International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 4 Issue 6, June-2015, pp: (259-265), Impact Factor: 1.252, Available online at: www.erpublications.com

A analytical approach on Zig-zag Convolutional Codes over AWGN Channel

Vijay Kumar[1], Shamsher Malik[2]

[1,2]Department of Electronics and Communication Engineering, University Institute of Engineering and Technology, M.D.U. Rohtak, Haryana India

---

Abstract: **This paper presents the design and evaluation of a new class of low turbo codes with reduce the decoder complexity known as modified turbo codes (MTC). Decoder complexity is the main consideration in the design of turbo codes. This research work presents two types of modified turbo codes, Low Complexity Hybrid turbo codes (LCHTC) and Improved Low Complexity Hybrid Turbo Codes (ILCHTC). Proposed LCHTC uses parallel concatenation of constituent encoders. Parallel concatenation of RSC codes has main advantages that if one of the output code words has low weight, the other usually does not, and there is a smaller chance of producing an output with very low weight. Higher weight is beneficial for the performance of the decoder First few constituent encoders are designed using serial concatenation of zigzag codes with convolutional codes and remaining encoders are zigzag encoders. Parallel concatenation of convolutional codes with zigzag codes has been presented for ILCHTC to reduce decoding complexity of overall decoder. LCHTC, ILCHTC and turbo convolutional codes (TCC) are simulated using Matlab simulation over Additive White Gaussian Noise (AWGN). LCHTC and ILCHTC shows same Bit Error Rate (BER) as shown by TCC but with reduced decoding complexity as compared to TCC. Rate (R) = 1/3 ILCHTC shows a BER ≈ 4×10^{-6} and R = 1/3 LCHTC shows a BER ≈ 8×10^{-6} at E_b/N_o = 2 dB over AWGN channel. Analytical study give result that computations required by R = ½ ILCHTC and R = ½ LCHTC is reduced by a factor of approximation three.**

Keywords: **Coding and decoding complexity, parallel concatenated codes, iterative decoding, zig-zag codes and error convergence.**

I. INTRODUCTION

Forward error correction (FEC) technique, introduces redundant bits into the information bits to correct error occurred during transmission. TCC codes are excellent error correcting codes which achieve BER near Shannon's limit [1], [2] using aposteriori probability (APP) iterative decoding. However APP algorithm is highly complex. If maximum a posteriori probability in log domain (Log-MAP) [3], [4] algorithm is used, the decoder complexity in term of addition equivalent operations per information bit per iteration (AEO/IB/I) is about 512. TCC consist of parallel concatenation of covolutional codes connected using interleaver. It has been shown that if convolutional codes and block codes are used for concatenation, performance near the theoretical limit is achieved, however such codes require less decoder complexity [4], [5]. Such codes are termed as modified turbo codes. In proposed LCHTC and ILCHTC, zig-zag codes are used as constituent code with recursive systematic convolutional (RSC) codes for concatenation. Zig-zag code shows BER performance near theoretical limitsusing soft-in soft-out (SISO) decoding algorithm with much less decoding complexity [6], [7] buterror convergence is low for zig-zag codes.

Proposed LCHTC codes show BER performance comparable to that of TCC. In LCHTC encoder, first zig-zag encoder encodes each row of information bits then R = ½ RSC encoder encodes these zig-zag parity bits. In LCHTC decoding, first RSC decoder decodes zig-zag parity bits then zig-zag decoder decodes information bits [6], [7]. Global iterative decoding is used. LCHTC decoder shows more decoding complexity than zig-zag codes but less decoding complexity than TCC. Zig-zag parity bits instead of information bits are decoded using RSC decoder in LCHTC so error convergence of LCHTC is slow.

ILCHTC encoder encodes information bits using zig-zag encoder as well as RSC encoder. In ILCHTC encoder, first RSC encoder compute parity bits for L column of information bits then zig-zag encoder compute parity bits for each row of information bits. In ILCHTC decoder, first information bits are decoded using RSC decoderthen zig-zag decoder decode information bits. Decoder complexity of ILCHTC is more than LCHTC and less than TCC. Error converges of ILCHTC is faster than LCHTC. BER ≈ 4×10^{-6} is shown by R = 1/3 ILCHTC at E_b/N_o = 2dB which is 0.4 dB away from E_b/N_o for the same BER for R = ½ TCC. BER ≈ 8×10^{-6} is achieved by R = 1/3 LCHTC at E_b/N_o = 2dB which is 0.5 dB away from E_b/N_o for the same BER for R = ½ TCC.

Unlike ILCHTC and LCHTC, in TCC decoder complexity remain unchanged after using puncturing to change the code rate. Paper is organised as follows: Section II describes turbo codes. Section III, presents the description of LCHTC. Section IV, presents the description of ILCHTC. In section V, simulation results and discussions are shown. Conclusions are drawn in section VI.

II. TURBO ENCODER AND DECODER

Here TCC encoder and decoder structure has been described.

1. TCC encoder

TCC consists of parallel concatenation of RSC encoders using random interleaver between them. RSC encoder 1 and RSC encoder 2 encode information bit sequence using a 16 state generator polynomial (G) = [5, (37 21),37] [1], [2],[8].

The structure of TCC encoder is shown in the fig.1.

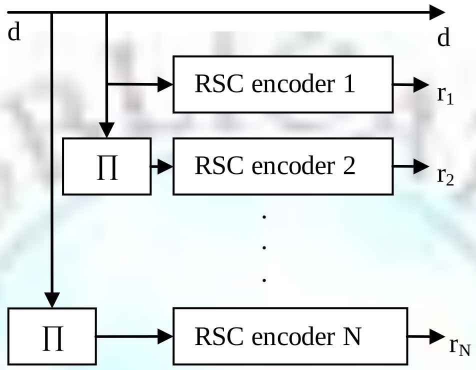

Figure 1: TCC encoder where ∏ is interleaver [9].

2. TCC decoder

SISO decoders are used for iterative decoding of turbo convolutional codes [1], [2], [3]. APP decoding algorithm to evaluate posteriori likelihood ratio (LRs) [3], [4] of received information bits is used and LRs can be given as:

$$LR(b'/b) = \frac{Pr(b'/b=+1)}{Pr(b'/b=-1)} \quad ... (1)$$

And their logarithms (LLRs) can be given as:

$$L = Log\big(LR(b'/b)\big) \quad ... (2)$$

III. LCHTC ENCODER AND DECODER

Information bit sequence of size N×1 is arranged row wise in an array of size I×J. Information bit can be represented as:

$$D = d(i,j), \quad 1 < i < I \text{ and } 1 < j < J \quad ... (3)$$

Also,

$$N = I \times J \quad ... (4)$$

1. Zig-Zag encoder

Data nodes d (i, j) are the information bits and $z^{(n)}(i)$ represent zig-zag parity nodes of n^{th} constituent encoder. Parity bits are computed progressively as follows:

$$z^{(n)}(1) = \sum_{j=1}^{J} d(1,j) \bmod 2 \quad ... (5)$$

International Journal of Enhanced Research in Science Technology & Engineering, ISSN: 2319-7463
Vol. 4 Issue 6, June-2015, pp: (259-265), Impact Factor: 1.252, Available online at: www.erpublications.com

And

$$z^{(n)}(i) = \sum_{j=1}^{J} d(i,j) + z^{(n)}(i-1) \bmod 2 \quad \dots (6)$$

Clearly zig-zag code are uniquely characterised by two parameters (I, J) [6], [7].

2. LCHTC encoder

LCHTC encoder consists of parallel concatenation of zig-zag codes with RSC codes. Fig.2 shows structure of LCHTC constituent encoder. For each constituent encoder, first zig-zag encoder computes parity bits for each row of information bits then zig-zag parity bits are encoded using rate ½ RSC encoder. Let zig-zag parity bits of n^{th} zig-zag encoder are represented by $z^{(n)}$ and parity bits of n^{th} RSC encoder are given by $r^{(n)}$. Code word C_L and code rate R_L for n^{th} constituent encoder is given by:

$$C_L = \{d, z^{(1)}, z^{(2)}, \dots, z^{(n)}, r^{(1)}, r^{(2)}, \dots, r^{(n)}\} \dots (7)$$

$$R_L = J/(J + (N + 2)) \quad \dots (8)$$

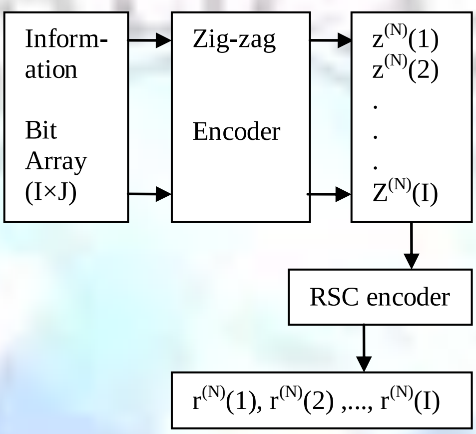

Figure 2: Nth constituent encoder for LCHTC [10].

Code rate can be adjusted by changing J and N. Here N is No. of constituent encoders. In overall LCHTC encoder interleaver is used except for the first encoder.

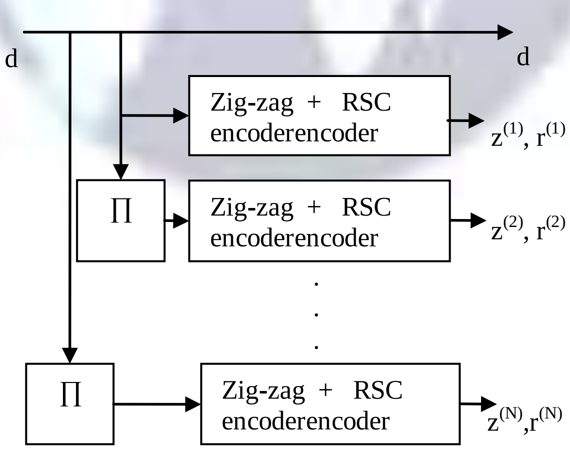

Figure 3: Overall LCTC encoder [9].

3. LCHTC decoder

Let R' represent the received noisy vector as, R'={d',z',r'}. Here d' is the received noisy data vector, z' received noisy zig-zag parity vector and r' is the received noisy convolutional parity vector. APP decoding algorithm to evaluate posteriori likelihood ratio (LRs) of received information bits is used as given by eq. (1). And their logarithms (LLRs)

[3], [4] can be given by eq. (2). SISO APP decoding is used for decodingof LCHTC codes. Decoding algorithm can be implemented in two steps as follows:

(i) Decoding of zig-zag parity bits is performed using RSC decoder.
(ii) Zig-zag decoder decode information bits taking out-put of convolutional decoder as input with nosy information bits (d') as another input [6], [7].

Iterative decoding process is used for decoding.

IV. ILCHTC ENCODER AND DECODER

1. ILCHTC encoder

First L columns of information bits are encoded using R = ½ RSC encoder and each row of information bits is encoded using zig-zag encoder. For R = 1/3 ILCHTC first constituent encoder encodes all J columns of information bits. Constituent encoder and overall encoder for ILCHTC are shown in fig. 4 and fig. 5 respectively. Let $r_j^{(n)}$ represents the convolutional parity bits for j^{th} row of the n^{th} constituent encoder. Let C_{IL} represent the transmitted code word and R_{IL} represent code rate for ILCHTC encoder then C_{IL} and R_{IL} can be given as

$$C_{IL} = \{d, r_1^{(n)}, r_2^{(n)}, \dots, r_L^{(n)}, z^{(1)}, z^{(2)}, \dots, z^{(n)}\} \dots (9)$$

$$R_{IL} = J/(J + L + N) \dots (10)$$

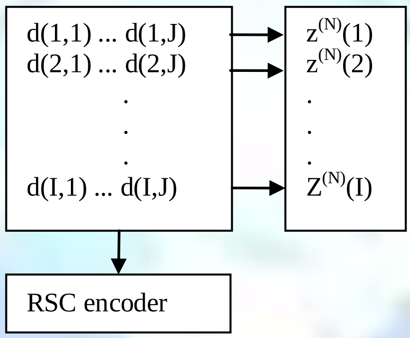

Figure 4: Nth constituent encoder for ILCHTC [10]

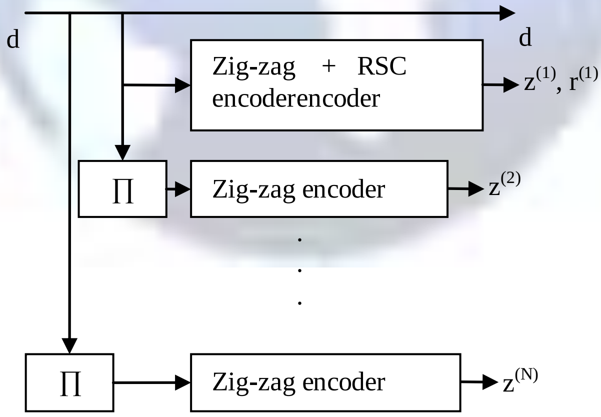

Figure 5: Overall ILCHTC encoder [9].

2. ILCHTC decoder

Let R' represent the noisy received code word as, R'={d',r',z'}. Here d' represents noisy received data bits, r' represents noisy received RSC parity bits and z' represents the noisy received zig-zag parity bits. APP decoding algorithm is used to evaluate LRs and LLRs given by eq. 1 and eq. 2 respectively. SISO decoding can be performed in two steps as:

(i) L columns of information array are decoded using convolutional decoder using Log-MAP algorithm.
(ii) Taking the result of convolutional decoder zig-zag decoder decodes each row of information array.

Overall iterative decoding is performed to improve the BER.

V. Decoder Complexity for LCHTC and ILCHTC

For TCC decoder complexity depends on trellis length T_L of convolutinal code [10]. Let N_m is multiplication required per information bit per iteration (M/IB/I) and N_a is the addition required per information bit per iteration (A/IB/I). For ILCHTC and LCHTC decoder complexity is less due to use of zig-zag code which has lower decoding complexity as compared to TCC. For ILCHTC trellis length depends on No. of column L, encoded by RSC encoder, highest for the constituent encoder for which L=J. For TCC trellis length is equal to 2N where N is No. of information bits.

N_m and N_a for zig-zag decoder [6] can be given as follows:

$$N_M = 0 \quad \text{... (11)}$$

$$N_a = N(4 + 4/J) \quad \text{... (12)}$$

For TCC N_m and N_a [10] can be given as follows:

$$N_m = 8 \times N_s \times N \quad \text{... (13)}$$

$$N_a = (16 \times N_s + 2) \times N - 2 \quad \text{... (14)}$$

M/IB/I and A/IB/I for different decoder is shown below in the table 1.

TABLE 1: Computational Complexity of Decoder

Decoder	R	Parameters	N_m	N_a
TCC	1/2	N_s = 16, N = 2 (punctured)	248	512
	1/3	N_s = 16, N = 2	248	512
LCHTC	1/2	N_s = 16, N = 2, J = 4, L = 2	62	139
	1/3	N_s = 16, N = 4, J = 4, L = 4	124	278
ILCHTC	1/2	N_s = 16, N = 2, J = 4, L = 2	62	139
	1/3	N_s = 16, N = 4, J = 4, L = 4	124	278

It is shown in the table 1 that number of M/IB/I and A/IB/I is nearly half for rate R = 1/3, ILCHTC and LCHTC as compared to TCC for N constituent encoder and computations required for R = ½, LCHTC and ILCHTC is nearly one third as compared to TCC. N_s represent No. of state for RSC encoder.

VI. Simulation Results

A 16 state encoder is used for RSC codes with generator polynomial G = [5, (37, 21), 37] [8]. Fig. 6 shows the BER at different E_b/N_o for R = ½ LCHTC and R = 1/3 LCHTC. Fig.6 shows that a BER ≈ 8×10^{-6} is achieved at E_b/N_o = 2dB for R = 1/3 LCHTC and a BER ≈ 10^{-4} is achieved at E_b/N_o = 2dB for R = ½ LCHTC.

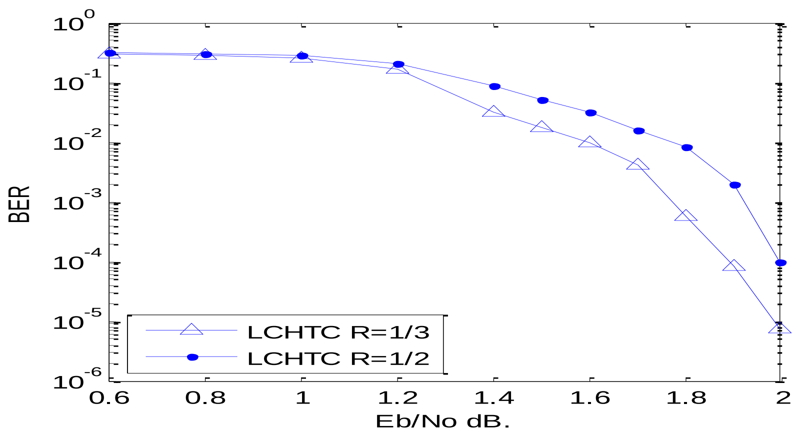

Figure 6 BER performance of LCHTC.

Fig. 7 shows the BER performance of R = ½ and R = 1/3 ILCHTC. Simulation result indicate that BER ≈ 4×10^{-6} is achieved at E_b/N_o = 2dB for R = 1/3 ILCHTC. Different parameters for R = 1/3 ILCHTC are J = 4, L = 4 and number of constituent encoder N = 4. BER ≈ 3×10^{-5} is achieved at E_b/N_o = 2dB for R = ½ ILCHTC with J = 4, L = 2 and N = 2.

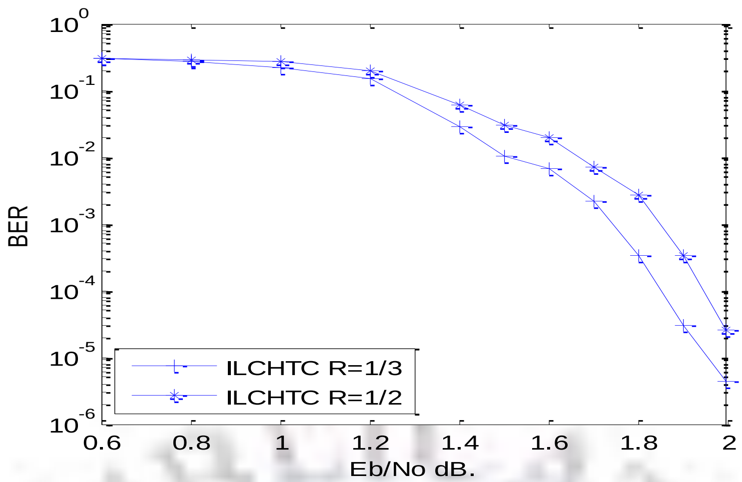

Figure 7 BER performance of ILCHTC.

Comparison of BER for TCC, LCHTC and ILCHTC in AWGN channel is illustrated in fig.8. ILCHTC shows better error convergence than LCHTC. In term of BER performance, ILCHTC is better than LCHTC because in ILCHTC information bits are decoded using zig-zag decoder as well as RSC decoder and the RSC decoder shows better error convergence than zig-zag decoder. While in LCHTC, first zig-zag parity bits are decoded using RSC decoder, then using these parity bits zig-zag decoder decode information bits.

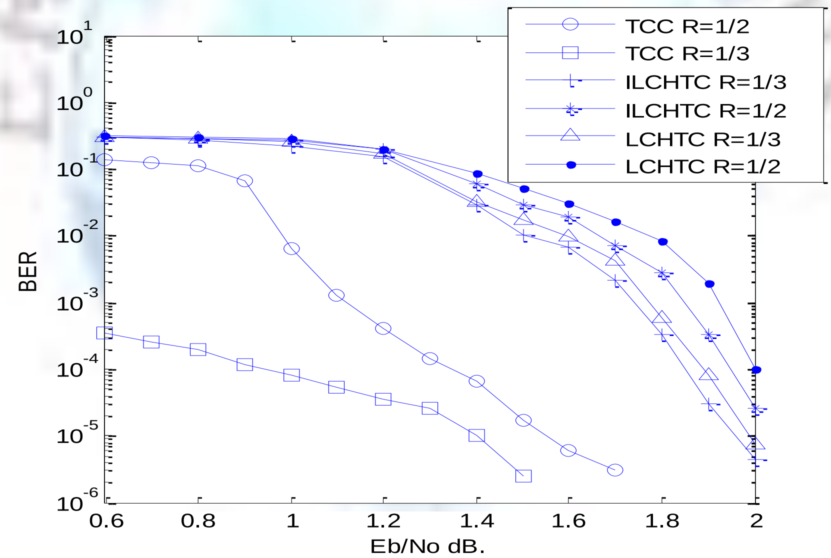

Figure 8: BER performance of TCC, LCHTC and ILCHTC.

Simulation results show that a BER ≈ 8×10^{-6} at E_b/N_o = 2 dB is achieved for R = 1/3 LCHTC which is 0.5 dB away from the E_b/N_o for the same BER for R =½ TCC. BER ≈ 4×10^{-6} is achieved at E_b/N_o = 2 dB for R =1/3 ILCHTC which is 0.4 dB away from the E_b/N_o for the same BER for R = ½ TCC.

CONCLUSION

This paper presents multiple concatenations of zig-zag code with RSC code to design low complexity turbo code termed as MTC. BER ≈ 10^{-4} at E_b/N_o = 2 dB is achieved for R = ½ LCHTC which is 0.1 dB away from the E_b/N_o for the same BER for R = ½ ILCHTC. ILCHTC shows BER slightly improved than LCHTC and error convergence is also better than LCHTC. Decoder complexity is reduced by a factor of nearly 2 for R = 1/3 ILCHTC and R = 1/3 LCHTC as compared to TCC. For R = ½ MTC, decoder complexity is reduced by a factor of nearly three as compared to TCC. Error convergence of TCC is best. ILCHTC and LCHTC require more number of iteration than TCC to get significant BER but overall decoding complexity is less than TCC.

REFERENCES

[1]. C. Berrou, A. Glavieux, P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in proc. IEEE ICC'93, vol. 2, pp.1064-1070, May 1993.

[2]. S. Benedetto and G. Montorsi, "Unveiling turbo codes: Some result on parallel concatenate decoding scheme," IEEE Trans. Inform. Theory, vol. 42, pp. 409-428, Mar. 1996

[3]. T. Gnanasekaran and V. Aarthi, "Performance Enhancement of Modified Log Map Decoding Algorithm for Turbo Codes,"Proc. IEEE conference INCOCCI, pp. 368-372, Dec. 2010.

[4]. Li. Ping, "Turbo-SPC Codes," IEEE Trans. Comm., vol. 49, No. 5, pp. 754-759, May 2001.

[5]. Keying Wu and Li Ping, "An Improved Two-state Turbo-SPC Codes for Wireless Communication System," IEEE Trans. Comm., vol. 52, No. 8, pp. 1238-1241, Aug. 2004.

[6]. Li Ping, X. Huang and N. Phamdo, "Zigzag Codes and Concatenated Zigzag Codes," IEEE Trans. Inform. Theory, vol. 47, No. 2, pp. 800-807, Feb. 2001.

[7]. Xiaofu Wu, YingjianXue and Haige Xiang, "On Concatenated Zigzag Codes and Their Decoding Schemes," IEEE Comm. Letter, vol. 8, No. 1, pp. 54-56, Jan. 2004.

[8]. S. Benedetto, R Garello and G. Montorsi, "A Search for Good Convolutional Codes to be Used in the Construction of Turbo Codes," IEEE Trans. Comm., vol. 46, No.9, pp. 1101-1105, Sept. 1998.

[9]. Dr. D. J. Shah, Prof. Vijay K. Patel and Prof. Himanshu A. Patel, "Performance analysis of Turbo Code for CDMA 2000 with convolutional coded IS-95 System in Wireless Communication System," in proc. IEEE ICECT 2010, pp. 42-45, 2010.

[10]. ArchanaBhise and Prakash D. Vyavahare, "Low Complexity Hybrid Turbo Codes," in Proc. IEEE WCNC' 2008, pp. 1050-1055, Mar. 2008.

[11]. A. Banerjee, F. Vatta and B. Scanavino, "Nonsystematic Turbo Codes," IEEE Trans. Comm., vol. 53, No. 11, pp. 1841-1849, Nov. 2005.

[12]. John G. Proakis, Digital Communications, 4th Edition, McGraw-Hill International Edition, 2001.

[13]. Molisch, wireless communications, Wiley publication Ltd. 2nd edition , 2011

[14]. C.heegard and S. B. Wicker, Turbo Coding, Kluwer Academic Publishers, 1999.

[15]. KeattisakSripimanwat, Turbo Code Applications, Springer Publications, 2005.

[16]. Claude Berrou, Codes and Turbo Codes, Springer Publications, 2010.