Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 51(65), Fascicola 2, 2006

The Minimum Likelihood APP Based Early Stopping Criterion for Multi-Binary Turbo Codes

Horia Balta[1], Catherine Douillard[2], Maria Kovaci[1]

Abstract – This paper presents a simple and efficient criterion for stopping the iteration process in multi-binary symbol turbo-decoding with a negligible degradation of the error performance. The criterion is devised starting to minimum log-likelihood ratio (LLR) based stopping criterion used for binary turbo codes (BTC). Two variants consist in particularizations of the same idea in the MAP and MaxLogMAP decoding algorithm cases. The proposed two variants criterion has efficiency close to the optimum (genie) criterion and is simple to perform.

Keywords: Iterative decoding, stopping criterion, multi-binary turbo codes.

I. INTRODUCTION

A classical turbo-code [1] calls for a parallel concatenation of two single-binary recursive, systematic convolutional codes (RSC) (with 1/2 coding rate). In the decoding process, the corresponding constituent decoders exchange extrinsic information through an iterative process.

With each iteration in the turbo decoding, the signal to noise ratio (SNR) required to obtain a specified bit-error rate (BER) decreases [1]. But the improvement in SNR becomes smaller with each iteration.

For the SNR of practical interest, after a limited number of iterations (3 or 6), the turbo decoder corrects the received block and is able, through hard decision, to retrieve the transmitted original data sequence. Only for a small proportion of the received blocks, the turbo decoder must perform a greater iteration number (8 or 15) to manage the total correction or in big proportion of these blocks. But this computational effort will be reflected by a consistent diminution of the BER.

Thus, it becomes rightful to perform a different iteration number for each received block in part, number which will be established by an early stopping iterations criterion. By the diminution of the computational effort, the using of such stopping iterations criterion will bring also others advantages: the decrease of the decoding average time (in the case of buffers use), the increasing possibility of the maximum number of the imposed iterations, the decrease in the used power in the decoding.

But, on the other side, the stop criterion must not alter the BER performance obtained through the realization of all iterations. The utility of such iteration stopping criterion, which constitutes an optimal compromise between the two constraints (the elimination of the unnecessary iterations and the conservation of the BER performance) is proved by the considerable number of publications on this theme, for instance [2] [3] [4] [5].

These publications apply to BTC. Owing to the decoding of the specific Multi-Binary Turbo Codes (MBTCs), [6], the stop criteria, built for BTC, have not the same efficiency for MBTCs, or they can even not be applied to MBTCs.

After an extensive review of existing stopping criteria, this paper presents a new stopping criterion, in two variants, applicable in the case of the symbol decoding [7] used in the MBTCs case. The two variants of the stopping criterion represent its adaptation to the MAP and, respective, Max Log MAP decoding algorithms.

II. TURBO DECODING AND EARLY STOPPING CRITERIA

Through this paragraph, after a short presentation of the used notations, we review the existing early stopping criteria.

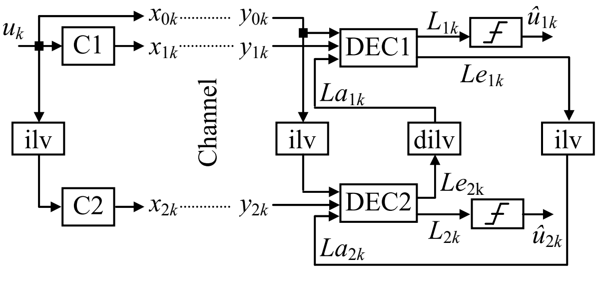

Fig. 1 A Turbo-Code scheme.

[1] Department of Communications, Faculty of Electronics and Telecommunications, Bd. V. Pârvan Nr. 2, 300223 Timişoara, e-mail: balta@etc.upt.ro, kmaria@etc.upt.ro

[2] Electronic Department,ENST Bretagne - Technopôle de Brest Iroise CS 83 818 - 29238 BREST Cedex 3, France, e-mail: Catherine.Douillard@enst-bretagne.fr

For the beginning, we consider the BTC that consists of two rate-1/2 recursive and systematic convolutional codes (RSC), shown in Fig.1. Let $\boldsymbol{u} = (u_1, u_2, \ldots u_N)$ be an information block of length N and $\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots \boldsymbol{x_N})$ be the corresponding coded sequence, where $\boldsymbol{x_k} = (x_{0k}, x_{1k}, x_{2k})$, for $k \in I = \{1, 2, \ldots N\}$ is the output code block at time k. Assuming BPSK transmission over an AWGN channel, u_2 and x_{jk} all take values in $\{-1, +1\}$ for $k \in I$ and $0 \le j \le 2$. Let $\boldsymbol{y} = (\boldsymbol{y_1}, \boldsymbol{y_2}, \ldots \boldsymbol{y_N})$ be the received sequence, where $\boldsymbol{y_k} = (y_{0k}, y_{1k}, y_{2k})$ is the received block at time k. Then $y_{jk} = x_{jk} + w_{jk}$, where w_{jk} is a Gaussian random variable with zero mean and variance σ^2. Like it is shown in Fig.1, $\boldsymbol{\hat{u}_n} = (\hat{u}_{n1}, \hat{u}_{n2}, \ldots \hat{u}_{nN})$, with n =1 or 2, we may denote the estimate of $\boldsymbol{u}$ given by DEC1 and DEC2 respectively. We also note by La_{nk}^{i}, Le_{nk}^{i} and L_{nk}^{i} the a priori information, the extrinsic one and the log-likelihood ratio (LLR), respectively, from decoder n for the u_k bit at ith iteration.

Mainly, an early stopping rule consists in a comparison of a measure, calculated after each iteration, with a threshold µ. In the following, we briefly present some of the existing main stopping criterions.

(1) Cross Entropy (CE) [2]. Based on the assumptions presented in [3] the CE between the distribution of the estimates at the outputs of the decoders at iteration i can be approximated by:

$$T(i) \approx \sum_{k=1}^{N} \frac{\left(Le_{2k}^{i} - Le_{2k}^{i-1}\right)^2}{e^{\left|L_{1k}^{i}\right|}} \qquad (1)$$

The decoding process is stopped after iteration i for $i \ge 2$, if:

$$\frac{T(i)}{T(1)} < \mu , \qquad (2)$$

where $T(1)$ is the approximated CE after the first iteration and the threshold μ is around 10^{-3}.

(2) Sign-Change Ratio (SCR) [3]. In the SCR criterion the decoding will be stopped at the ith iteration if the ratio $C(i)/N$ is lower than µ where $C(i)$ is the number of the sign differences between Le_{2k}^{i-1} and Le_{2k}^{i}. The threshold µ can takes values between 0.005 and 0.01 as a function of SNR and N.

(3) Sign-Difference Ratio (SDR) [4]. The SDR criterion is a replica of SCR in which $C(i)$ is calculated as the number of sign differences between La_{2k}^{i} and Le_{2k}^{i}.

(4) Hard Decision-Aided (HDA) [3]. Proceeding from the HDA criterion, the decoding process is stopped after iteration i for $i \ge 2$, if:

$$sign\left(L_{2k}^{i}\right) = sign\left(L_{2k}^{i-1}\right) \quad \forall k \in I. \qquad (3)$$

(5) Improved Hard Decision-Aided (IHDA) [8]. According to IHDA, at iteration i, we compare the hard decisions of the information bit based on $L_{2k}^{i} - Le_{2k}^{i}$ with the hard decision based on L_{2k}^{i}. If they agree with each other for the entire block, the decoding process is terminated at iteration i.

(6) Mean Estimate (ME) [5]. After each iteration i the mean $M(i)$ of the absolute values of the LLRs is calculated, and the decoding process is stopped if:

$$M(i) = \frac{1}{N} \cdot \sum_{k=1}^{N} \left|L_{2k}^{i}\right| > \mu . \qquad (4)$$

(7) Minimum LLR (mLLR) [9]. The mLLR rule stops the decoding process after iteration i for $i \ge 1$, if:

$$\min_{1 \le k \le N} \left|L_{2k}^{i}\right| < \mu \qquad (5)$$

(8) Sum-Reliability (SR) [10]. After each iteration i the sum of the absolute values of the LLRs is calculated:

$$S(i) = \sum_{k=1}^{N} \left|L_{2k}^{i}\right| \qquad (6)$$

and the decoding process is stopped after iteration i for $i \ge 2$, if $S(i) \le S(i-1)$.

(9) CRC Rule (CRC). A separate error-detection code, especially a CRC, can be concatenated as an outer code with an inner Turbo Code in order to flag erroneous decoded sequences. The decoding process is stopped after iteration i whenever the syndrome of the CRC is zero.

(10) Genie (GENIE). The GENIE (optimum) criterion can be used in the case where the original information bits are known. Then, the iteration is stopped immediately after the frame is correctly decoded. This unrealizable criterion gives a limit for all possible criteria.

Other criteria based on the presented ones have also been developed. In [11] a two decision threshold and a measure are introduced to enhance performance and simplify the stopping (CE) criterion. Combinations of these techniques are used too. In [5] two such combined rules (ME combined with HDA give the MSC criterion which is further combined with CRC rule) are presented. Another combination, between SR and mLLR stopping rules is done in [10]. An interesting idea is presented in [12]. The proposed (bit level) criterion stops the decoding process just for same bits (which are satisfying the rule). The iterations are performed only for the remaining bits, thus decreasing the computation volume. The simulations show that the bit level stopping is particularly effective for fading channels. Another useful idea, presented in [13], consists of taking into account and stopping also the non-convergent frames.

All of the presented stopping criterions were developed for binary turbo-codes. Considering the MBTCs with symbol decoding [6], some of these rules are not applicable (or not in that form). This fact is due to the disappearance of the LLR and their substitution by a posteriori probability (APP).

Note: When symbol decoding is performed, the decoder can compute Log-APPs or, in practice, normalized Log-APPs that can be considered as LLRs (this is the case in hardware decoders).

Moreover, we can not calculate the CE in the form (1). Also, because the La_{nk} and Le_{nk} constitute the pure probabilities, we can not speak about their sign. So, only the CRC stopping rule can be performed for MBTCs without any changes.

Remark: We say that the above stopping criteria can't be directly applied to MBTC, but they can probably be modified in order to cope with these codes.

With a small change the HDA and the derived IHDA could be adapted for MBTC. It can be made in two ways. The differences between symbols could be encountered without any other modification in the decoding structure. This means that the maximum likelihood (ML) sequences after two consecutives iterations will be compared. Some of these criteria are presented in [14] and [15]. But, as an alternative, the symbols could be decomposed in bits, and the HDA or its derived IHDA could be applied in original form over these bits.

In this paper we present an early stopping criterion, in two variants adapted for MAP and MaxLogMAP MBTC's algorithms, which is an adaptation of the mLLR rule for the MBTC case.

III. THE PROPOSED EARLY STOPPING CRITERION

Considering that the turbo code depicted in Fig.1 is multi-binary, as we mentioned above, we must replace the LLR by APP or Log-APP. With this only change in meaning we can keep all the notations, but we must add one more index for the input number. So, let R be the inputs number and r the index of these inputs. Consequently we have $\boldsymbol{u} = (\boldsymbol{u_1}, \boldsymbol{u_2}, \ldots \boldsymbol{u_N})$ the symbols information block with $\boldsymbol{u_k} = (u_{1k}, u_{2k}, \ldots u_{Rk})$; $\boldsymbol{x_j} = (\boldsymbol{x_{j1}}, \boldsymbol{x_{j2}}, \ldots \boldsymbol{x_{jN}})$ the coded sequences, with $\boldsymbol{x_{jk}} = (x_{1jk}, x_{2jk}, \ldots x_{Rjk})$ and $x_{rjk} = \pm 1$; $\boldsymbol{y_j} = (\boldsymbol{y_{j1}}, \boldsymbol{y_{j2}}, \ldots \boldsymbol{y_{jN}})$ the received sequences, with $\boldsymbol{y_{jk}} = (y_{1jk}, y_{2jk}, \ldots y_{Rjk})$ and $y_{rjk} = x_{rjk} + w_{rjk}$, where w_{rjk} is a Gaussian random variable with zero mean and variance σ^2; $\hat{\boldsymbol{u}}_n = (\hat{\boldsymbol{u}}_{n1}, \hat{\boldsymbol{u}}_{n2}, \ldots \hat{\boldsymbol{u}}_{nN})$, with $\hat{\boldsymbol{u}}_{nk} = (\hat{u}_{1nk}, \hat{u}_{2nk}, \ldots \hat{u}_{Rnk})$ and $n = 1$ or 2, denotes the estimate of $\boldsymbol{u}$ given by DEC1 and DEC2 respectively. For simplicity, we shall consider that $\boldsymbol{u_k}$ and $\hat{\boldsymbol{u}}_{nk}$ are integers from the $J = \{0, 1, \ldots(2^R\text{-}1)\}$ set. Thus, in the MAP decoding algorithm, $La^i_{nk}(d)$, $Le^i_{nk}(d)$ and $L^i_{nk}(d)$ represent the a priori, the extrinsic and the a posteriori (APP) probabilities that the n decoder estimates (after ith iteration) the original u_k symbol at d integer, i.e. the above probabilities that $\hat{\boldsymbol{u}}_{nk} = d \in J$. We also note that an information block contains N symbols and $R{\times}N$ bits.

Considering the decoding of a convergent block, as the iterative process advances, the APP probabilities corresponding to the original symbols sequence take values close to 1. Because of that, any other APP probabilities, i.e. $L^i_{nk}(d)$ with $d \neq u_k$, will take values close to 0. We use this fact to construct an early stopping criterion:

The iterative decoding process is stopped at iteration i if, at any time k, an APP probability value is greater than an imposed threshold μ:

$$\textit{stop iterations if } \forall k \in I,\ \exists d \in J \textit{ so that } \mu < L^i_{2k}(d) < 1. \tag{7}$$

We call this criterion the minimum likelihood APP (mlAPP). However, the mlAPP rule could not be used in this form for the case of the MaxLogMAP decoding algorithm. This is because the transfer of the APP values in the log domain.

For the MaxLogMAP case we consider the following judgement. If $d = \boldsymbol{u_k}$ than $L^i_{2k}(d) \approx -log(\mathrm{P}\{\hat{\boldsymbol{u}}_{nk} = \boldsymbol{u_k}\} \approx 1)$ so $L^i_{2k}(d) \to 0$ when i is increasing. If $d \neq \boldsymbol{u_k}$, then $L^i_{2k}(d) \approx -log($*a probability that is very small*$)$ so $L^i_{2k}(d) >> 0$. (In the previous judgement we suppose that in MaxLogMAP decoding algorithm normalization has been performed between the APP values corresponding to the same k time, i.e. from all APP values has been subtracted the minimum one.)

Thus, the mlAPP stopping criterion for the MaxLogMAP algorithm case is:

The iterative decoding process is stopped at iteration i if, at any time k, all the APP probabilities, except the zeros one, are bigger than an imposed threshold μ:

$$\textit{stop iterations if } \forall k \in I \textit{ and } \forall\ L^i_{2k}(d) > 0 \textit{ result that } L^i_{2k}(d) > \mu\,. \tag{8}$$

Or in another formulation:

$$\textit{stop iterations if } \min_{d \in J^*}\left\{L^i_{2k}(d)\right\} > \mu\,,\ \forall k \in I \tag{8'}$$

where $J^* = J \setminus \{d^*\}$ and d^* is the value of the k symbol for which $L^i_{2k}(d^*) = 0$.

IV. SIMULATIONS RESULT

We have performed simulations using the 8 state double-binary turbo code and the interleaving defined in [6], with data block length $N = 752$ double-binary symbols, that is = 1504 bit data blocks. We assumed a transmission over an AWGN channel, using QPSK modulation and no quantization is performed at the decoder input.

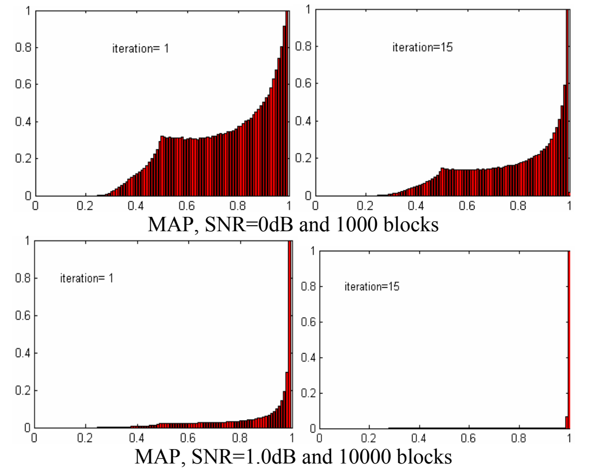

Fig. 2 APP normalized histograms for MAP decoding algorithm

The diagrams from Fig.2 and Fig.3 present the APP normalized histograms for both MAP and MaxLogMAP cases, performing 15 iterations without stop. The histogram's normalization means that all the obtained values have been divided by the biggest value after each iteration. In the MAP case only the APP for the maximum likelihood symbols were considered. The simulations show that these APP values get near to 1 when the SNR and iteration *i* increasing.

In the MaxLogMAP case all APP values were taken into account. Except for the maximum likelihood symbols APPs (which all are zero) the other APPs take values bigger when the SNR and *i* is increasing.

These results confirm the previous suppositions about the APP values. The remaining question is what the optimum values for the threshold μ are?

The BER and FER curves from Fig.4 compare the turbo-code performances with or without stop criterion. We used thresholds with the values: 0.99 and 0.9999 for MAP and 4 and 10 for MaxLogMAP. The results show that the second threshold values for each algorithm give practically the same performance as is the case without iterations stop.

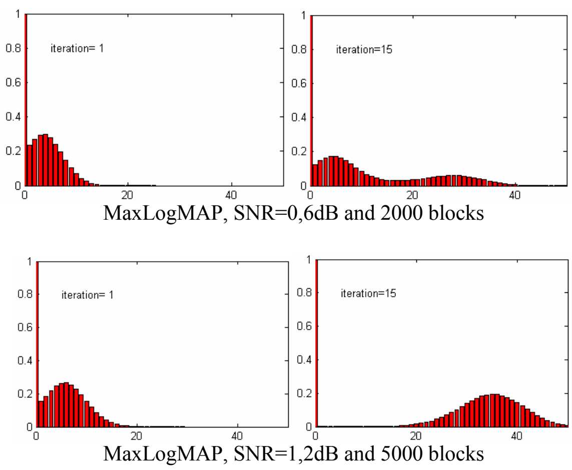

Fig. 3 APP normalized histograms for MaxLogMAP decoding algorithm

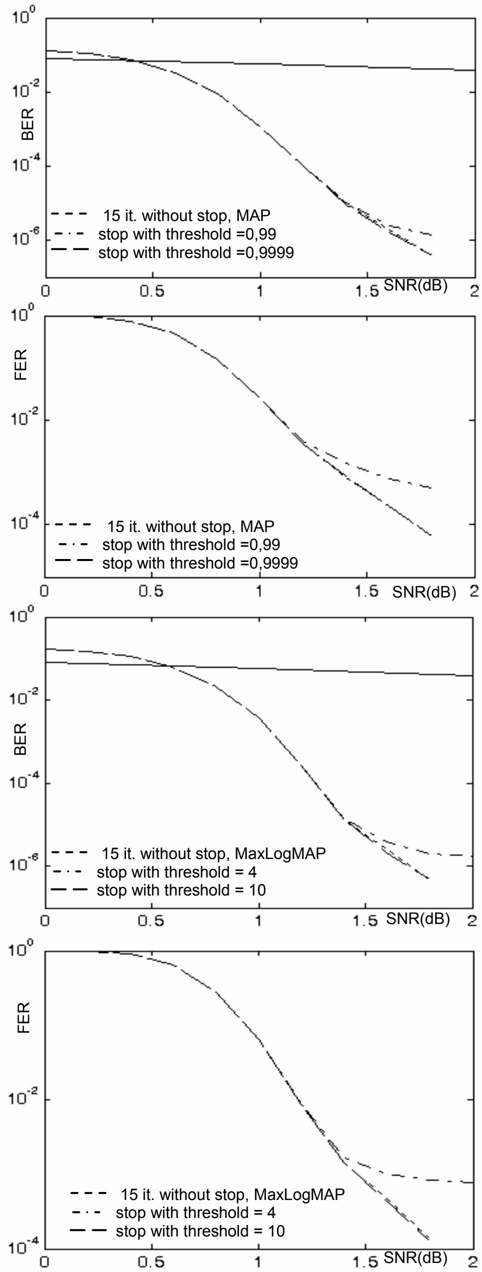

Fig. 4 BER and FER performance for two values of the threshold μ

In order to evaluate the efficiency of the proposed mlAPP criteria in Fig.5 we have plotted the average number of performed iterations as a function of SNR, and we have compared with the cases "without stop" and "genie". The genie criterion stops the iteration when (and only when) there are no more errors in the decoded block.

V. CONCLUSIONS

In this paper we present a new early stopping criterion, mlAPP, usable in the multi-binary turbo-codes cases when the symbol decoding is performed. The proposed criterion has two variants corresponding to the MAP and the MaxLogMAP decoding algorithm.

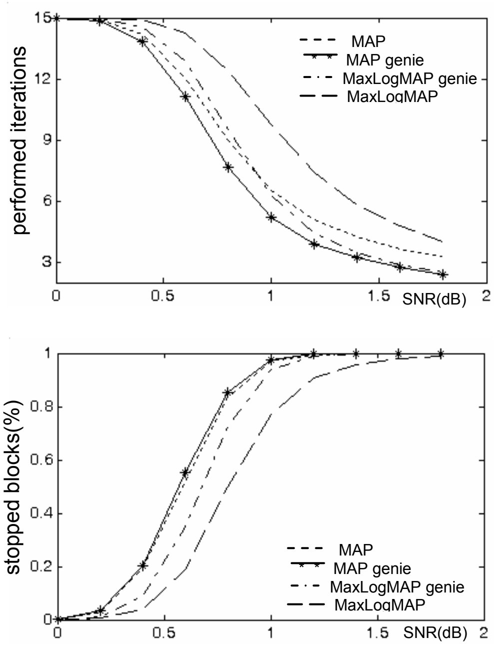

Fig. 5 The mlAPP stoping criterion efficiency.

The simulations show that both variants perform without any differences compared to the case of fixed number iterations and have the same efficiency close to the optimum (genie) stopping criterion.

REFERENCES

[1] C. Berrou, A. Glavieax, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbocodes", Proc. ICC'93, Switzerland, May 1993, pp.1064-1070

[2] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes", IEEE Trans. Inf. Theory, 1996, 42, pp.429-445

[3] Rose Y. Shao, Shu Lin, and Marc P. C. Fossorier, "Two Simple Stopping Criteria for Turbo Decoding", IEEE Trans. Comm., vol.47, No.8, August 1999, pp.1117-1120

[4] Yufei Wu, Brian D. Woerner, and William J. Ebel, "A Simple Stopping Criterion for Turbo Decoding", IEEE Comm. Lett., vol.4, No. 8, August 2000, pp.258-260

[5] Fengqin Zhai, and Ivan J. Fair, "Techniques for Early Stopping and Error Detection in Turbo Decoding", IEEE Trans. Comm., vol.51, No. 10, October 2003, pp. 1617-1623

[6] C. Douillard, and C. Berrou, "Turbo Codes With Rate-m/(m+1) Constituent Convolutional Codes, IEEE Trans. Comm., vol.53, No. 10, Ocober, 2005, pp.1630-1638

[7] J. Tan, G. L. Stüber, "A MAP equivalent SOVA for non-binary turbo codes", in Proc. *IEEE International Conference on Communications, ICC*, pp. 602-606, New Orleans, L. A., June 2000.

[8] T. M. N. Ngatched, and F. Takawira, "Simple stopping criterion for turbo decoding", IEEE Electronics Letters, vol. 37, No. 22, October 2001, pp.1350-1351

[9] A. Matache, S. Dolinar, and F. Pollara, "Stopping Rules for Turbo Decoders", TMO Progress Report 42-142, August 2000, http://tda.jpl.nasa.gov/progress_report/, Jet Propulsion Laboratory, Pasadena, California

[10] Gilben Frank, Frank Kienle, and Norbert Wehn, „Low Complexity Stopping Criteria for UMTS Turbo-Decoders", in Proc. *Vehicular Techn. Conf. (VTC Spring '03)*, Jeju, Korea, April 2003.

[11] Nam Yul Yu, Min Goo Kim, Yong Serk Kim, and Sang Uoon Chung, "Efficient stopping criterion for iterative decoding of turbo codes", IEEE Electronics Letters, vol.39, No.1, January 2003, pp.73-74

[12] Dong-Ho Kim, and Sang Wu Kim, "Bit-Level Stopping in Turbo Decoding", *IEEE Communication Letters*, vol.10, March 2006, pp. 183-185.

[13] A. Taffin**,** "Generalised stopping criterion for iterative decoders", IEEE Electronics Letters, vol.39, No.13, June 2003, pp.993-994

[14] A. Hunt, S. Crozier, K. Gracie and P. Guinand, "A completely safe early stopping criterion for max-log Turbo code decoding", in Proc. *4th International Symposium on Turbo Codes and Rlated Topies*, Munich, Germany, April 3-7, 2006

[15] K. Gracie, A. Hunt, and S. Crozier, "Performance of Turbo Codes using MLSE-Based Early Stopping and Path Ambiguity Cheching for Inputs Quantized to 4 Bits", in Proc. *4th International Symposium on Turbo Codes*, Munich, April 3-7, 2006.