ISSN: 2229-6948(ONLINE)
DOI: 10.21917/ijct.2016.0189

ENHANCEMENT OF ITERATIVE TURBO DECODING FOR HARQ SYSTEMS

Imed Amamra[1] and Nadir Derouiche[2]
Laboratoire de Recherche en Electronique de Skikda, Université of 20 Août 1955 of Skikda, Algeria
E-mail: [1]amrimedz@yahoo.fr, [2]nadirouiche@yahoo.fr

Abstract

This paper presents a new method for stopping the iterative turbo decoding. First, a bit-level convergence test using the cross-entropy analyses is used to select non converged bits and establish a simple and effective stopping rule. Next, an adaptive approach is used to compute a scaling factor for normalizing the extrinsic information of the previously selected bits. The extra coding gain obtained from this normalization can compensate for the performance degradation of the stopping rule. The simulation results of the proposed stopping criterion show an interesting application in a hybrid automatic repeat request systems with turbo coding scheme, where the decoding complexity can be fairly reduced.

Simulation results of the proposed criterion, in comparison with previously published stopping rules, were presented for illustrating the adaptive termination according to a changing SNR environment.

Keywords:

Cross Entropy, HARQ, Iterative Decoding, Stopping Criterion, LogMAP, Turbo Code

1. INTRODUCTION

With the adoption of advanced channel coding in the physical layer of wireless systems such as 3G and 4G mobile systems or the WiMAX (802.16), error control protocols are combined with forward error correction. Turbo codes [1] have already been incorporated in broadband mobile wireless communication standards.

It is well known that turbo codes with iterative decoding can achieve near Shannon limit performance. However, this limit is achieved in general with a large number of iterations and since, the complexity increases linearly with the number of iterations. So, if the chosen iteration depth is high, the achieved performance is not worth the resulting computational overhead. The development of efficient stopping criteria for the iterative process have attracted much attention and researches were published to decrease this overhead as the works of [2]-[5].

Under severely unreliable channel with conventional stopping criteria, iterative decoding of turbo codes frequently fails and therefore most frames hit the predetermined maximum number of iterations, hence such early stopping criterion would not save unnecessary computational power and decoding latency.

In real-time communication, the reliability can be enhanced by combining forward error correction codes with automatic repeat request (*ARQ*), known as hybrid-ARQ (*HARQ*) protocols. However, many research efforts will mostly concern about turbo decoding combined in *HARQ* schemes. In many severe environments, when codewords are not successfully decoded at the receiver, a *NAK* (Negative Acknowledgement) message is fed back to the transmitter for requesting the retransmission. Frequent decoding failures imply that the decoder reaches the maximum number of iterations many times and consequently most of computational power is wasted on failed decoding if a codeword is retransmitted many times.

It is a bit ironical to consume most of power on decoding failures, but it is hard to detect bad codewords prior to decoding because of the imperfect knowledge on the received signal characterization of decoding failure. In this case it is important to derive an effective criterion to stop the iterative process, which can meet the conventional stopping criterion performance at high *SNR*, as well as to early terminate the iterative decoding at low *SNR*. Meanwhile a *HARQ* scheme can save the computation power and increase the decoder throughput.

In the literature, numerous early stopping criteria have been introduced to reduce complexity and power consumption by terminating the decoding process as soon as the intermediate decoding result is reliable enough and further improvement is unlikely. Some known stopping criteria, such as cross-entropy (*CE*) [3], sign change ratio (*SCR*) [4], the number of sign difference between likelihoods (*NSD*) [5], hard decision aided (*HDA*) [4] and histogram of the log-likelihood ratio (*HLLR*) [6], offer a trade-off between speed and performance. Although, many of them are efficient for solvable frames at high signal-to-noise ratios (*SNR*) they fail for the unsolvable ones where already the decoder has no capability to decode and further iterations are only a waste of time. In addition to, some techniques are constrained by a complete iteration to decide whether to terminate the iterative decoding or not, so the minimum controllable number of iterations is one.

In this paper, it is assumed that the decoder may encounter the unsolvable situation where there is too little information for the decoder to work correctly. In this case, it is important to derive an effective criterion to stop the iterative process, so as to prevent from unnecessary computations and decoding delay.

In addition to, a normalization technique is combined with a proposed stopping criterion, in order to compensate for the performance loss due to early stopping the iterative decoding. So the normalization is performed with an adaptive correcting factor which adjusts the too optimistic reliability-information in the SISO decoder for unreliable channel conditions.

The rest of the paper is organized as follows. Section 2 starts with a brief review of the iterative maximum a posteriori decoding algorithm in the log domain. The formulations of the cross-entropy based information measurements and observations are introduced in section 3. A description of the proposed stopping criterion is given in section 4. In section 5, extensive computer simulations were performed to validate the given arguments. Finally, the conclusion is drawn in section 6.

2. ITERATIVE TURBO DECODING ALGORITHM

In this section a turbo code is considered, which consists basically of two parallel concatenated encoders. Each component is a recursive systematic convolutional (RSC) code. At the receiver, an iterative decoder based on the log-MAP algorithm is used as shown in Fig.1.

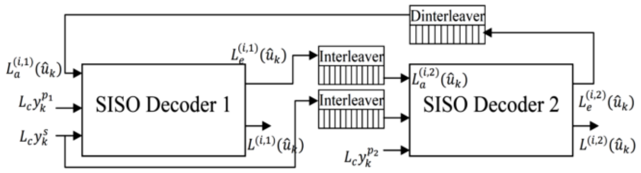

Fig.1. Iterative turbo decoder

Let $u = \{u_k,\ 1 \le k \le N\}$ be a data block of length N to be transmitted. At the receiver, $\left(y_k^s, y_k^{p_1}, y_k^{p_2}\right)$ are the systematic and two parity symbols respectively, corresponding to u_k. These symbols are fed to two Soft-Input Soft-Output (SISO) decoders (using log-MAP algorithm) to produce the estimates $\hat{u} = \{\hat{u}_k, 1 \le k \le N\}$. Then, in the i^{th} iteration the first SISO decoder receives the channel sequence $\left(L_c y_k^s, L_c y_k^{p_1}\right)$ and the a priori information $L_a^{(i,1)}(\hat{u}_k)$ provided by de-interleaving the extrinsic information $L_e^{(i-1,2)}(\hat{u}_k)$ of the second SISO decoder, where L denotes the log-likelihood ratio values. Hence it calculates an improved a-posteriori information $L^{(i,1)}(\hat{u}_k)$. After that, the second SISO decoder uses the interleaved sequence $\left(L_c \tilde{y}_k^s, L_c y_k^{p_2}\right)$ and the a priori information $L_a^{(i,2)}(\hat{u}_k)$ derived by interleaving $L_e^{(i,1)}(\hat{u}_k)$, to calculate the a-posteriori information $L^{(i,2)}(\hat{u}_k)$.

So each SISO decoder delivers two output values $L^{(i,m)}(\hat{u}_k)$ and $L_e^{(i,m)}(\hat{u}_k)$, that are the LLR of the information bit $\hat{u}_k$ and the extrinsic estimate respectively, where the sign of the LLR values gives the hard decision for that symbol and the magnitude quantifies the decision certainty. Here m indicates the first or the second decoder ($m = 1, 2$).

The performance improvement with increasing iterations tends to be insignificant as the number of iterations gets large, because of the diminishing effect. Consequently researchers are motivated to find an appropriate and efficient criterion for stopping the iterative process in order to increase the average decoder throughput or to reduce the average power consumption. It is shown in [3] that:

$$L^{(i,1)}(\hat{u}_k) = L_c y_k^s + L_a^{(i,1)}(\hat{u}_k) + L_e^{(i,1)}(\hat{u}_k) \tag{1}$$

$$L^{(i,2)}(\hat{u}_k) = L_c \tilde{y}_k^s + L_a^{(i,2)}(\hat{u}_k) + L_e^{(i,2)}(\hat{u}_k) \tag{2}$$

where,

$$L_a^{(i,1)}(\hat{u}_k) = \Pi^{-1}\left[L_e^{(i-1,2)}(\hat{u}_k)\right] \tag{3}$$

$$L_a^{(i,2)}(\hat{u}_k) = \Pi\left[L_e^{(i,1)}(\hat{u}_k)\right] \tag{4}$$

$\Pi[\cdot]$, $\Pi^{-1}[\cdot]$ denote interleaving and de-interleaving operations respectively.

3. BIT-SELECTION USING CROSS-ENTROPY CONVERGENCE TEST IN ITERATIVE DECODING

Suppose that $P(\hat{u})$ and $Q(\hat{u})$ are two a-posteriori distributions of subsequent decoding operations. The cross-entropy, also known as discrimination information, between $P(\hat{u})$ and $Q(\hat{u})$ is defined as the expectation over $Q(\hat{u})$:

$$E_p\left\{\log \frac{P(\hat{u})}{Q(\hat{u})}\right\} = \int_{\hat{u}} P(\hat{u}) \log\left(\frac{P(\hat{u})}{Q(\hat{u})}\right) d\hat{u}. \tag{5}$$

This gives a measurement of the difference between the two distributions. Assuming that the LLRs of the information bits are statistically independent after each iteration, it was shown in [3],[5],[7] that the cross entropy for bit u_k can be approximated as:

$$T^{(i,m)}(u_k) \approx \frac{\left|\Delta L_e^{(i,m)}(\hat{u}_k)\right|}{\exp\left(\left|L^{(i,m)}(\hat{u}_k)\right|\right)}. \tag{6}$$

The block (frame) cross entropy is given by,

$$T_{block}^{(i,m)} \approx \sum_{k=1}^{N} T^{(i,m)}(u_k). \tag{7}$$

The experience with computer simulations has shown that once convergence is achieved, $T_{block}^{(i,m)}$ drops by a factor of 10^{-2} to 10^{-4} compared with its initial value $T_{block}^{(1,m)}$. Thus it is reasonable to use it as a stopping rule for iterative decoding:

$$T_{block}^{(i,m)} < 10^{-3} T_{block}^{(1,m)}. \tag{8}$$

Note that Eq.(8) essentially compares the averaged convergence status of all bits and does not reflect the convergence status of individual bits. Another criterion based on the convergence status of individual bits is given by [7],

$$T^{(i,m)}(u_k) < 10^{-3} T^{(1,m)}(u_k). \tag{9}$$

Bits satisfying Eq.(9) are classified as converged and the others as non-converged. According to Eq.(9) the information estimates at the output can be rewritten as two parts, that is the non-converged term $\hat{u}_\Omega$ and the converged term $\hat{u}_{\bar{\Omega}}$ where,

$$\Omega \equiv \{j \mid T^{(i,m)}(u_j) \ge 10^{-3} T^{(1,m)}(u_j),\quad 1 \le j \le N\} \tag{10}$$

and

$$\bar{\Omega} \equiv \{1 \le j \le N\} - \Omega. \tag{11}$$

4. PARTIAL CROSS-ENTROPY STOPPING CRITERION WITH ADAPTIVE NORMALIZATION

However it has been found that the output information of the SISO turbo decoder doesn't correctly predict the a-posteriori probability of the hard decision for bad channels [5],[8]. In fact, the output information is too optimistic and thus, a correction or scaling for the output LLRs is necessary to improve the performance. In Fig.(2), the evolution of the converged and non-converged parts of the cross-entropy is shown for one SISO decoder under bad channel conditions (low E_b/N_0 values).

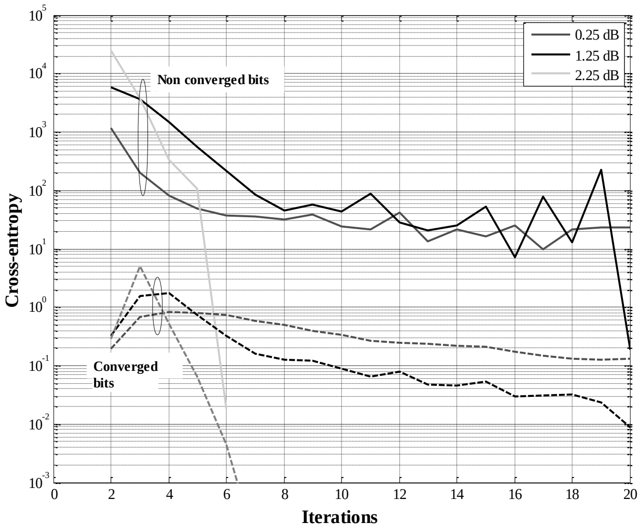

Fig.2. Cross-entropy evolution with iterations of the converged and non-converged bits

This illustration provides a useful insight into the behavior of iterative decoding. It can be seen from the figure that the cross-entropy magnitude of the converged part is negligible in comparison with the non-converged part. In addition, the slope of this latter is decreasing (monotonic).

Thus, the convergence rate of iterative decoding is based on the non-converged bits, which are determinative in an early stopping criterion and a partial scaling is more efficient.

It is known that once the decoding has converged, the cross-entropy of the converged bits is assumed negligible compared with the cross-entropy of the non-converged bits. Thus the frame cross-entropy stopping rule in Eq.(8) is mainly determined by the individual cross-entropy of the non-converged bits that is the sum in Eq.(7). As a result, the too optimistic LLR of the non-converged bits is normalized by an adaptive factor to compensate the output of SISO decoders. So at each iteration, it is suggested to multiply the extrinsic information at the outputs of SISO decoders by a normalization factors denoted as $C_{NSD}^{(i,m)}$:

$$\tilde{L}_a^{(i,1)}\left(\hat{u}_j\right) = C_{NSD}^{(i-1,1)}\Pi^{-1}\left[\tilde{L}_e^{(i-1,2)}\left(\hat{u}_j\right)\right] \quad (12)$$

$$\tilde{L}_a^{(i,2)}\left(\hat{u}_j\right) = C_{NSD}^{(i,2)}\Pi\left[\tilde{L}_e^{(i,1)}\left(\hat{u}_j\right)\right] \quad (13)$$

where, $j \in \Omega$ and $\tilde{L}_e^{(i,1)}\left(\hat{u}_j\right)$, $\tilde{L}_e^{(i,2)}\left(\hat{u}_j\right)$, $\tilde{L}_a^{(i,1)}\left(\hat{u}_j\right)$, $\tilde{L}_a^{(i,2)}\left(\hat{u}_j\right)$ are the normalized versions of $L_e^{(i,1)}\left(\hat{u}_j\right)$, $L_e^{(i,2)}\left(\hat{u}_j\right)$, $L_a^{(i,1)}\left(\hat{u}_j\right)$, $L_a^{(i,2)}\left(\hat{u}_j\right)$ respectively, for the non-converged bits only.

The normalization factors are calculated from the whole frame using the number of sign difference *NSD* between the extrinsic output and the a priori input information sequences at the i^{th} iteration. They are given in [5] by,

$$C_{NSD}^{(i,1)} = 1 - \frac{NSD^{(i-1,2)}(\hat{u})}{N} \quad (14)$$

$$C_{NSD}^{(i,2)} = 1 - \frac{NSD^{(i-1,1)}(\hat{u})}{N}. \quad (15)$$

However, the stopping rule is modified to focus on the non-converged bits only. These modifications will enhance the stopping criterion to track not only the solvable frames in high *SNR* conditions, but also to early terminate the iterations for the unsolvable frames in low *SNR* conditions, where the decoder fails to decode and further iterations are unnecessary.

The proposed cross-entropy stopping criterion based on the non-converged bit-selection is given by,

$$T_{\Omega}^{(i,m)} < 10^{-3} T_{block}^{(1,m)} \quad (16)$$

where,

$$T_{\Omega}^{(i,m)} = \sum\nolimits_{j\in\Omega} T^{(i,m)}\left(u_j\right). \quad (17)$$

5. SIMULATION RESULTS AND DISCUSSIONS

In this section, the simulation results for a BPSK (inter-symbol interference free) transmission over an AWGN channel, with a signal-to-noise ratio expressed by E_b/N_0 and ranging from -0.25 to 3.5dB, are presented. The simulated turbo decoder consists of two RSC codes with generators $G_1 = 7$ and $G_2 = 5$ in octal, and an overall code rate of 1/3. A random interleaver is used, and the maximum number of iterations the turbo decoder can perform is limited to *MaxItr* = 10 as no significant improvement in performance is obtained with further iterations. A block of $N = 1568$ bits is considered and $Bk = 1000$ blocks are transmitted for each value of E_b/N_0.

The performance of the proposed criterion is evaluated in terms of the bit error rate *BER* and the average number of iterations as a function of E_b/N_0. To investigate the performance of the proposed stopping criterion in *HARQ* systems, Type-I *HARQ* system is used with a maximum number of retransmissions per codeword fixed to *MaxRq* = 4. The performance of the *HARQ* system is evaluated in terms of throughput. Recall that the throughput, in a channel coding system, is defined as the average number of accepted frames in the time it takes to transmit one frame.

$$Throughput = \frac{1}{Bk}\sum_{i=1}^{Bk}\frac{1}{1+Rq(i)}. \quad (18)$$

We propose a new measure of the system throughput, $Throughput_{PCE}$ which takes in consideration the average number of iterations to decode the received frame, so it is calculated as,

$$Throughput_{PCE} = \frac{1}{Bk}\sum_{i=1}^{Bk}\frac{MaxItr}{\sum_{j=1}^{MaxRq+1} Itr(i,j)}. \quad (19)$$

$Itr(i,j)$ is the iteration index of the frame i at the transmission index j. The Eq.(19) gives a joint measure of the saved time from stopping criteria and the retransmission wasted time, which is more accurate and shows the real performance of the system.

The number of sign difference *NSD* between the extrinsic output and the a priori input information bits, i.e. the bits the two decoders disagree on, is used as a retransmission criterion (requesting retransmission), when the iterative process is early terminated by the stopping criterion.

The Fig.3 shows the *BER* plotted versus E_b/N_0 for a log-MAP decoder with the number of iterations fixed to ten, in normal decoding and with partial adaptive normalization. As can be seen, the *BER* of the SISO decoder was improved by means of the partial adaptive normalization.

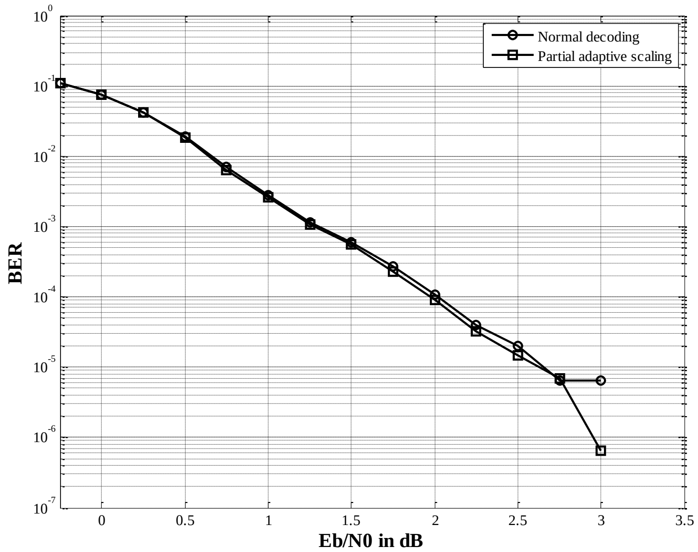

Fig.3. *BER* versus E_b/N_0 for a turbo decoder with partial adaptive normalization

To observe the achievable empirical bounds, the simulations for several approaches discussed previously are performed and the conventional *GENIE* curve is also provided.

The Fig.4 and Fig.5 illustrate the simulation results of the average iteration number and *BER* versus E_b/N_0 for *HDA*, *NSD*, *HLLR*, *CE*, and the proposed criterion *PCE* with two combined thresholds, 10^{-3} for the selection of non-converged bits and 10^{-2} for stopping the iterative decoding. As shown from Fig.4 the increasing order for the amount of saved iterations in low *SNR* conditions is: *PCE>HAD>HLLR>CE>NSD*. It is clear that the proposed criterion *PCE* exhibits an advantage over the other criteria in the low *SNR* region, that is the turbo decoder performs with very few iterations without significant loss of *BER* performance as can be seen in Fig.5. On the other hand, in most of the compared criteria the turbo decoder performs with maximal iterations in low *SNR* situations, which is considered as useless energy expenditure and an unnecessary delay, before requesting a retransmission. Except for the proposed method where it reduces considerably the required number of iterations.

In the region of high *SNR*, Fig.4 shows the following ordering of the simulated criteria, from the most efficient to the less, in terms of the required number of iterations: *HLLR>PCE>HDA>CE>NSD*. As in low *SNR* region, the *PCE* criterion exhibits good results in comparison with the simulated criteria. It is to be noted that the difference of the required number of iterations, between the compared criteria in this region, is a fraction of the iteration, which can't be considered as an important advantage.

Moreover in Fig.5, performances of the simulated criteria are close to each other, either in low or high *SNR* situations. Obviously, the computational load of the partial cross-entropy based stopping criterion is significantly lower than the computational complexity of the unnecessary extra iterations in a full iterative decoding.

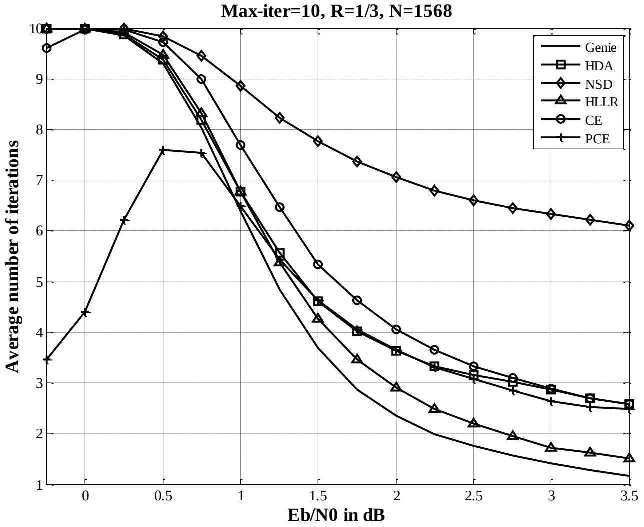

Fig.4. Comparison of the average number of iterations using various criteria as a function of E_b/N_0

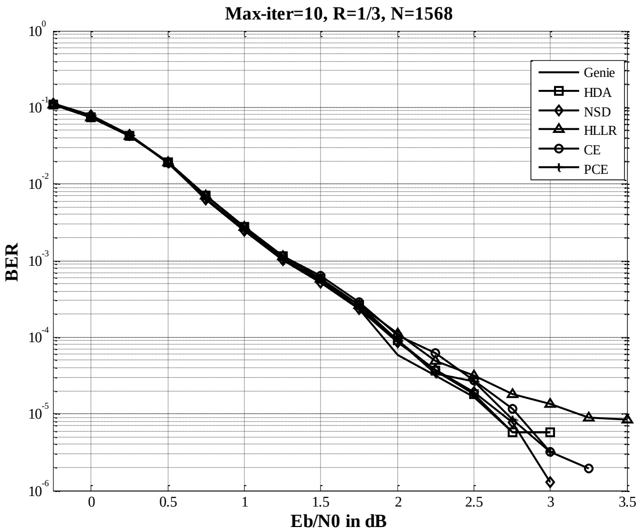

Fig.5. BER performance for various criteria

In the following simulations we consider a fixed E_b/N_0 for all retransmissions, which is equal to the E_b/N_0 value at the first transmission.

The Fig.6 shows the Type-I *HARQ* scheme system throughput calculated by Eq.(18) for the proposed stopping criterion, using different threshold values of the *NSD*. It can be seen that the throughput is maintained acceptable as the *SNR* decreases to 0.5dB which covers a wide range of the simulated interval and falls rapidly after this value (bad channel conditions). In the case of NSD_{Th} = 0 the results are out of the accepted range because a perfect matching between the two extrinsic information of the constituent decoders is difficult, and the maximum number of retransmissions is always reached even if the codeword is error-free.

By considering the benefit of early terminating a successful iterative decoding in less than the default required time, to start decoding a new received frame, within the remaining portion of that time, a cumulative time portions, especially in good channel conditions, results in an impressive increase of the overall system throughput.

This is shown in Fig.7 by using Eq.(19) to take account of this property. As the *SNR* conditions are good, the *HARQ* system has the possibility to increase its throughput up to four times the nominal capacity. Even in very poor channel conditions the overall throughput is maintained more than half the nominal capacity.

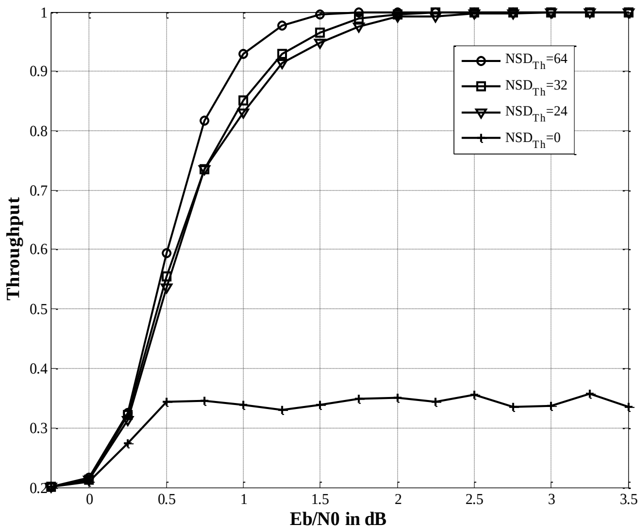

Fig.6. *HARQ* system's throughput using *PCE* criterion with different retransmission thresholds according to Eq.(18)

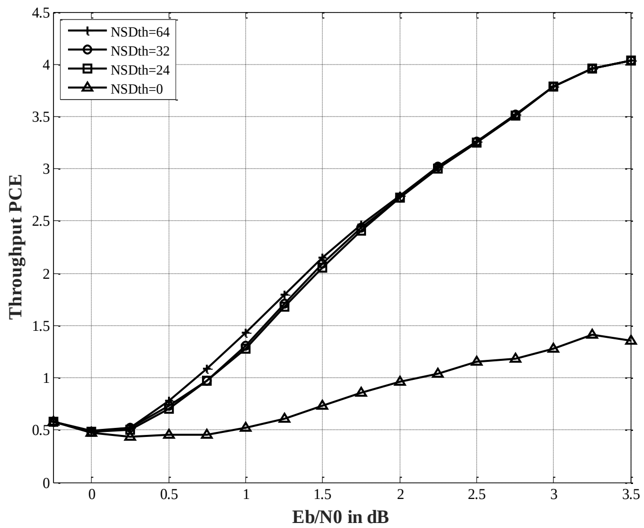

Fig.7. *HARQ* system's throughput using *PCE* criterion with different retransmission thresholds according to Eq.(19)

6. CONCLUSION

In this paper, a new stopping criterion namely *PCE* is proposed, this is based on bit convergence identification of iterative decoding via cross-entropy, and the log-MAP algorithm improvement. At first, bits are classified in terms of convergence status after the initial complete iteration. Afterwards, partial adaptive correction and partial cross-entropy stopping rule are applied on the non-converged bits.

The simulations show that the achieved average number of iterations of the proposed *PCE* criterion outperforms the other discussed criteria. It can stop in either high-SNR situation where the input information is highly reliable or in low-SNR situation where the decoder already has no ability to decode. Thus the overall complexity is reduced by decreasing the number of iterations, while the loss of *BER* performance is quite small and satisfies the system specification. The combined turbo decoding and Type-I *HARQ* allow the system to benefit from the reduced computational complexity and to improve the overall system throughput up to four times by an effective error detection and correction strategy.

Note that, the proposed criterion is not limited to the log-MAP algorithm and therefore perspective includes the possibility of generalizing to other iterative decoding algorithms within this new framework.

REFERENCES

[1] C. Berrou and A. Glavieux, "Near Optimum Error Correcting Coding and Decoding: Turbo-codes", *IEEE Transactions on Communications*, Vol. 44, No. 10, pp. 1261-1271, 1996.

[2] Fan-Min Li and An-Yeu Wu, "On the New Stopping Criteria of Iterative Turbo Decoding by Using Decoding Threshold", *IEEE Transactions on Signal Processing*, Vol. 55, No. 11, pp. 5506-5516, 2007.

[3] J. Hagenauer, E. Offer and L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes", *IEEE Transactions on Information Theory*, Vol. 42, No. 2, pp. 429-445, 1996.

[4] R.Y. Shao, Shu Lin and M.P.C. Fossorier, "Two Simple Stopping Criteria for Turbo Decoding", *IEEE Transactions on Communications*, Vol. 47, No. 8, pp. 1117-1120, 1999.

[5] Erl-Huei Lu, Yi-Nan Lin and Wei-Wen Hung, "Improvement of Turbo Decoding using Cross-Entropy", *Computer Communications*, Vol. 32, No. 6, pp. 1034-1038, 2009.

[6] Imed Amamra and Nadir Derouiche, "A Stopping Criteria for Turbo Decoding based on the LLR Histogram", *Proceedings of 16th IEEE Mediterranean Electrotechnical Conference*, pp. 699-702, 2012.

[7] J. Wu, Z. Wang and B. Vojcic, "Partial Iterative Decoding for Binary Turbo Codes via Cross-Entropy based Bit Selection", *IEEE Transactions on Communications*, Vol. 57, No. 11, pp. 3298-3306, 2009.

[8] Lajos L. Hanzo, T.H. Liew and B.L. Yeap, "*Turbo Coding, Turbo Equalisation and Space-Time Coding for Transmission Over Fading Channels*", Wiley, 2002.

[9] Beomkyu Shin, Sang-Hyo Kim, Jong-Seon No and Dong-Joon Shin, "New Stopping Criteria for Iterative Decoding of LDPC Codes in H-ARQ Systems", *International Journal of Communication Systems*, Vol. 26, No. 11, pp. 1475-1484, 2013.

[10] P. Haase and H. Rohling, "Truncated Turbo Decoding with Reduced Computational Complexity", *Wireless Personal Communications*, Vol. 20, No. 2, pp. 101-112, 2002.

[11] Q. Luo, R. Hoshyar-Jabal-Kandi and P. Sweeney, "Cross-Entropy-based Method to Analyse Iterative Decoding", *IET Communications*, Vol. 2, No. 1, pp. 113-120, 2008.

[12] Z. Ma, P. Fan, W.H. Mow and Q. Chen, "*A Joint Early Detection-early Stopping Scheme for Short-frame Turbo Decoding*", *AEU-International Journal of Electronics and Communications*, Vol. 65, No. 1, pp. 37-43, 2011.

[13] E. Uhlemann, "Adaptive Concatenated Coding for Wireless Real-time Communications", PhD Dissertation, Department of Computer Engineering, Chalmers University of Technology, 2004.