electronics and Informatics Engineering, University POLITEHNICA of Bucharest, Romania, e-mail: cs@elia.pub.ro

[2] Student, Dept. of Electronics Applied Electronics and Informatics Engineering, University POLITEHNICA of Bucharest, Romania, e-mail: rotarucodrut@gmail.com

The output sequence is a multiplexing of the 3 output sequences, as seen in the following example:

$$\left.\begin{array}{l} X=[a, b, c\ldots] \\ V_0=[a, b, c\ldots] \\ V_1=[@, \#, \$\ldots] \\ V_2=[*, /, +\ldots] \end{array}\right\} \rightarrow V=[a\ @\ *, b\ \#\ /, c\ \$\ +\ldots]$$

Turbo Code decoding takes place between 2 decoders. The decoders exchange extrinsic information for a fixed number of iterations or until there is nothing to correct, but mostly it is used the first option (fig. 2).

The inputs of the decoders are the outputs of the coding schema altered by the AWGN noise. Depending on the formula for the extrinsic information, the decoder can be: SOVA decoders, MAP decoders, Log-MAP decoders, Max-Log-MAP decoders, Enhanced Log-Map decoders.

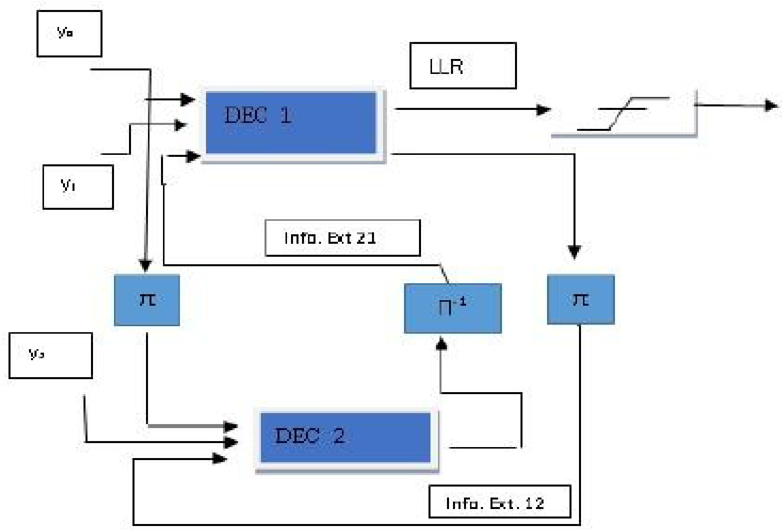


Fig. 2 (Turbo Decoder)

## 2. Decoding Algorithms

The **MAP** algorithm increases after each iteration, the information of the bit you are looking for. MAP is a Forward-Backward algorithm and it is using the LLR (Log-Likelihood Ratio) metric.

$$L(u_k) = \ln \frac{P(u_k = +1)}{P(u_k = -1)}, \tag{2.1}$$

The probabilities in relation (2.1) are rewritten depending on the received sequence y and the starting (s) and ending (s') state of each bit. Also the received sequence is broken in 3 parts: $y_p$(past), $y_k$(present), $y_f$(future). In the end with the help of the Bayes rule we obtain the following terms:

$\alpha_{k-1}(s') = P(s', y_p)$ - Forward metric

$\beta_k(s) = P(y_f / s)$ - Backward metric

$\gamma_k(s', s) = P(s, y_k / s', y_p)$ - Transition metric

Relation (2.1) became:

$$L(u_k) = \frac{\sum_{u_k=+1} \alpha_{k-1}(s')\beta_k(s)\gamma_k(s',s)}{\sum_{u_k=-1} \alpha_{k-1}(s')\beta_k(s)\gamma_k(s',s)}, \tag{2.2}$$

Considering an AWGN channel the transition metric's formula is:

$$\gamma(s',s) = A_k B_k \exp[\frac{1}{2} * L(v_k^1) * v_k^1 + L_c * \frac{1}{2} * y_k^{1,s} * v_k^1] \exp[\sum_{i=2}^{q} (L_c * \frac{1}{2} * y_k^{i,p} * v_k^i)], \tag{2.3}$$

Where we define a new metric (partial transition metric):

$$\gamma^e(s',s) = \exp[\sum_{i=2}^{q} (L_c * \frac{1}{2} * y_k^{i,p} * v_k^i)] \tag{2.4}$$

The $A_k$ , $B_k$ terms don't matter because they cancel each other in the LLR form. After formula manipulation the LLR form is:

$$L(u_k) = L(v_k^1) + L_c * y_k^{1,s} + \ln \frac{\sum_{u_k=+1} \alpha_{k-1}(s')\beta_k(s)\gamma^e(s',s)}{\sum_{u_k=-1} \alpha_{k-1}(s')\beta_k(s)\gamma^e(s',s)}, \tag{2.5}$$

$$\text{LLR} = \text{Lapriori} + \text{Lchannel} + \text{Lextrinsec}$$

First term, at the beginning is 0 and the second one doesn't change trough out the decoding process, because it depends only the channel type. From iteration to iteration the apriori information becomes the extrinsic information. An example is illustrated below:

First iteration: $L_1(u_k) = 0 + L_{channel} + L_{1,1}^e$

$$L_1(u_k) = L_{1,1}^e + L_{channel} + L_{1,2}^e$$
$$\text{Iteration 2: } L_1(u_k) = L_{1,2}^e + L_{channel} + L_{2,1}^e$$
$$L_1(u_k) = L_{2,1}^e + L_{channel} + L_{2,2}^e$$
$$\text{Iteration 3: } L_1(u_k) = L_{2,2}^e + L_{channel} + L_{3,1}^e$$
$$L_1(u_k) = L_{3,1}^e + L_{channel} + L_{3,2}^e$$

After a fixed number of iterations a decision is made depending on the sign of $L(u_k)$. This is called hard decision.

**Log-MAP** algorithm uses the Jacobian logarithm to approximate the branch metrics and the extrinsic information:

$$\ln(e^{x_1} + e^{x_2}) \cong \max(x_1, x_2) + \ln(1 + e^{-|x_1 - x_2|}) = \max(x_1, x_2) + f_c(x_1, x_2)$$

implies the modification of metric like this:

$$1)\ A_k(s) = \ln(\alpha_k(s)) = \max_{s'}(A_{k-1}(s) + \Gamma_k(s', s)) + f_{c\,s'}(A_{k-1}(s) + \Gamma_k(s', s))$$
$$2)\ B_{k-1}(s') = \ln(\beta_k(s')) = \max_{s'}(\beta_{k-1}(s') + \Gamma_k(s', s)) + f_{c\,s'}(\beta_k(s') + \Gamma_k(s', s))$$
$$3)\ \Gamma(s', s) = \ln(\gamma_k(s', s)) = C' + \frac{1}{2}u_k L(u_k) + \frac{Lc}{2}\sum_{l=1}^{n} u_{k_l} x_{k_l}$$

And now the LLR is:

$$L(u_k \mid y) = [\max_{(s',s)\to u_k=+1}(A_{k-1}(s') + B_k(s) + \Gamma^e(s', s)) + f_{c\,s'}(A_{k-1}(s') + B_k(s) + \Gamma^e(s', s))] -$$
$$-[\max_{(s',s)\to u_k=-1}(A_{k-1}(s') + B_k(s) + \Gamma^e(s', s)) + f_{c\,s'}(A_{k-1}(s') + B_k(s) + \Gamma^e(s', s))]$$

Max-Log-MAP algorithm uses a weaker approximation than Log-MAP:

$$\ln(e^{x_1} + e^{x_2}) \cong \max(x_1, x_2)$$

After this approximation the metrics are:

$$A_k(s) = \ln(\alpha_k(s)) = \max_{s'}(A_{k-1}(s) + \Gamma_k(s', s))$$
$$B_{k-1}(s') = \ln(\beta_k(s')) = \max_{s'}(\beta_{k-1}(s') + \Gamma_k(s', s))$$
$$\Gamma(s', s) = C' + \frac{1}{2}u_k L(u_k) + \frac{Lc}{2}\sum_{l=1}^{n} u_{k_l} x_{k_l}$$

$$L(u_k \mid y) = [\max_{(s^{'},s)\rightarrow u_k=+1}(A_{k-1}(s^{'}) + B_k(s) + \Gamma^e(s^{'},s))] - [\max_{(s^{'},s)\rightarrow u_k=-1}(A_{k-1}(s^{'}) + B_k(s) + \Gamma^e(s^{'},s))]$$

Enhanced Log-Map algorithm uses two constants to scale the extrinsic information and are introduce in the transition metric formula:

$$\Gamma(s^{'},s) = C^{'} + \frac{1}{2}u_k L(u_k) + s_i * \frac{Lc}{2}\sum_{l=1}^{n} u_{k_l} x_{k_l}, \text{ where } s_i = \{s_1, s_2\} = \{0.9;0.85\}$$

The modified turbo decoder schema is present in Fig. 3.

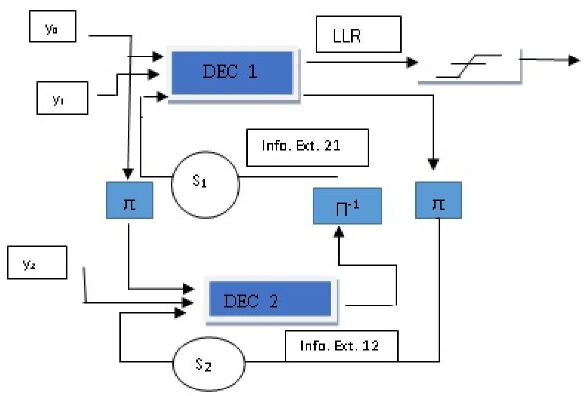


Fig. 3 Turbo Decoder for Enhanced Log-Map algorithm

The SOVA algorithm used the same LLR as MAP and takes a hard decision depending on the sign of the LLR. The probability of selecting the path with the minimal metric as being the path of maximum plausibility is proportional with:

$$\left.\begin{array}{l} P_r\{c_t = 1 \mid r_1^{\tau}\} \sim e^{-\mu_{r,min}} \\ P_r\{c_t = 0 \mid r_1^{\tau}\} \sim e^{-\mu_{t,c}} \end{array}\right\} \Rightarrow$$

$$\log\frac{P_r\{c_t = 1 \mid r_1^{\tau}\}}{P_r\{c_t = 0 \mid r_1^{\tau}\}} \sim \log\frac{e^{-\mu_{r,min}}}{e^{-\mu_{t,c}}} == \mu_{t,c} - \mu_{r,min}$$

where $\mu_t^1$ is the minimal metric for all paths with ct=1 and $\mu_t^0$ is the minimal metric for all paths with ct=0.

In the end the soft decision of the decoder will be the difference between the minimal metric for al paths with 0 and the minimal metric for all paths with 1.

### 3. CAMAG simulator

CAMAG is dedicated to magnetic recording channel analysis (PR4, EPR4 and E2PR4 partial response channels). The in house built platform is dedicated to the read/write process, for a suite of coding and decoding procedures, R/W parameters and interfering noises. In our case, CAMAG was patched with MAP and SOVA decoding of a stream of data input, randomly affected by controlled noise over a controlled length. The CAMAG diagram is show in Fig. 4.

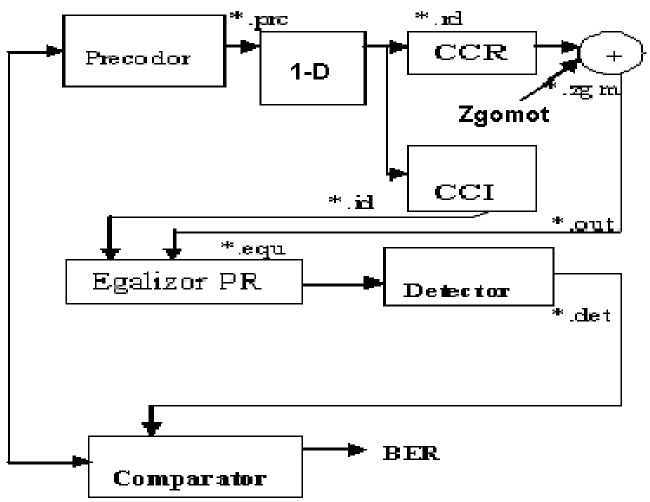


Fig. 4 CAMAG simulator diagram

The first block performs a precoding of data to reduce propagation of errors that could occur in detection. The formula is different for each channel: (1+D2) mod 2 for PR4 channel; (1-D+D2+D3) mod2 for EPR4 channel and (1-2D+2D3+D4) mod2 for E2PR4 channel. For read/write head we used 1-D operator to simulate the characteristic differential. In the CAMAG software we approximate magnetic channel behavior using Lorentzian function. The detection is Maximum A Posteriori, presuming that at each of some iterations set for the channel, an exchange of the extrinsic information is made between the two codes (the one with interleaved and the one without, both affected by noise). This way, after each iteration, the first decoder receives more information regarding the bit that is to be detected for each position, minimizing the chance of errors with each step taken. BER (Bit Error Rate) values are calculated.

### 4. Simulation results

The simulation results are average values after 10 simulations. A CAMAG simulation took place under the following parameters:

- AWGN file of 100 000 bits, average 0 and dispersion 1;
- Number of iterations: 4;

- Interleaver: Even/Uneven with file size 2048 bits;

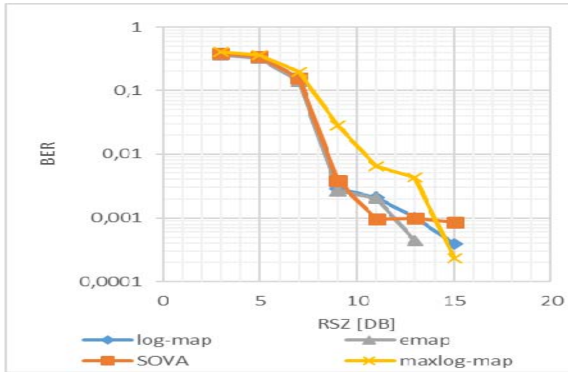


Fig. 5 CAMAG simulations results

## 5. Conclusions

Following the results, we concluded that Enhanced-Log-Map algorithm can be used with high confidence in magnetic recording. In terms of performance error correction again Enhanced-Log-Map algorithm is distinguished by the simple fact that in the same test conditions is the only one who corrects all errors for a SNR bigger than 13dB.

Another advantage of this algorithm is that it has approximately the same calculation complexity SOVA algorithms. As seen in theory SOVA algorithm calculates a difference with recursion before, and Enhanced-Log-Map algorithm still calculates a difference, but the forward and backward recursion. From this point of view Log-Map and Max-Log-Map are more advantageous to Map which is the most complex and difficult to implement. The downside of Max-Log-Map comes from the very poor approximation that it uses and this is visible in fig. 5. His only advantage is the low execution time. Under certain conditions (low noise> 16 dB) corrects all errors.

REFERENCES

[1]. *Stancescu, S, Sandu, C.* "Random-Interleaver-Design-în-Turbo-Coding-for-Digital-Magnetic-Recording-Channel ",MEET / MIPRO 2013 , p150.

[2]. *C. Berrou, A. Glavieux, and P. Thitimajshima*, "Near Shannon limit error- correctting coding and decoding: Turbo codes" Proc. 1993 Int. Conf. Commun., pp. 1064-1070.

[3]. *Victor Soare, Ştefan Stăncescu*, "Modular Partial Response Channel Software" MeleCon 2000, May 2000, pp. 786-789.

[4]. *Mustafa Taskaldiran, Richard C.S. Morling, and Izzet Kale*, "The Modified Max-Log-MAP Turbo Decoding Algorithm by Extrinsic Information Scaling for Wireless Applications", Springer US, pp. 203-213.

[5]. *Sadjadpour, Hamid R*. Maximum a posteriori decoding algorithms for turbo codes. In: AeroSense 2000. International Society for Optics and Photonics, 2000, p. 73-83.
[6]. *Robertson, P, Villebrun, E., Hoeher, P.* “A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain“ Proceedings of the International Conference on communications, Seattke, USA, pp. 1009-1013.
[7]. *Kovaci, Maria* ; Naforniță, Miranda “Contribuții la analiza şi îmbunătățirea performanțelor turbo codurilor în canale cu fading plat” Politehnica, 2009, p 159 .
[8]. *Gnanasekaran, T.; Aarthi, V.* “Performance enhancement of modified log MAP decoding algorithm for turbo codes” Communication and Computational Intelligence (INCOCCI), 2010 International Conference on. IEEE, 2010. p. 368-373.