International Journal of Scientific Engineering and Technology Research

www.ijsetr.com

ISSN 2319-8885
Vol.04,Issue.10,
April-2015,
Pages:1845-1851

LLR Enhanced Model with Turbo Decoder for Index Assignment of Exit Optimization

CH. BALARAMA KRISHNA SAIBABA[1], CH. RAVI KUMAR[2]

[1]PG Scholar, Dept of ECE, Sir C.R.Reddy College of Engineering, Eluru, AP, India, E-mail: chidaraboinabalaram@gmail.com.

[2]Sr.Assistant Professor, Dept of ECE, Sir C.R.Reddy College of Engineering, Eluru, AP, India, E-mail: ravich230@gmail.com.

Abstract: This project considers a Turbo decoder for which the log likelihood ratios (LLRs) exchanged between constituent decoders are subject to independent bit errors. The resilience of the decoder against these errors can be improved –without increasing decoding complexity - by optimizing the index assignment(IA) of the LLRs with respect to the extrinsic information transfer (EXIT) characteristic of the decoder. To enable an effective optimization of the IA we propose an approximation of the mutual information of the extrinsic decoder output that can be computed efficiently without histogram measurements. In particular the turbo decoder is trying to find the maximum likelihood solution under the false assumption that the input to the encoders were chosen independently of one another, subject to a constraint on the probability that the messages so chosen happened to be the same. so to provide exact analytical function of constraint, and then turbo decoder is an iterative method originally suggested by gauss, which is trying to solve the optimization problem by solving the equations of lagrange with a lagrange multiplier of -1. The extrinsic information is the information about a given bit available by the use of knowledge about other bits in a transmitted data stream. Most often, the extrinsic information is modeled as a Gaussian random variable. However, a generic mathematical model capturing the operations of a TEQ seems to be missing in literature. It is with this motivation that we propose a suitable mathematical model for a TEQ under converging conditions. This is the log likelihood ratio (LLR) enhancement model that describes the evolution of the extrinsic information as observed at the equalizer output. The LLR becomes a monotonically increasing function of the signal to noise ratio (SNR) from the no apriori condition to the perfect apriori condition about the transmitted bits.

Keywords: Turbo Codes, Error Resilience, Index Assignment, EXIT Chart, Decoder, Unreliable LLRS, TEQ.

I. INTRODUCTION

A. Architectures of Digital Communication Systems

Advances in integrated circuit technology and digital communication systems over the years enabled personal data communications to become practical, economical and widespread. These systems rely on data compression and error control coding to achieve higher spectral efficiency, noise robustness and increased fidelity. Many different digital communication standards have been developed during the past three decades for wireless and wireline transmission of voice and data, such as, the GSM and IS-54 second generation (2G) digital wireless cellular standards using rate ½ convolutional codes along with advanced speech compression techniques. These 2G systems replaced the old first generation (1G) analog cellular phone technology based mostly on frequency modulation, such as the Advanced Mobile Phone System (AMPS), which used multiple repititions of data as well as BCH block codes to perform error correction. A continued need for advanced services led to the research, development and eventual deployment of the third generation (3G) wireless systems during the late 1990's which use a high class of error correction codes known as turbo codes. Similarly, there has been a steady improvement in phone-line modem technology that resulted in standards that allow fast (up to several Mbits per second) Internet access in residential homes and businesses.

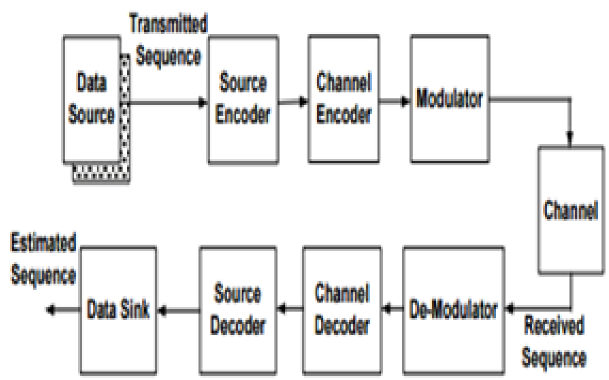

Fig.1. Architecture of a typical digital communication system.

Fig.1 shows a block diagram of a digital communication system architecture used for point-to-point communication in the abovementioned standards. The data source in Fig. 1 generates messages which are transmitted to the receiver over a physical medium. The source encoder removes unnecessary information from the source data (via compression), while exploiting the source redundancy and statistics. This typically involves pulse shaping and up conversion from baseband to passband frequency range. The receiver modules, shown in Fig. 1, invert the operations performed by the corresponding transmitter blocks, i.e., demodulator inverts the function of the modulator to get back the coded signal which is then passed through the channel decoder. This decoder removes redundunacy which is added by the channel encoder at the transmitting side. Finally, the source decoder decompresses the data before passing it on. For text files, 50 to 70% compression rates are possible, whereas for images, up to 90% compression can be achieved.

II. OVERVIEW OF TURBO CODES

Concatenated codes were first proposed as a method for achieving large coding gains by combining two or more simple component codes. Their earlier versions have been successfully used in deep space communication systems and data storage. A high class of error correcting codes known as Turbo codes were first introduced by Berrou, Glavieux and Thitimajshima in 1993 and used the concept of iterative decoding. They were the first practical coding technique that came close to approaching the Shannon capacity limit for a noisy Gaussian channel, e.g., within about 0.1 dB. They have been recently used to perform successful near-entropy data compression [1]. Turbo codes are built like a crossword puzzle, i.e., one encoder provides a set of clues about "rows" of the transmitted message data, then the interleaver changes the data order to a "columns" format and finally the second encoder generates the "column" clues. The iterative turbo decoding process proceeds in a manner that is similar to solving a crossword puzzle, i.e., one decoder is responsible for using the "row" clues and the other decoder is responsible for using the "column" clues. Both decoders keep exchanging their guesses about the decoded message symbols in order to successfully complete the "decoding" process.

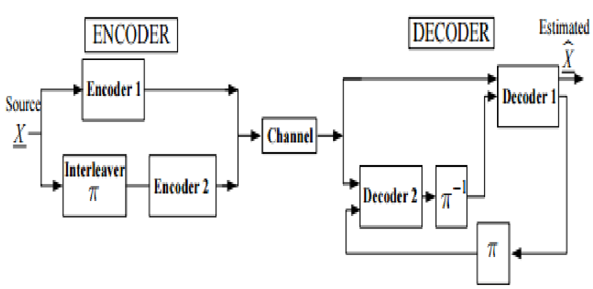

Fig.2. Parallel Concatenated Turbo Encoder/Decoder system.

This procedure continues for a prescribed number of iterations or until the decoders converge to the same solution. A typical turbo coding system is shown in Fig. 2 where π and $\pi-^{1}$ are the interleaving and de-interleaving functions respectively. To generate the hints, the two constituent encoders in a turbo code usually employ a trellis structure to encode the message data. An example of a trellis is shown in Fig. 3 which shows edges which start and end in a particular state. For example, if the input is 1 and the current state being 00, following the edge for a 1, the output is 11 and the destination state is 11 as can be seen from Fig. 3. Note that there is one trellis stage for one incoming bit in this case, thus n stages would be needed to encode/decode a sequence containing n bits.

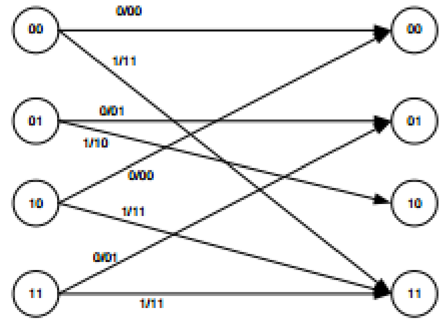

Fig. 3. One stage of a generic trellis structure.

A. Information Measures: Entropy and Mutual Information

Since EXIT charts modeling and construction uses several concepts from information theory, these concepts will be reviewed in this subsection. This exposition is an adaptation of the material from the classic textbook by Cover and Thomas and the course notes from a graduate course on information theory. For a discrete random variable X described by its probability mass function PX and taking values on a discrete alphabet X , the entropy of a discrete random variable X is defined as,

$$H(X) = E\left(\log_2 \frac{1}{P_X(x)}\right) = -\sum_{x\in\mathcal{X}} P_X(x)\log_2(P_X(x)) \quad \textbf{(1)}$$

The entropy H(X) represents the amount of uncertainty about the discrete random variable X, i.e., the average amount of information resolved by observation of a specific realization of X.

B. Typical Sequences and Source Coding for a Single User

The concept of typical sequences is of great use in the field of data compression as it provides a link between the the theoretical limit of compression and the entropy of a data source. This thesis will make use of this important connection when presenting theoretical and numerical results. Furthermore, this thesis will mostly be concerned with discrete memory less (data) sources using the following definition:

Two Multi-users Source Coding Problems: In most sensor networks, there are two (or more) correlated source nodes that wish to communicate with a common date fusion node. The goal of the multi-user or distributed source coding problem is to compress the sources, approaching the same optimal performance which is possible if the individual sources were able to communicate with each other. The following sub-sections will look at two such multi-user source coding problems.

1. The Slepian-Wolf Problem: Slepian and Wolf (SW) showed that in the case of distributed (separate) compression of two discrete memoryless sources X and Y followed by joint decoding, it is theoretically possible to achieve the performance of a system where the encoders can communicate with each other 1. Wyner considered a specialization of the SW problem i.e. the case in which the decoder is primarily interested in decoding only one source X and has the other source Y available at the decoder as the side information. The Slepian-Wolf problem deals with separate encoding of correlated sources, when joint recovery is performed at the receiver, as illustrated in Fig. 2.1 (a). Let X and Y be two discrete memoryless sources defined on alphabets X and Y respectively.

2. Rate Distortion Theory and the Wyner-Ziv Problem: An extension of the lossless data compression involves data representation subject to a distortion or fidelity criterion. This could include the storage or transmission of a continuous amplitude signal or a real number requiring an infinite number of bits. Under a fidelity criterion, when a certain amount of distortion is acceptable, the representation of continuous signals (or data) is possible. This kind of communication is more commonly known as source coding with a fidelity criterion, rate distortion coding or lossy source coding and comes under the framework of Rate-Distortion theory. Rate distortion theory gives the theoretical bound for how much compression can be achieved using lossy compression methods. In other words, it addresses the problem of determining the minimal rate R (in bits) that is needed to communicate the data over a channel or store the data without exceeding a given distortion level.

III. TURBO ENCODING AND DECODING

In this section, the structure of turbo codes along with the process of turbo encoding and decoding is reviewed [7]. Turbo codes, proposed in [10], have been shown to operate within a fraction of one dB of the capacity limit on AWGN channels. They have become part of the 3G wireless and DVB-RCS standards. A typical parallel concatenated turbo system is shown in Fig. 4.

A. Parallel and Serial Turbo Encoding

A turbo encoder consists of parallel or serial concatenation of simpler component codes which are usually recursive convolutional codes. A parallel

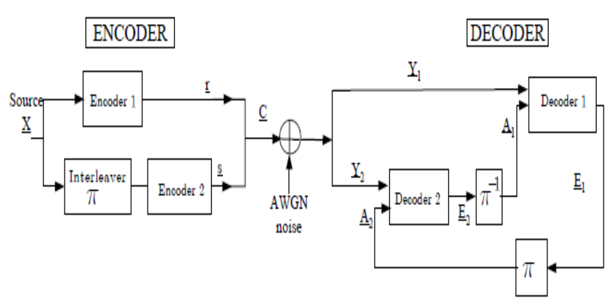

Fig.4. Parallel Concatenated Turbo Encoder/Decoder system.

concatenated code is shown in Fig.4. The message sequence X ∈ S (S is the set of message sequences of length S N bits) is encoded by the first encoder, while the interleaver π reorders the message symbols before they enter the second encoder. Although these codes were serial concatenated, they did not use the iterative decoding concept which a turbo coded system uses. Also, these codes did not have an inter leaver between the two serial constituent codes. A serial concatenated (turbo) code is shown in Fig. 5. The message source is encoded by Encoder 1 (also known as the outer decoder) producing a coded bit sequence C which is permuted by an inter leaver and consequently encoded by Encoder2, which is typically referred to as the inner encoder.

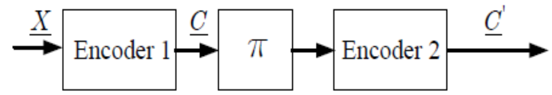

Fig. 5. Schematic block diagram of a serial concatenated (turbo) code.

B. Iterative Turbo Decoding

The most important part of the Turbo coded system is the iterative decoder which allows practical complexity decoding. For the turbo encoder structure shown in Fig. 4, optimal decoding which minimizes the bit error probability is too complex. This is due to the fact that the number of states of the joint trellis of both the encoders increases exponentially with the message block length. Thus, keeping in mind the complexity constraints, a suboptimal structure was proposed in [10], consisting of two separate decoders connected in an iterative loop. One constituent decoder operates on the first encoder's output (and the systematic stream). The other constituent decoder operates on the second encoder's output. In this process, the two decoders exchange information on the message bits in case of a parallel scheme. In a serial scheme, the decoders exchange information on the first encoder's coded bits. The methodology and notation used in this sub-section is based on [7], [8]. An iterative decoder for a parallel concatenated code. The two BCJR decoders are represented as 1 δ and 2 δ and are connected in an iterative loop. The output of decoder 1 is E_1 which stands for the extrinsic information, whereas the input of decoder 1 is A1 which signifies that the input is the a-priori information.

The same applies for decoder 2. Thus I j V are the labels of the input and output of the decoders where V can be either E or A depending on the input or the output. Also, i is the time index (or the iteration count) and j is either 1 or 2 depending on the decoder. An important point worth mentioning here is that the extrinsic of one decoder is the a-priori of the other decoder subject to interleaving/de-interleaving. More specifically, ( ) 1 2 A_i = π E_i and 1 ( )2 $1A_i$ = π− E_i as shown in Fig. 2.7 where π(i) is the interleaving function and π−1 (i) is the de-interleaving function. The turbo iterative decoding, in the i^th iteration can be described as

$$E_1^i = \delta_1\left(Y_1, A_1^i\right) - A_1^i \qquad \textbf{(2)}$$

$$E_2^i = \delta_2\left(Y_2, A_2^{i-1}\right) - A_2^{i-1}, \qquad \textbf{(3)}$$

where 1 Y and 2 Y are log-likelihood ratio (LLR) vectors on the coded bits from the two encoders (1 Y also includes LLRs on the systematic bits) and 0 0 [ ] 1 2 E = A = 0, 0,…, 0 . In Fig. 5, the appropriate interleavering π(i) and de-interleaving $\pi{-}1$ (i) is performed. The outputs of the decoding functions are logarithmic ratios on the a-posteriori probabilities,

$$\delta_2\left(Y_2, A_2^{i-1}\right) = \Pr\left(X|Y_2, A_2 = A_2^{i-1}\right) \quad (4)$$

$$\delta_1\left(Y_1, A_1^{i}\right) = \Pr\left(X|Y_1, A_1 = A_1^{i}\right), \quad (5)$$

Where vectors 1 E and 2 E are known as extrinsic information. These are then fed as apriori information to the next decoder. The subtraction makes sure that the next decoder only receives new information on the message bits. Finally, hard decisions on the message bits are performed by thresholding the output of decoder 1 by using a thresholding function h (a) = 1 if a > 0 and h (a) = 0 if a < 0 , i.e.,

$$\widehat{X} = h\left(\Pr\left(X|Y_1, A_1^{i}\right)\right). \quad (6)$$

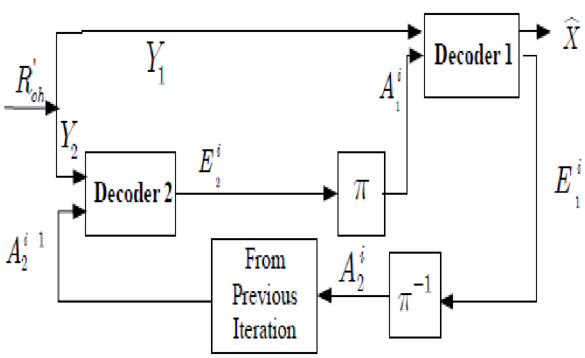

Fig.6. Schematic block diagram of the parallel concatenated turbo decoder.

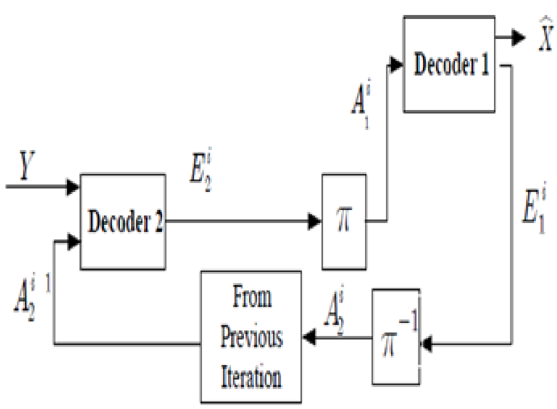

Fig.7. Schematic block diagram for a serially concatenated turbo decoder.

The previous discussion focuses on parallel concatenated codes. The treatment for serial concatenated codes is a little different. Fig. 6 shows the schematic diagram of a decoder for a serially concatenated turbo code. The turbo decoder, in the i-th iteration for a serial concatenated system can be described as

$$E_1^{i} = \delta_1\left(A_1^{i}\right) - A_1^{i} \quad (7)$$

$$E_2^{i} = \delta_2\left(Y, A_2^{i-1}\right) - A_2^{i-1}, \quad (8)$$

where Y is the log-likelihood ratio vector on the coded bits from the outer encoder produced by the demodulator and 0 0 [ ] 1 2 E = A = 0, 0,…, 0 . The decoding functions 1 δ and 2 δ produce logarithmic ratios on the outer encoder's coded bits and message bits, respectively:

$$\delta_2\left(Y, A_2^{i-1}\right) = \Pr\left(C|Y, A_2 = A_2^{i-1}\right) \quad (9)$$

$$\delta_1\left(A_1^{i}\right) = \Pr\left(X|A_1 = A_1^{i}\right) \quad (10)$$

The extrinsic information vectors 1 E (1 A) and 2 E (2 A) contain independent log-ratios on the coded bit sequence C from the outer encoder as shown in Fig.7. The decisions on the message bits are determined like the parallel case by thresholding the output of decoder 1

$$\widehat{X} = h\left(\Pr\left(X|A_1^{i}\right)\right) \quad (11)$$

IV. EXIT CHART ANALYSIS

Extrinsic Information Transfer (EXIT) charts, one of the two recently proposed and successfully used techniques for analysis and design of turbo coded communication systems.4 The main objectives of this chapter are as follows:

- Explain the EXIT charts and their construction methodology in detail.
- Properly extend the EXIT charts method to compressive turbo codes.
- Derive several new analytical results regarding the EXIT charts.
- Extend the EXIT chart performance analysis to analog iterative decoders.

A. EXIT Chart Methodology for Turbo Channel Codes

Step 1: A Step-by-Step Construction of an EXIT Chart

The construction of the EXIT chart can be divided into three steps as explained in this sub-section. Before going into the details of these steps, it is important to note that a memory less source X is used which produces symbols -1 and 1 with probabilities p and (1-p), respectively. The information in the decoding process is expressed in terms of probabilities and to make the decoding process practical, log likelihood ratios (LLR) are used e.g. for a source X with output symbols +1 and -1, the LLR is defined as:

$$L(X) = \ln\frac{P(x=+1)}{P(x=-1)} \quad (12)$$

Thus, from this point on, the input and output of the constituent BCJR decoders will be denoted in the form of LLRs. Prior to presenting the step by step construction of the EXIT charts, it is necessary to present the LLR notation used and define an AWGN channel.

Step 2: Replacement Model of an Actual Constituent Decoder

For the purposes of EXIT chart construction, Fig. 8 (b) shows the replacement model of the constituent BCJR decoder 1. The actual extrinsic information array 1 LA from the original setup in Fig. 8 (a) will be replaced in Fig. 8 (b) by input message bits corrupted by Gaussian noise with a prescribed noise variance, so that $\widetilde{A}_1 = x + n$, where the noise ha zero mean and a prescribed variance σ_n^2 . In literature, this is referred to as passing the message through a a-priori channel with a prescribed noise variance5. The a-priori information, $\tilde{L}$A1, available at the input of the constituent BCJR decoder in Fig. 8 (b) has approximately the same distribution as the actual a-priori information 1 LA available at the input of a actual constituent BCJR decoder 1 in Fig. 7(a). We write this fact as $L_{A1} \overset{i.d.}{\approx} \tilde{L}_{A1}$ and similarly observe the same for the outputs of the BCJR decoders in Fig. 8 (a) and Fig. 8(b), i.e., $L_{E1} \overset{i.d.}{\approx} \tilde{L}_{E1}$. Note that the same procedure is applied for the other constituent BCJR decoder 2 with the subscripts indices 1 and 2 replaced with respect to each other.

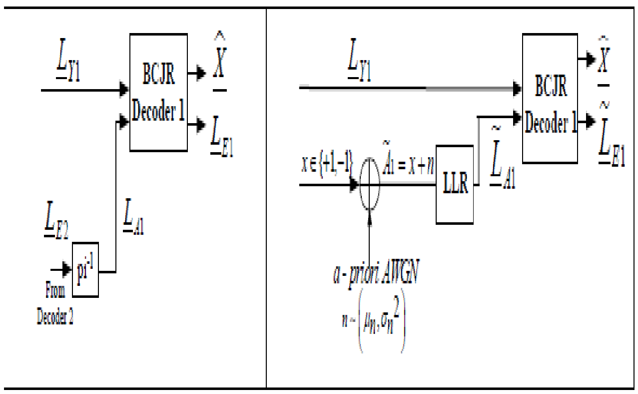

Fig.8(a) Constituent decoder 1 in an actual iterative turbo decoding system; (b) Constituent decoder 1 model used to approximate the a-priori information for the construction of the EXIT chart.

Step3: Construction of an EXIT Curve for a Constituent Decoder

To quantify the amount of decoding information using a single variable, we can evaluate $I(X;L_{A1})$ and $I(X;L_{E1})$ the mutual information between the data source X and the two extrinsic information arrays at the input and the output of the constituent decoder. If $I(X;L_{E1})=0$ there is no information about X in the extrinsic information array, while if $I(X;L_{E1})=H(X)$ the decoder has the information about source message fully available. Furthermore, for different values of $I(X;L_{A1})$ we can also plot $I(X;L_{E1})$ as a function of $I(X;L_{A1})$ in order to graphically observe the improvement achieved by the constituent decoder. This gives an indication about how much information is added by the constituent BCJR decoder and made available to the next decoder. For an actual iterative decoder, achieving the objectives stated in the above paragraph turns out to be analytically untraceable and excessively computationally difficult. Consequently, we will use the replacement model from Fig. 8 (b) for a constituent decoder and evaluate $I(X;\tilde{L}_{E1})$ and $I(X;\tilde{L}_{A1})$ instead of $I(X;L_{E1})$ and $I(X;L_{A1})$.

.

In other words, once the replacement extrinsic information arrays $\tilde{L}$A1 and $\tilde{L}$E1 have been numerically generated, the next step involves the evaluation of the mutual information between these quantities and the symbols from source X, i.e., $I(X;\tilde{L}_{E1})$ and $I(X;\tilde{L}_{A1})$.As proposed in [5], these mutual information quantities can be evaluated for a given source X, constituent BCJR decoder, replacement model from Fig. 7 (b) and σ_n^2 by averaging over many message sequences. The following theorem from [3] presents the formula used for numerical evaluation of the mutual information in the replacement model.

V. PROPOSED SCHEME

Based on this model, the terms "reliability information" and "reliability value" are discussed and some properties are stated. The general model is depicted. Binary info symbols2 U ∈ B that are independent and uniformly distributed are transmitted over a binary input symmetric memory less channel (BISMC) U → V . The channel outputs V ∈ R are required to be "LLR-like"; i.e., when estimating the transmitted info symbol u on basis of a soft-value v, the error probability is assumed to be large for small magnitudes of v, and it is assumed to be small for large magnitudes of v. A precise definition of "LLR-like" is given below. The channel output V is considered as an observation and the corresponding posteriori LLR L ∈ R, is computed. For a given channel output v, we define

$$l := L(U|V=v) = \ln \frac{\Pr(U=+1|V=v)}{\Pr(U=-1|V=v)}. \tag{13}$$

Assume a BISMC U → V. The soft-value V is called LLR-like if L(U|V = v) is monotonically increasing in v. This property has two implications: first, the channel U → L is a BISMC, and second, the sign of v is the ML (and MAP) estimate for U. Thus, if a soft-value is LLR-like, the soft-value and the corresponding LLR differ only in their absolute values, and this difference is "well-behaved" in some sense. If a soft-value is LLR-like but no LLR, we may call it mismatched LLR in the sequel3 . This model is very general and includes in particular the following three scenarios:

- A symbol is transmitted over an AWGN channel and the channel output is converted to an LLR assuming a wrong SNR. The channel U → V models the channel between the symbol and the mismatched LLR.
- Assume a transmission system with linear channel encoding and MaxLogAPP decoding. Then, the channel U → V models the channel between an info symbol and its mismatched a-posteriori LLR.
- Assume a transmission system with linear channel encoding and iterative decoding. Then, the channel U → V models the channel between an info or a code symbol and its mismatched extrinsic LLR. Notice that possible dependencies between info or code symbols and possible dependencies between soft-values are explicitly not taken into account, as we investigate only the symbol-wise reliability of soft-values.

Algorithm Apoproached:

Step1: If (llrs (index) == -Inf)
Step2:Histogram(bits(index)+1,1)=histogram(bits(index)+1, 1)+1;
Step3: Elseif (llrs(index) == Inf)

Step4:Histogram(bits(index)+1,bin_count)=histogram(bits(index)+1,bin_count)+1;
Step5: else
Step6: If (lots_of_bins == true)
Step5: If (bin_width > 0.0)
Step6:Histogram(bits(index)+1,floor(llrs(index)/bin_width)-bin_offset+1)=histogram(bits(index)+1,floor(llrs(index)/bin_width)-bin_offset+1)+1;
Step7: else
Step8:Histogram(bits(index)+1,2)=histogram(bits(index)+1,2)+1;
Step9: end
Step10: else
Step11: Histogram(bits(index)+1,bits(index)+2)=histogram (bits (index)+1,bits(index) +2)+1;
Step12: End
Step13: End

VI. RESULTS

Results of this paper is shown in bellow Figs.9 to 13.

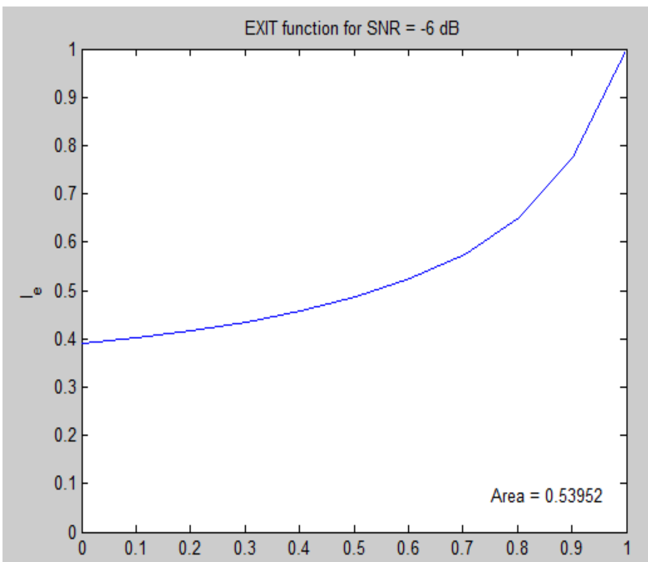

Fig.9. exit function for SNR -6dB.

A. 3 D EXIT functions

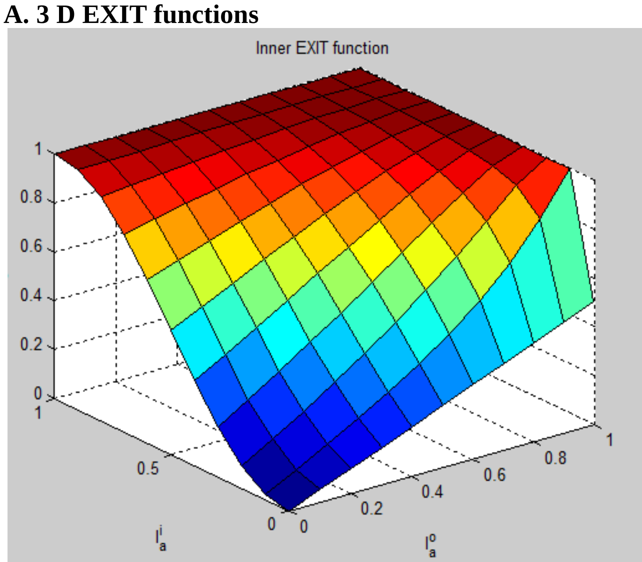

Fig.10. 3D inner EXIT function.

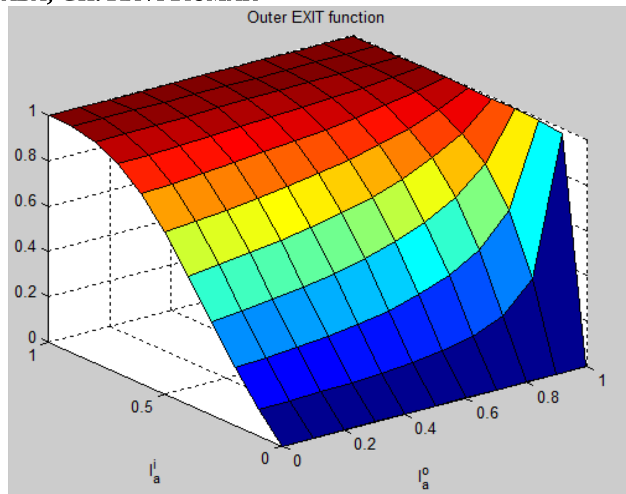

Fig.11. 3D outer exit function.

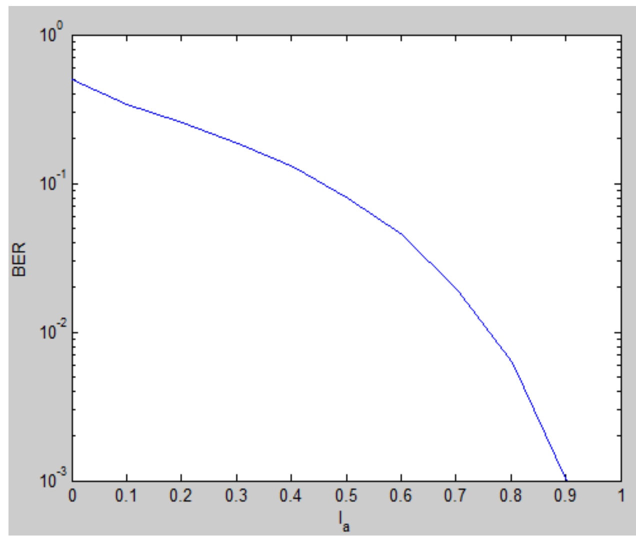

Fig.12. BER performance with the input energy.

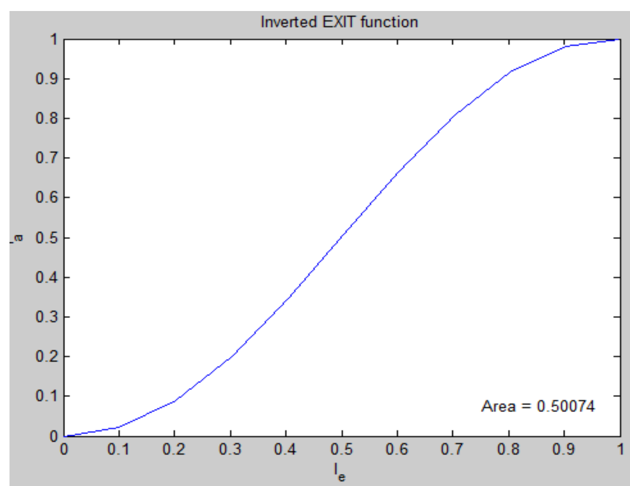

Fig.13. Inverted EXIT function between Input and the error with respect to area.

VII. CONCLUSION

LLR transfer can be improved by matching decoder metrics to the actual error distribution. Adequate design of the LLR's index assignment can provide additional resilience with histogram. Although the commonly adopted

NBC is better than "randomly" assigned indices, an optimization of the IA based on the EXIT characteristic of the Turbo decoder yields yet improved performance without increasing computational complexity of the decoder itself. In this letter we have shown that an easily computable approximation of the objective function enables efficient optimization of this complex combinatorial problem. Supporting numerical results have been provided for AWGN channels.

Future Scope: This motivated the employment of an inner URC codec, which used MBPSK = 2 Binary Phase Shift Keying (BPSK) to facilitate transmissions over an uncorrelated narrowband Rayleigh fading channel, as shown in Figure 10.1. Here, the maximum effective throughput was equal to the maximum capacity of log2 (MBPSK) = 1 bit per channel use and no effective throughput loss was incurred. Indeed, the areas beneath the URC EXIT functions provided were found to be equal to the corresponding channel capacities, as predicted by the area property of EXIT charts.

VIII. REFERENCES

[1] A. Aaron and B. Girod, "Compression with side information using turbo codes," in Proc. IEEE Data Compr. Conf., pp. 252-261, Snowbird, UT, April 2002.

[2] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: Model and erasure channel properties," IEEE Transactions on Information Theory, vol. 50, no. 11, pp. 2657–2673, November 2004.

[3] A. Ashikhmin, G. Kramer, and S. ten Brink, "Extrinsic information transfer functions: A model and two properties," in Proc. Conf. Information Sciences and Systems, pp. 742-747, Princeton, NJ, March 20–22, 2002.

[4] A. Ashikhmin, G. Kramer, and S. ten Brink, "Code rate and the area under extrinsic information transfer curves," in Proc. IEEE Int. Symp. Information Theory (ISIT 2002), p. 115, Lausanne, Switzerland, July 2002.

[5] AT&T, "Milestones in AT&T history," available at www.att.com/history/milestones.html.

[6] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," IEEE Transactions on Information Theory, vol. 20, pp. 284-287, March 1974.

[7] J. Bajcsy, "Course notes ECSE 686B – Codes and Graphs," Department of Electrical and Computer Engineering, McGill University, Winter 2006.

[8] J. Bajcsy, C. V. Chong, J. A. Hunziker, D. A. Garr, and H. Kobayashi, "Iterative decoding in some existing systems," IEEE J. Select. Areas Commun., vol. 19, no. 5, pp. 883 – 890, May 2001.

[9] S. Benedetto and G. Montorsi, "Unveiling turbo-codes: Some results on parallel concatenated coding schemes," IEEE Trans. Inform. Theory, vol. 42, no. 2, pp. 409 – 428, March 1996.

[10] T. Berger, Rate Distortion Theory: A Mathematical Basis for Data Compression. Englewood Cliffs, NJ: Prentice-Hall, 1971.