ROBUST WIRELESS IMAGE TRANSMISSION USING ASYMMETERIC TURBO CODES

[1]G. Vijay Kumar,[2](K. Bhavana, N. Sai Teja, G.Veera Venkata Lakshmi)
Assistant Professor, Dept. Of ECE, Andhra Loyola Institute of Engineering and Technology
Vijayawada, Andhra Pradesh, India.
Btech scholars, Dept. Of ece, Andhra Loyola Institute of Engineering and Technology
Vijayawada, Andhra Pradesh, India.

Abstract: Today's world thrives on information exchange. Hence the need of the day is the information be protected well enough to the transmitted over a noisy environment. This is achieved by adding redundant bits to the information bit streams. If the purpose of adding redundancy bits is just to detect error and inform the sender to re-transmit the information. Forward error correction(FEC) is another way of adding redundancy to the information bit stream. So, error can be detected and corrected by preventing needed re-transmission.

Turbo codes is a very powerful error correction technique that has made tremendous impact on channel coding in last few years. Turbo code bit error rate drops very rapidly with increasing Eb/N0 values. It achieves 10^-5 BER with Recursive Systematic Convolution encoder .The iterative decoding mechanism,RSC and use of inter-leaver are the characteristics features of turbo codes.That it enhance data transmission efficiency in digital communication system.Turbo codes play a major role in multimedia services in mobile phones.The performance of turbo codes is superior with a little Eb/No.

The original JPEG(Joint Photography Expert Group) image is encoded using turbo codes and subjected to additive white gaussian noise.In this random inter-leaver is used and MAP decoding algorithm is used. We can almost retrieve original image by number of iterations by iterative decoder. As the number of iterations increases the noise in image removed.

IndexTerms - *Turbo Coding, Forward error correction, Interleaving, puncturing, Iterative decoding, MAP decoding.*

I. INTRODUCTION

Turbo Code proposed in 1993 by Berrou *et al*, is known for excellent coding gain. It provides the error free communication near to Shannon Limit at great extent. Due to many research efforts of the turbo coding community, it is used in standardized system such as third-generation (3G) mobile radio system and so many other emerging wireless Applications.

Basically, the Turbo code can be classified into two types based on their generator polynomial structures. The component with identical encoders is basically known as symmetric turbo codes, otherwise asymmetric turbo code. The parallel concatenated turbo codes can assumes identical component code, as in the Symmetric turbo codes, have either a good “waterfall” Bit Error Rate (BER) performance or a good “error floor” BER performance but not both. Since, the asymmetric turbo code uses non identical component codes and can be designed with proper selection of weight configuration for better BER performance. In this paper, several new classes of asymmetric turbo codes are introduced which improves the performance compared to the original turbo codes (symmetric) over the entire range of signal to noise ratios. A practical setup with symmetric and asymmetric turbo codes is described and the performance results are discussed.

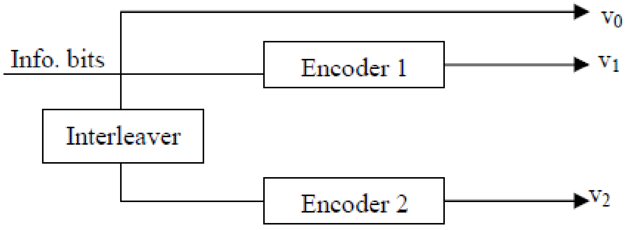


Figure.1. Block diagram of turbo code encoder.

II. ASYMMETRIC TURBO CODE

The turbo code with non-identical component encoders is known as Asymmetric turbo codes. The BER curve of a turbo code is divided into two region i.e. “waterfall” region and “error floor” region. “Waterfall” region is given as a steep slope for a long block of information bits and appears at a smaller SNR’s but “error floor” region appears at higher SNR’s and has a shallower slope due to code words of small weight. So, by using symmetric turbo codes we can't get the better BER performance simultaneously for both waterfalls as well as error floor region. In that respect asymmetric turbo code satisfy the requirements for both the regions provided the selection of component encoders are proper.

A. Asymmetric Turbo Encoders

The asymmetric turbo code, like symmetric code has two un-identical recursive systematic convolutional (RSC) codes which generate the systematic codeword that consist of parity bit and information bit. The block diagram of turbo encoders are shown in figure 1. Two components encoders are separated by an interleaver.

In fig.1, we can see that there are three outputs, systematic output (v0), and two recursive convolutional sequence output (v1 and v2). Two parallel concatenated RSC encoders are joint with an interleaver. The simple structure of turbo encoders with code rate 1/2, constraint length 3 with un-identical components shown in figure 1.

B. Asymmetric Turbo Decoders

In this case also we can use similar decoding algorithms which are applicable for symmetric turbo decoders like Maximum-a-posteriori (MAP), Logarithmic Maximum-a posteriori (Log-MAP), Maximum Logarithmic Maximum-a posteriori (Max Log-MAP) and Soft output Viterbi decoding (SOVA). However we use un-identical component code in the Corresponding turbo decoders.

The MAP algorithm is the optimum decoding technique but the complexity is high. It is used to determine the most probable information bit that was transmitted but the SOVA is used for most probable information sequence that was transmitted. In Max Log-MAP the values and operation are easier to implement due to logarithmic domain but Log-MAP avoid the approximation as in Max Log-MAP. Hence, we used Log-MAP decoding algorithm for performance evaluation with low computational cost without much compromise in the BER performance. The block diagram of Log MAP turbo decoder is shown in figure 2.

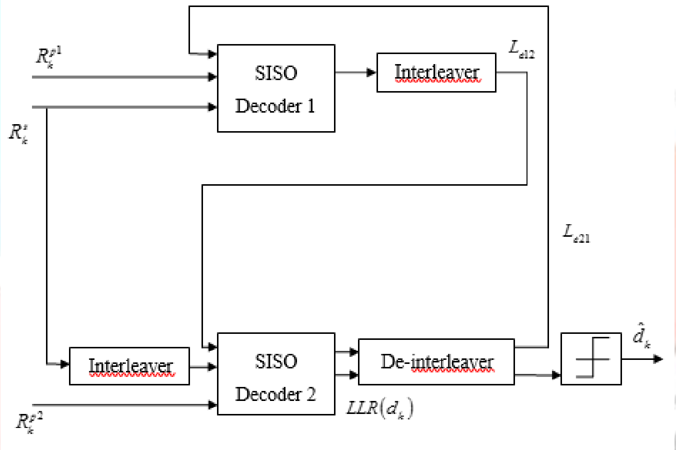


Figure.2.Block Diagram of Turbo decoder

The improvement in the error floor region can be done by serially concatenated turbo code or a parallel concatenated code of primitive components which have worse performance than original Berrou code in waterfall region. In asymmetric turbo code we consider the performance characteristic in both region i.e. in "waterfall" region as well as "error floor" region. In this paper we reduce the flattening of the "error floor" curve by applying asymmetric turbo code. The asymmetric turbo code with encoder with (7,5) and (15,17) component codes taking half code rate is shown in figure 3. Here, we can see that the two component codes are not identical, so it can be treated as asymmetric turbo code. The generated polynomial of the components codes are constructed with mixed type of the primitive polynomial and prime polynomial.

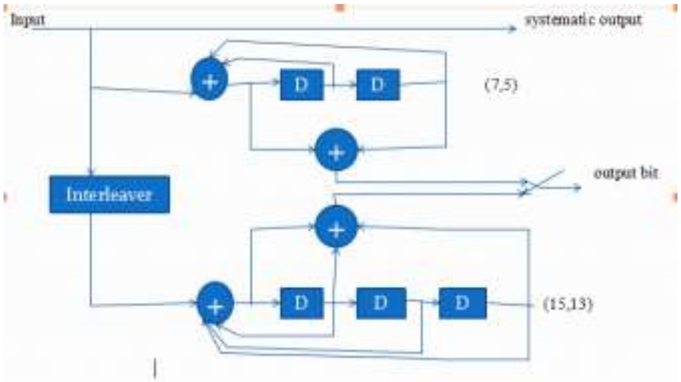


Figure.3.Block Diagram of Asymmetric Turbo Encoder

Table I

STANDARD PARAMETERS USED IN ASYMMETRIC TURBO CODE

channel	Additive White Gaussian Noise (AWGN)
Modulation	Binary Phase Shift Keying (BPSK)
Component Encoder	Two Non-identical Recursive convolutional codes (RSC_s)
RSC Parameters	N=2,k=1,K=3, G_1=7,5 (or 37,21 or 17,15); G_2=5,7 (or 21,37 or 15,17) etc
Puncturing used	YES (depends on coderate)
Components Decoder	Log-MAP decoder
Iteration	8

III. THE EFFECT OF VARIOUS CODEC PARAMETERS IN ASYMMETRIC TURBO CODE

There are many parameters, which affect the performance of asymmetric turbo codes. The various simulation results for asymmetric turbo code by using Binary Phase Shift Keying (BPSK) over Additive White Gaussian Noise (AWGN) channels are presented in this section. The parameters which affect the performance are as follows:-

a. The number of decoding iteration
b. Puncturing (or code rate)
c. Frame-Length
d. Component codes
e. Constraint length

The parameters which we have used in our simulation are shown in Table I. Before going through the various results for different parameters, we have tested and verified the simulation model of asymmetric turbo code by substituting generator polynomials as g1=g2= (15, 13) than comparing the result with symmetric turbo code for g0= (15, 13)

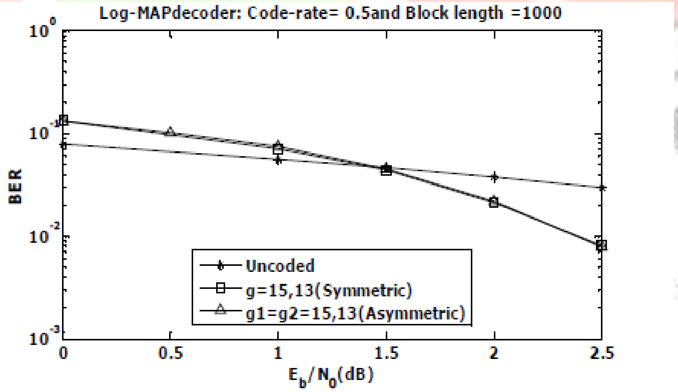


Figure 4: comparison between Asymmetric & Symmetric after identical components

The curves shown in figure 4 verify the asymmetric turbo codes simulation model. It is found that the result is matching with symmetric turbo codes BER performance, as expected. Mainly, the generated polynomials are optimum in terms of maximizing the minimum free distance of the components codes [9]. Most of the results in this paper for half code rate and also the decoding technique used are Log MAP decoder. All simulation results are taken over an AWGN channel with BPSK modulation.

A. The Effect of Number of iterations used

The performance of an asymmetric turbo code using Log-MAP algorithm with different decoding iteration is shown in figure 5. The generated polynomial used for the encoders are taken as (7, 5) and (5, 7). It can be seen from the above figure that the performance is nearly same as the encoded bits at low Eb/N0 but at high Eb/N0 the BER performance is improved after one iteration. When we increase the iteration like 2, 4, 6 and 8 then we get the better performance progressively. But after 6 iteration, there are a little improvement in performance approximately less than 0.1dB, so we use only 8 iteration due to complexity reason because as we increase the iteration more accurate the result, so more complexity.

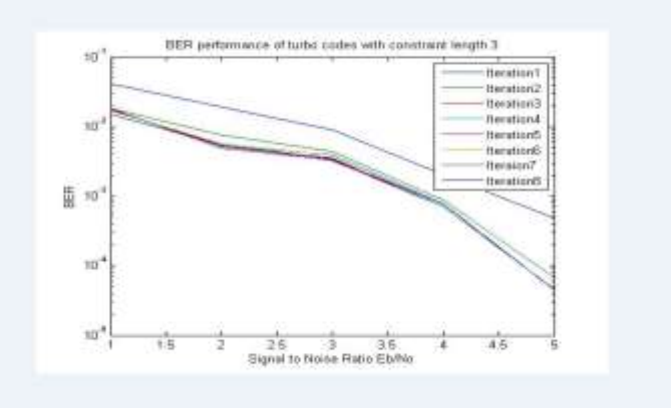


Figure 5: Performance using different number of iteration

B. The Effect of Puncturing or different code rate

Half of the parity bits from each component encoders are punctured when we use the half-rate code. But it is possible to avoid the puncturing and transmit all the parity bits through both the components encoder with one third code rate. Hence, the figure 6, shows the performance of BER taking parameters from table I, but the code rate is different i.e. half and one third. Like symmetric turbo code, the effect of puncturing in asymmetric turbo is also similar and effect reflected in the figure 6 which shows the performance graph for rate one-third is better than the rate one-half.

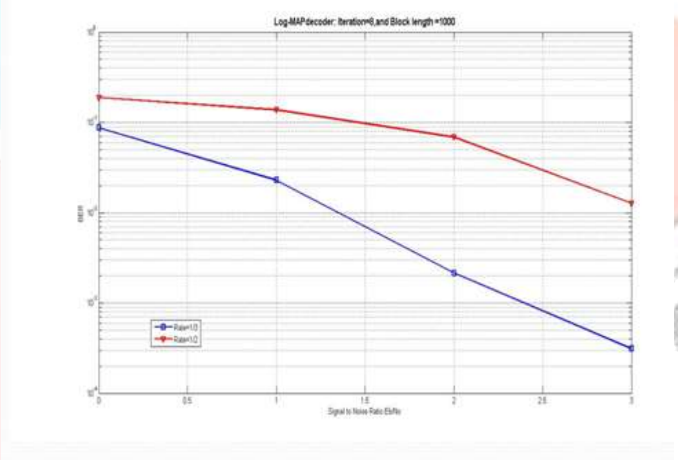


Figure 6: performance using half and one third code rate

C. The Effect of Frame Length

The BER performance is better as we increase the frame length. Since, the analysis of associated theoretical performance limits as a function of the coded frame length is already given by Dolinar *et a*.So, a large number of frame length is an unacceptable in real time performance because of the delay in transmission. In speech transmission we use 169 bit code while in video transmission we use 1000 bit code .So as we increase the frame length we don't get the real time transmission however it would be useful in data or non-real time transmission.

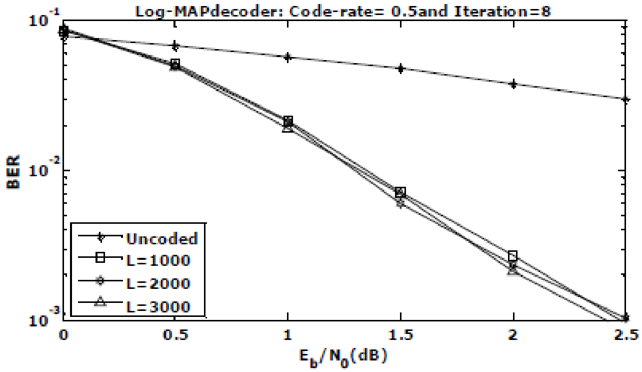
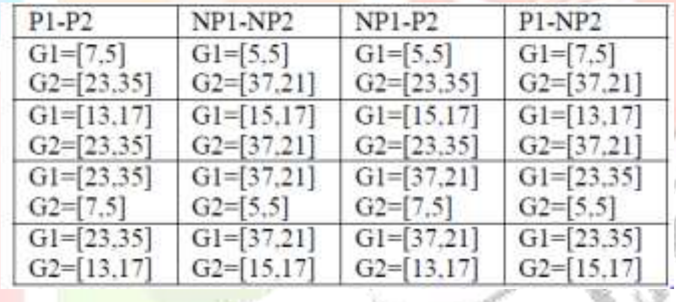


Figure 7: Performance using different frame length

D. The Effect of Components codes

The generator polynomial is also the important parameter used in the component codes. Figure 8, shows the different generated polynomials which affect the performance by using Log-MAP decoding technique. The generated polynomial used in this paper is for maximizing the minimum free distance of the component codes. The performances corresponding to the generator polynomials
(7, 5; 5, 7) and (15, 17; 17, 15) are shown and compared in this graph. The orders of generated polynomial are important and mostly the octal value can be used for feedback to the encoder. There is an important role of generated polynomials in affecting the performance of turbo codes. The generator polynomials which consist of the primitive polynomials (P-G) and non-primitive polynomials (NP-G) as the feedback generator polynomials (known as mixed type of generator polynomials), gives the best performance in water-fall and error-floor region at low SNR as well as high SNR. Table 2 consists of different types of generated polynomials.

Table 2
DIFFERENT TYPES OF ASYMMETRIC TURBO CODES WITH
DIFFERENT GENERATOR POLYNOMIALS



P1-P2	NP1-NP2	NP1-P2	P1-NP2
G1=[7,5] G2=[23,35]	G1=[5,5] G2=[37,21]	G1=[5,5] G2=[23,35]	G1=[7,5] G2=[37,21]
G1=[13,17] G2=[23,35]	G1=[15,17] G2=[37,21]	G1=[15,17] G2=[23,35]	G1=[13,17] G2=[37,21]
G1=[23,35] G2=[7,5]	G1=[37,21] G2=[5,5]	G1=[37,21] G2=[7,5]	G1=[23,35] G2=[5,5]
G1=[23,35] G2=[13,17]	G1=[37,21] G2=[15,17]	G1=[37,21] G2=[13,17]	G1=[23,35] G2=[15,17]

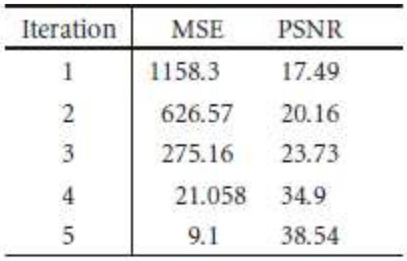
IV. IMAGE TRANSMISSION USING TYPICAL AND PROPOSED ASYMMETRIC TURBO CODES

In this section, an image transmission system over AWGN and Rayleigh fading channels using typical and proposed asymmetric turbo codes as error control coding is provided. The baseline JPEG algorithm is used to compress a QCIF (176 × 144) “Suzie” image.

A. The baseline JPEG image coding

The implementation of JPEG algorithm in this work is based on the baseline sequential DCT based, which is lossy. At the input to the encoder, the source image samples will be grouped into 8 × 8 blocks. Then the elements will go through level shift, FDCT, quantization, zigzag, run length and DC encoding, and then the entropy encoding. Finally, a bit stream of compressed image data will be obtained at the end of the encoder. Decompression is the exact reverse process. To deal with synchronization problems due to channel errors for bit streams containing variable length codes, restart intervals are implemented during the encoding process by keeping track the size of each interval. The decoding process will be performed on each interval individually, instead of the whole stream of image data bits. Using this method, any error will be contained in the particular interval only, without propagating the error to subsequent data. After decoding an interval, the process will resynchronize and restart to decode the next interval.

Table 3
Reconstructed image quality using typical turbo code over
AWGN channel.



Iteration	MSE	PSNR
1	1158.3	17.49
2	626.57	20.16
3	275.16	23.73
4	21.058	34.9
5	9.1	38.54

B. Simulation results of image transmission system

Simulations are done to compress a QCIF (176 × 144) grey level “Cameraman” image for the quality factor of 68. The JPEG compressed data is then encoded using typical and proposed asymmetric turbo codes. BPSK modulation is used. The image transmission system is shown in Figure 9. After every iteration, the output of turbo decoder is given to the JPEG decoder to reconstruct the image and the decoded image is compared with the original to compute mean square error (MSE) and peak signal-to-noise ratio (PSNR) according to the following formula:

$$\mathrm{MSE} = \left(\sum_{i=1}^{M} \sum_{j=1}^{N} \left(f(x,y) - f'(x,y) \right)^2 \right) \times (M \times N)^{-1}.$$

$$\mathrm{PSNR} = 20 \, \mathrm{Log}_{10} \left(\frac{255}{\mathrm{RMSE}} \right).$$

The original and the decoded “Cameraman” images at the output of typical turbo code system over AWGN channel for iteration 1 to iteration 5. The *Eb/N*0 is set as 2 dB. As shown in Table 3, the MSE Therefore, a zero MSE value is achieved for identical images. Higher values denote higher deviation between the original and degraded images. Note that a low MSE does not necessarily indicate high subjective quality. PSNR is derived using the root mean square error (RMSE) to denote deviation of a compressed image from the original in dB. For an eight-bit image, with intensity values between 0 and 255, the PSNR is given by decreases and PSNR increases as we increase the iteration. It is also noticed that even after 5th iteration, MSE of 9.1 is left uncorrected, which conforms that baseline JPEG is lossy. The original and the decoded “Camera man” images at the output of proposed asymmetric turbo code system over AWGN channel are shown in Figure 11. It is observed that it requires only four iterations to correct the errors where as typical turbo code requires five iterations. The quality of the reconstructed images for every iteration. The decoded image quality (in PSNR) of typical turbo code and the proposed turbo code systems over AWGN channel are also provided in Figures respectively. We observe that higher performance gains are achieved using proposed asymmetric turbo code for all iterations and there is no increase in gain after the fourth iteration. The original and the decoded “Cameraman” images at the output of proposed asymmetric turbo code system without interleaver over AWGN channel.

V. RESULTS

In this paper, we presented the results of a study on the performance of an image transmission system using typical and proposed asymmetric turbo codes. Although the search procedure of perfect parameters for good component encoder at low and high SNR is quiet exhaustive, the modifications in turbo encoder really contribute performance improvements in turbo code system. The simulation results in-dicate that the performance of image transmission system using proposed asymmetric turbo code is superior to that using typical turbo code for different channel conditions.

Figure 8: Original and decoded “Cameraman” images over AWGN channel using proposed asymmetric turbo code without interleaver with an *Eb/N*0 of 2 dB.

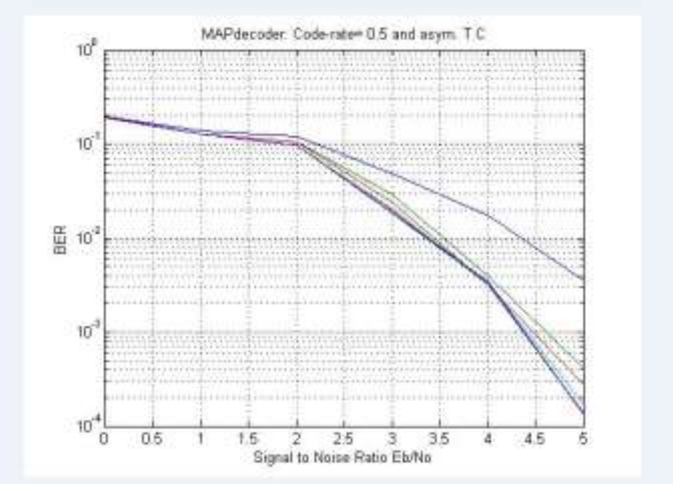


Figure 9: Performance Of Asymmetric Turbo codes with g1= (7, 5) and g2= (15, 13).

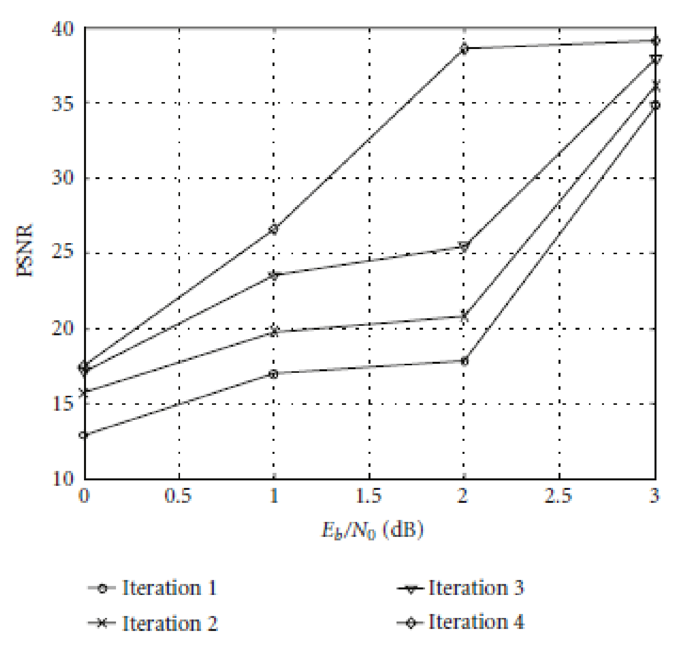


Figure 10: Decoded image quality (in PSNR) of proposed asymmetric turbo code with inetrleaver over AWGN channel.

REFERENCES

[1] C. Berrou, A. Glavieux, andO. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: turbo-codes. 1,” in *Proceedings of International Conference on Communication (ICC ’93)*, vol. 2, pp. 1064–1070, Geneva, Switzerland, May 1993.

[2] M. C. Valenti and J. Sun, “Turbo codes,” in *Handbook of RF and Wireless Technologies*, chapter 12, pp. 375–400, Newnes, Oxford, UK, 2004.

[3] L. C. Perez, J. Seghers, and D. J. Costello Jr., “A distance spectrum interpretation of turbo codes,” *IEEE Transactions on Information Theory*, vol. 42, no. 6, part 1, pp. 1698–1709, 1996.

[4] O. Y. Takeshita, O. M. Collins, P. C. Massey, and D. J. Costello Jr., “A note on asymmetric turbo-codes,” *IEEE Communications Letters*, vol. 3, no. 3, pp. 69–71, 1999.

[5] P. C. Massey, O. Y. Takeshita, and D. J. Costello Jr., “Contradicting a myth: good turbo codes with large memory order,” in *Proceedings of IEEE International Symposium on Information Theory*, p. 122, Sorrento, Italy, June 2000.

[6] K. Ramasamy, B. Balakrishnan, and M.U. Siddiqi, “A new class of asymmetric turbo code for 3G systems,” *AEU - International Journal of Electronics and Communications*, vol. 60, no. 6, pp. 447–458, 2006.

[7] J. D. Andersen, “Selection of component codes for turbo coding based on convergence properties,” *Annales des Telecommunications*, vol. 54, no. 3, pp. 208–213, 1999, special issue on iterated decoding.

[8] K. Ramasamy and M. U. Siddiqi, “Weight distribution analysis of proposed asymmetric turbo code for improved performance,” *AEU - International Journal of Electronics and Communications*, vol. 60, no. 7, pp. 479–493, 2006.

[9] J. D. Andersen and V. V. Zyablov, “Interleaver design for turbo coding,” in *Proceedings of International Symposium on Turbo Codes*, pp. 154–156, Brest, France, September 1997.

[10] D. Divsalar and F. Pollara, “On the design of turbo codes,” The Telecommunications and Data Acquisition Progress (TDA) Progress Report 42-123, pp. 99–121, Jet Propulsion Laboratory (JPL), Pasadena, Calif, USA, November 1995.

[11] S. Benedetto and G. Montorsi, “Unveiling turbo codes: some results on parallel concatenated coding schemes,” *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 409–428, 1996.

[12] D. Divsalar, S. Dolinar, F. Pollara, and R. J. McEliece, “Transfer function bounds on the performance of turbo codes,” The Telecommunications and Data Acquisition Progress (TDA) Progress Report 42-122, pp. 44–55, Jet Propulsion Laboratory (JPL), Pasadena, Calif, USA, August 1995.