Volume 2, Issue 1, Feb 2014 ISSN: 2320-5288

International Journal of Engineering Technology & Management Research
Journal homepage: www.ijetmr.org

A Survey Paper on Different Turbo Decoders and Their Comparison

Gaurav Suman
M.Tech Student,
Electronics and Communication department
Takshshila Institute of Engineering & Technology,
Jabalpur (M.P.) [INDIA]
gaurav.suman@gmail.com

Abstract— In order to have reliable communication, channel coding is often employed. Turbo code as a powerful coding technique has been widely studied and used in communication systems. Turbo coding is an advanced forward error correction algorithm. Ultimate Performance that approaches the Shannon limit requires a new approach using iteratively run soft in/soft out (SISO) decoders called turbo decoders. However, the implementation of various Turbo Decoders suffers from a large delay and high power consumption. For this reason, they are not suitable for many applications like mobile communication systems. In this paper, a comparative study has been made and various decoding algorithm used in SISO Turbo Decoders have been analyzed viz. MAP, Log-MAP, Max- Log-MAP and SOVA, to overcome this drawback. This paper examines the principles of turbo coding and decoding algorithms and compare their BER performance.

Keywords— Turbo Code, MAP, SOVA, FPGA, Convolutional code, BER Performance

1. INTRODUCTION

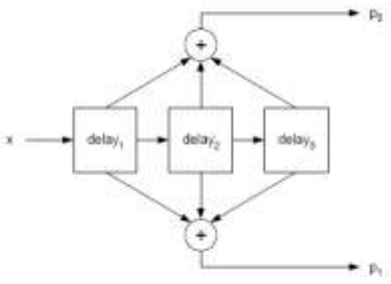
In 1949 Claude Shannon published a classic paper[1] that established a mathematical basis for the consideration of the noisy communications channel. In his analysis he quantified the maximum theoretical capacity for a communications channel, the Shannon limit, and indicated that error-correcting channel codes must exist that allowed this maximum capacity to be achieved. The intervening years have seen many well-considered channel codes inch towards the Shannon limit, but all contenders have required large block lengths to perform close to the limit. The consequent complexity, cost, and signal latency of these codes have made them impractical within 3 to 5 dB of the limit, but they provide useful coding gain at higher values of Eb/No and bit error rate. In 1993 Berrou, Glavieux and Thitimajshima[2] proposed "a new class of convolution codes called turbo codes whose performance in terms of Bit Error Rate (BER) are close to the Shannon limit". In seven pages the authors described an approach to coding that, in their supporting analysis, indicated that it was possible to operate within 0.7dB of the Shannon limit. The potential performance offered by turbo codes has excited both academic and industrial researchers. The last 7 years has seen a consequent explosion of research into all aspects of turbo codes.

This paper examines the principles of turbo coding and decoding, focussing on the coding and decoding algorithms. It then examines the performance of turbo codes, both in multilevel and simple constellations. It concludes by examining the state of turbo code research and development.

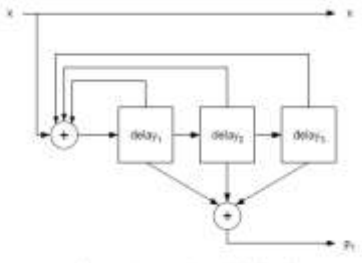
2. PRINCIPLE OF TURBO CODE

It is theoretically possible to approach the Shannon limit by using a block code with large block length or a convolutional code with a large constraint length. The processing power required to decode such long codes makes this approach impractical. Turbo codes overcome this limitation by using recursive coders and iterative soft decoders. The recursive coder makes convolutional codes with short constraint length appear to be block codes with a large block length, and the iterative soft decoder progressively improves the estimate of the received message.

Coding A specific type of convolutional coder is used to generate turbo codes. The convolutional coder shown in Figure 1a has a single input, x, outputs p 0 and p 1, and a constraint length K=3. Multiplexing the outputs generates a code of rate R=1/2.



a) convolutional coder



b) recursive systematic convolutional coder

Figure 1. Convolutional Coders

The convolutional coder shown in Figure 1b differs in that one of the outputs, p 0, has been “folded back” and is presenting one of its output sequences at the coder input, making it recursive. This has the effect of increasing the apparent block length without affecting the constraint length of the coder. The input is also presented as one outputs of the coder, making it systematic. Such coders are thus called recursive systematic convolutional (RSC) coders.

In non-recursive convolutional codes it is common practice to flush the coder with zeros to bring the decoder to an end state. Flushing with zeros does not readily work with recursive coders, however relatively simple binary arithmetic can establish the input sequence that will generate a zero state. RSC codes can thus be made to appear like linear block codes.

A turbo code is the parallel concatenation of a number of RSC codes. Usually the number of codes is kept low, typically two, as the added performance of more codes is not justified by the added complexity and increased overhead. The input to the second decoder is an interleaved version of the systematic x, thus the outputs of coder 1 and coder 2 are time displaced codes generated from the same input sequence. The input sequence is only presented once at the output. The outputs of the two coders may be multiplexed into the stream giving a rate R=1/3 code, or they may be punctured to give a rate R=1/2 code. This is illustrated in Figure 2.

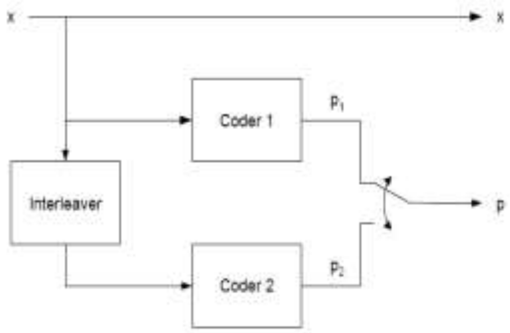


Figure 2. Punctured Rate R=1/2 Turbo Coder

The interleaver design has a significant effect on code performance. A low weight code can produce poor error performance, so it is important that one or both of the coders produce codes with good weight. If an input sequence x produces a low weight output from coder 1, then the interleaved version of x needs to produce a code of good weight from coder 2. Block interleavers give adequate performance, but pseudo random interleavers have been shown to give superior performance[3].

3. DECODING ALGORITHMS

At the receiver, the signal is demodulated with its associated noise and a soft output provided to the decoder. The soft output might take the form of a quantized value of the decoded bit with its associated noise, or it may be a bit with associated probability (i.e. 1 with P(1)=0.65). Most often it is the log likelihood ratio (LLR), which is defined as:

The LLR is a measure of the probability that, given a received soft input y, the message bit u(t) associated with a transition in the trellis is 1 or 0. If the events are equiprobable then the output is 0, but any tendency for u(t) towards 1 or 0 will result in positive or negative values of.

$\lambda(t)$

This section highlights two classes of trellis-based

algorithms which are typically used to decode turbo codes. Figure 3 lists the two classes of trellis-based algorithms and their derivatives in order to characterize their relationships to one another.

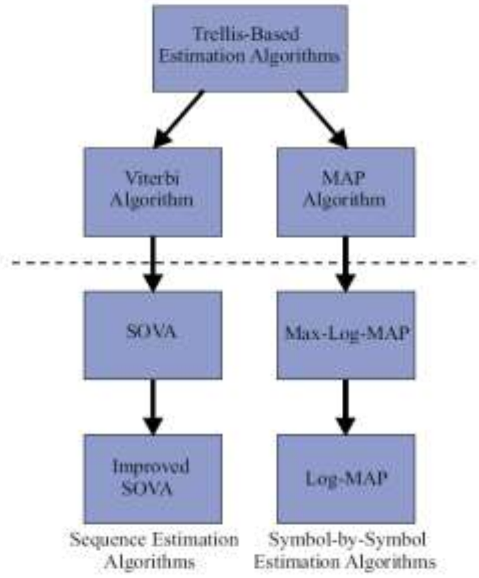


Figure 3. Trellis based turbo algorithm

The Viterbi algorithm (VA) accepts soft inputs but produces hard outputs. Although the maximum a posteriori (MAP) algorithm accepts soft inputs and produces soft outputs, it suffers from numerical instability. Their derivatives, which are shown below the dotted line, accept soft inputs and produce soft outputs; these algorithms, therefore, are suitable for use in turbo decoding applications. SISO algorithms are necessary for turbo decoding because the decoders are required to share their extrinsic information with each other. Although SISO decoding algorithms are more computationally complex, they allow iterative sharing of results between decoders, which permits the use of powerful concatenated coding structures.

The two main types of decoder are Maximum A Posteriori (MAP) and the Soft Output Viterbi Algorithm[4] (SOVA). MAP looks for the most likely symbol received, SOVA looks for the most likely sequence. Both MAP and SOVA perform similarly at high Eb/No. At low Eb/No MAP has a distinct advantage, gained at the cost of added complexity. MAP was first proposed by Bahl[5] and was selected by Berrou[2] as the optimal decoder for turbo codes. MAP looks for the most probable value for each received bit by calculating the conditional probability of the transition from the previous bit, given the probability of the received bit. The focus on transitions, or state changes within the trellis, makes LLR a very suitable probability measure for use in MAP.

SOVA is very similar to the standard Viterbi algorithm used in hard demodulators. It uses a trellis to establish a surviving path but, unlike its hard counterpart, compares this with the sequences that were used to establish the non-surviving paths. Where surviving and non-surviving paths overlap the likelihood of that section being on the correct path is reinforced. Where there are differences, the likelihood of that section of the path is reduced. At the output of each decoding stage the values of the bit sequence are scaled by a channel reliability factor, calculated from the likely output sequence, to reduce the probability of over-optimistic soft outputs. The sequence and its associated confidence factors are then presented to the interleaver for further iterations. After the prescribed number of iterations, the SOVA decoder will output the sequence with the maximum likelihood. Iterative architecture of Turbo Decoder is shown in figure 4.

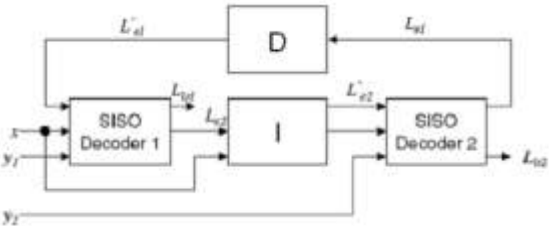


figure 4. Iterative architecture of Turbo Decoder.

4. COMPARATIVE STUDY OF MAP AND SOVA DECODER

- All MAP decoders are based on BCJR (Bahl-cocke-Jelinek-Raviv) algorithm and SOVA is based on Viterbi algorithm.
- MAP algorithm tries to minimize the code word error by maximizing the probability, which is a maximum likelihood (ML) algorithm while SOVA attempts to maximize the a-posteriori probabilities (APP) of the individual bit.
- MAP algorithm involves extensive multiplication and logarithmic computation, which are complicated in hardware implementation, but SOVA has relatively low computational complexity.
- MAP algorithm takes all paths into consideration and generates the sum of probabilities of all paths in the estimation. Its performance is optimal in terms of bit error rate (BER). On other hand SOVA produces the soft output by considering only two ML paths. Hence the extrinsic information depends strongly on the choice of the two paths. It yields in inferior soft output than MAP algorithm. been optimized for power consumption. They derived Adaptive SOVA and through experimentation they shown that up to a 52% power savings can be achieved.

5. LITRATURE SURVEY

Most of the literature is focused on the optimization of power and speed. Atluri & Arslan have proposed VLSI architecture for low power implementation of a Log-MAP decoder through the minimization of memory size for the storage of reverse state metrics. The technique resulted in 88% reduction in memory utilization at the expense of 13% increase in branch metric storage. This provides a net power reduction of 35%. Gord Allan has worked on VLSI implementation of a modified Log-MAP decoder that can process data at up to 60 Mbps, consuming 15 mW of power and 24000 gates. Bougard, Giulietti, Perre & Catthoor have developed and prototyped a novel decoder architecture whose throughput is up to 75.6 Mbps with an energy efficiency of 1.45 nJ/bit/ iterations. Weifeng & Ting have proposed a new structure of Log-MAP Turbo decoder. Their results of tests on Altera FPGA have proved that proposed structure could achieve good performance (BER-10^{-5} at SNR-1.5dB) with high throughput (440 Kbps), which meet the 384 Kbps requirements in 3G standard. Ituero, Vallejo & Mujtaba have proposed Application Specific Instruction Set Processor (ASIP) architecture for Max-Log -MAP algorithm. They achieved a high concurrency level, which reduces the total execution time. Vogt & Finger have given a method for improving the Max-Log-MAP algorithm for Turbo decoders. They applied a simple scaling factor to the extrinsic information, which improved the decoding performance by about 0.2 to 0.4 dB. Engling Yeo et all have implemented two 8-state, 7-bit Soft Output Viterbi decoders in 4 mm2 chip operates at 1.8V, 400 mW and verified to decode at 500Mb/ s. Jian Liang et all have presented a dynamically reconfigurable FPGA-based Turbo decoder which has been optimized for power consumption. They derived Adaptive SOVA and through experimentation they shown that up to a 52% power savings can be achieved.

6. SIMULATION

MATLAB routines for simulating turbo codes are available on the Internet[7], with the proviso that they are only used for educational purposes. In exploring the performance of turbo codes several simulations were run on a PC. For the purposes of this simulation a punctured turbo code at rate R=1/2 was used. The data block length was 400 bits, and a MAP decoder was used in the simulation. The results shown at Figure 5 are the BER vs Eb/No curves for different numbers of iterations from n=1, 2, 5 and 10. A BPSK channel was assumed.

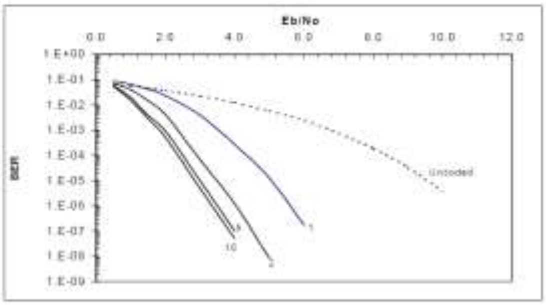


Figure 5. BER for Turbo Code

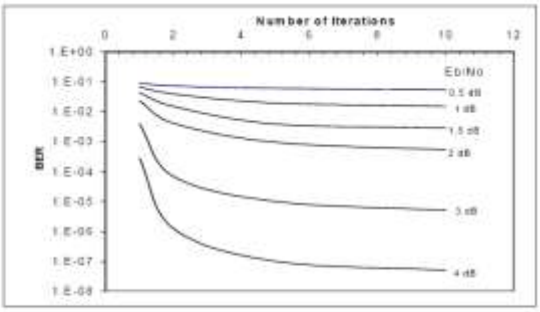


Figure 6. BER vs Eb/No as number of iterations varies

It can be seen that BERs of the order of $10^{-5}$are achievable with Eb/No >3 dB with modest numbers of iterations. A typical coding gain of Eb/No of 7dB, relative to an uncoded channel, was observed at a BER of 10^{-5}. Figure 4 infers that the BER should improve with each iteration, so a series of simulations were run to evaluate the improvement. It can be seen from Figure 5 that the first few iterations yield the most significant improvements in BER for any given Eb/No. Thereafter the results appear to converge onto a BER for each value of Eb/No. It is apparent that there is a trade off to be made between the number of iterations, processing power, and Eb/No when seeking a given BER.

A final simulation was run to compare the performance of the MAP and SOVA decoders, particularly at low values of Eb/ No. The number of iterations was set at 5, since Figure 5 and Figure 6 indicated that further iterations would yield marginal improvements. The results, shown at Figure 7, confirm that MAP is about 0.5

dB better than SOVA at low values of Eb/No.

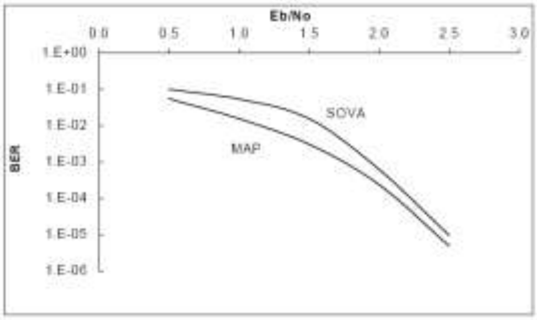


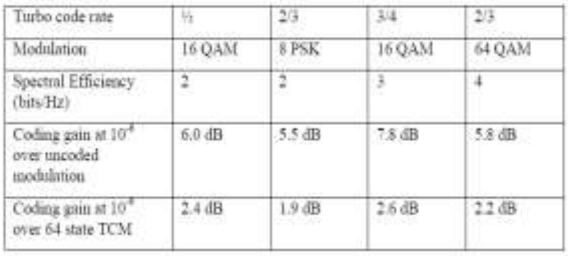
Figure 7. Comparison of MAP and SOVA at Low Eb/No

7. PUBLISHED RESULTS

Efficient and effective coding is not an end unto itself; the performance of the modulation scheme has equal importance to the communications channel. The ideal combination would be a spectrally efficient modulation scheme combined with a robust coding scheme.

Shortly after their original paper on turbo codes, Berrou and Glavieux joined with Le Goff to examine the integration of turbo codes with high spectrally efficient modulation schemes[8]. In a highly readable paper they examined the association between turbo codes and multi-level modulation schemes, QAM and MPSK, over Gaussian and Rayliegh channels. The coding gains obtained for an AWGN channel are at Table 1.

The graphs in the paper show consistently lower BER being predicted for turbo coded 16 QAM than for 64 TCM at BERs below 10^{-3}, for both Gaussian and Rayliegh channels. Chang and Wei[9] proposed methods for integrating turbo codes with q-ary modulation schemes, focussing on 4-ary (16 QAM) and 8-ary (64 QAM) modulation. Gray code mapping was used. The basic principle, which others seem to have followed, is to code, interleave, puncture and decode at symbol boundaries. For rate R = 1/2, 16 QAM, they reported a 1dB improvement over the corresponding binary (BPSK) turbo code at a BER of 10^{-5}. For a rate R=2/3, 64 QAM, code they reported a gain of about 0.6 dB at a BER of 10^{-5}, observing that the code was approaching the error floor.



Turbo code rate	½	2/3	3/4	2/3
Modulation	16 QAM	8 PSK	16 QAM	64 QAM
Spectral Efficiency (bits/Hz)	2	2	3	4
Coding gain at 10^{-5} over uncoded modulation	6.0 dB	5.5 dB	7.8 dB	5.8 dB
Coding gain at 10^{-5} over 64 state TCM	2.4 dB	1.9 dB	2.6 dB	2.2 dB

Table1. Coding Gains for AWGN Channels

Turbo Code Research & Development It is obvious that turbo codes have the potential to make a significant contribution to communications systems, particularly those that operate with a low Eb/No. The last 10 years have seen turbo codes move from theory, through simulation, to the emergence of the first products. The key enabling technology has been the emergence of electronic devices capable of implementing the required number of operations per second.

8. RESEARCH

A search of the IEEE and IEE websites reveals a rich vein of papers, reflecting the intensity of research into turbo codes. Initial research was focussed on establishing the fundamental properties of turbo codes and their performance envelope. Typical areas of research included:

Applying turbo codes to different modulation schemes[9], [10].
Establishing the factors that affect code performance [6], [11], [12].
Exploring the effects of interleaver design [13].
Types of decoders [14], [15], [16].
It is generally concluded that turbo codes perform very well when compared to other

convolutional and block codes, particularly when combined with multilevel constellations. The fundamental properties of turbo codes are now well understood. There is considerable potential for further research into the complex trade-offs between code mapping in constellations, coder and decoder design, and interleaver design. The latter seems to play an important role in turbo code performance, and has some affect on the observed error floor; this is the subject of ongoing research. Shoemake[17] have looked at 8-PSK modulation schemes, focussing on the constellation mapping. They observed the lack of an error floor on their simulation, and concluded that the constellation mapping and good interleaver design were critical to code performance.

More recently, the focus of published papers moved towards the application of turbo codes to real situations. Obvious applications include the protection of radio channels, where power limiting or fading can cause low signal to noise ratios. Typical areas of research are mobile satellite communications[18], HF data systems [19], jammed channels[20], and on channels subject to multipath fading[21]. Numerous recently published papers[22] have addressed a number of mainstream research areas of interleaver design, modifications to the Bahl decoder, and performance issues. Others have looked outside the mainstream of research at issues such as codes for low data rates, diversity radio systems, and codes for use in memory-limited decoders. The Proceedings for the MILCOM 1999 Conference[23] contain some interesting papers and concepts, including the implementation and performance issues for software radios, and code performance under jamming.

9. DEVELOPMENT

Manufacturers are looking seriously at the advantages of turbo codes over the well-established Viterbi and Reed-Solomon (RS) codes. The initial emphasis has been on producing chipsets to allow manufacturers to implement turbo codes in their hardware. High-speed electronics has aided the development of chips with sufficient processing power to implement turbo decoders, currently at data rates of a few tens of Mbit/s. AHA and Comatlas have produced turbo code chipsets at 36 and 40 Mbit/s respectively, while Small World Comms have an FPGA for turbo decoding up to 90 Mbit/s.

These enabling technologies have helped to bring the first generation of turbo code based equipment to the marketplace. Comtech have launched a satellite modem that uses a rate R = 3/4 turbo code claiming significant bandwidth and BER improvements over modems using Viterbi and RS codes. Other major satellite modem manufacturers such as Comsat Labs are introducing turbo codes into their product. Alantro are developing turbo code firmware for such diverse roles as satellite links and hard disk drives.

The mobile phone industry is looking at turbo codes to provide error correction for third generation handsets. There have been many recent papers on the subject of implementing turbo codes in CDMA systems for UMTS. Turbo codes can operate at reduced power levels offering improved safety and extended battery life, both of which are important to the mobile phone user.

Recommended Reading This short paper has given an introduction to the subject and its terminology. Material referred to and used in its preparation is listed at the end of this paper. The author would highly recommend “Digital Communications Fundamentals and Applications (second edition)” by Bernard Sklar as the next step in learning about turbo codes. This excellent textbook contains a chapter on turbo codes which takes the reader further than this paper; the remainder of the book sets coding in the wider context of the communications channel.

10. CONCLUSION

Turbo codes are a class of convolution code which exhibit the properties of large block codes through the use of recursive coders. Coder performance is heavily dependent on the design of the interleaver, which must ensure adequate weight for at least one of the codes. Soft decoders are used with turbo codes to allow the a posteriori probability to be passed between decoder iterations. The MAP decoder is generally preferred because it offers the same performance as SOVA at 0.5dB lower value of Eb/No. This performance edge is achieved at the cost of increased complexity.

A half rate turbo coded BPSK channel can offer coding gains of 7dB over an uncoded channel at a BER of 10-5. The coding gain depends on the number of iterations; typically 5 to 10 iterations generate most of the improvement. Turbo code performance has been simulated for a number of high order constellations, including 8PSK, 16 QAM and 64QAM, and the importance of code mapping within the constellation has been recognised. The reasons for the observed error floor are not yet fully understood, but indications are that it is linked in some way to interleaver design.

Research is now beginning to be directed towards applying turbo codes to resolve real communications problems. There are many potential applications for turbo codes, particularly in the field of radio communications. With suitable chipsets becoming available the first products are beginning to be marketed. The superior performance offered by turbo codes ensures that they have a good future in information systems. Several useful introductory texts have been identified in researching this paper and are recommended to newcomers to the subject [24], [25], [26].

11. FUTURE WORK

Although research work is going on in this area to optimize the Turbo Decoder implementation for low power consumption and high speed, but still more work is required. The design of low cost and low power Turbo Decoders is the need of an hour. So, I am planning to do work towards optimization of decoding algorithm for Turbo codes not only on the basis of power and speed but also for small area using VLSI technology.

REFERENCES:

[1] Shannon CE. “Communication in the presence of Noise”. Proceedings IRE, Jan 1949, reprinted Proceedings IEEE, Vol 86, No 2 Feb 98.

[2] Berrou C, Glavieux A & Thitimajshima P. “Near Shannon Limit Error-Correcting Coding and Decoding : Turbo Codes”. IEEE 1993.

[3] Andersen JD and Zyablov VV, “Interleaver Design for Turbo Coding”, Proceedings of the International Symposium on Turbo Codes, September 1997.

[4] Berrou C, Adde P, Angui E & Faudeil S. “A Low Complexity, Soft Output, Viterbi Decoder Architecture”. Proceedings IEEE. International Conference on Communications. May 1993.

[5] Bahl LR, Cocke J, jelinek F & Rajiv J.. “Optimal Decoding of Linear Codes for Minimising Symbol Error Rate”.

[6] Wang CC. “On the Performance of Turbo Codes”. IEEE, 1998.

[7] http://www.ee.vt.edu/yufei/turbo.html

[8] Le Goff S, Glavieux A, Berrou C. “Turbo Codes and High Spectral Efficiency Modulation”. IEEE 1994.

[9] Chang Y, Wei VK. “Q-ary Turbo Codes with QAM Modulations”.

IEEE 1996.
[10] Fazel K, Papke L. “Combined Multilevel Turbo Codes with 8-PSK modulation”. IEEE 1995.
[11] Divsalar D, Dolinar S, McEliece RJ & Pollara F. “Performance Analysis of Turbo Codes”. IEEE 1995.
[12] Dolinar S, Divsalar D and Pollara F. “Turbo Code Performance as a Function of Block Size”. IEEE 1998.
[13] Yuan J, Vucetic B & Feng W. “Combined turbo codes and interleaver design”. IEEE 1998.
[14] Jung P and Nasshan MM. “Comprehensive Comparison of Turbo Code Decoders”. IEEE 1995.
[15] Fossorier MPC, Burkert F, Lin s & Hagenauer J. “On the Equivalence Between SOVA and max- Log-MAP Decoders”. IEEE Communications letters 1998.
[16] Ambroze A, Wade G & Tomlinson M. “Iterative MAP Decoding for Serial Concatenated Comnvolutional Codes”. IEE Proceedings on Communications, April 1998.
[17] Shoemake MB, Heegard C & Rossin E. “Turbo Codes for High Order Constellations”. IEEE,1998.
[18] Fonseka JP. “Applications of Turbo Codes in Satellite Mobile Systems”. Electronics letters January 1999.
[19] Coulton P, Honary A, Darnell M & Wicker SB. “Application of Turbo Codes to HF Data Transmission”. IEE Conference on HF Radio Systems and Techniques”. July 1997.
[20] Jordan MA. “Turbo Code Performance in Partial Band Jamming”. IEEE 1998.
[21] Li KL and Cheung SW. “Performance of Turbo Codes with fading Compensation in Multipath Channels”. IEEE 1999.
[22] IEEE/IEE Electronic Library Online
[23] Various authors. “Military Communications Conference Proceedings, 1999. MILCOM 1999”. IEEE
[24] Valenti MC. “An Introduction to Turbo Codes”.
[25] Battail, G. “A Conceptual Framework for Understanding Turbo Codes”. IEEE Journal on Selected Areas in Communications, 1998.
[26] Sklar B. “Turbo Code Concepts