ADAPTIVE EARLY-STOPPING THRESHOLD FOR LTE TURBO DECODER

Rodolfo Torrea-Duran, David Novo, Claude Desset, Frederik Naessens, and Liesbet Van der Perre

Inter university Micro-Electronics Center (IMEC), Wireless Research Group
Kapeldreef 75, B-3001 Leuven, Belgium
phone: + (32) 16 82 20, email: Rodolfo.TorreaDuran@imec.be

ABSTRACT

Turbo coding has been adopted by 3GPP-LTE standard for error correction. However, decoding with a fixed number of iterations may lead to excessive processing, penalizing battery-operated terminals while fewer iterations are sometimes sufficient. Early-stopping criteria can stop the iterative process when a certain confidence threshold has been achieved. However, practical implementations use a fixed threshold. In this paper we propose an adaptive approach based on the block size and coding rate to select the best threshold for two state-of-the-art early-stopping techniques. We show gains up to 38% in average number of iterations with respect to a fixed-threshold approach and higher gains with respect to a fixed-iterations approach without degradation under the LTE performance constraints. Moreover, our approach does not require SNR knowledge.

1. INTRODUCTION

With the explosive growth of broadband mobile wireless demands, cellular standards such as Universal Mobile Telecommunications System (UMTS) need to evolve and provide higher data rate to remain competitive. As a result, the evolved version of UMTS called Long Term Evolution (LTE) and LTE Advanced target the daily increasing demand on mobile communications for the next years.

By adopting Orthogonal Frequency Division Multiplexing (OFDM), Multiple-Input Multiple-Output (MIMO) antenna schemes, and scalable frequency bandwidths, User Equipments (UE) are capable of downlink data rates of 326 Mbps in a 20 MHz bandwidth for a 4x4 antenna configuration [1]. However, these data rates could not be possible without an effective channel coding scheme like turbo coding, supported by LTE [2].

Introduced in 1993, turbo codes proved to have an error correction performance close to the Shannon limit [3]. Turbo encoding uses a parallel concatenation of codes separated by interleavers, while turbo decoding is based on alternately decoding each component code and passing soft information to the next decoding stage. However, the improvement in signal-to-noise ratio (SNR) becomes smaller with each iteration.

This iterative process is performed over a block of bits or code block. It yields a high level of complexity while the latency and energy consumption increase linearly with the number of iterations. The number of iterations for correct decoding strongly depends on channel characteristics. In some cases, successful decoding will never be reached even with infinite iterations. However, in most cases a few iterations are sufficient to provide a correct decoding.

Practical turbo decoders implement a fixed number of iterations for all the code blocks based on the worst case even if many code blocks could be successfully decoded with fewer iterations. If we could know the number of iterations that are sufficient to decode a code block or at least to reach a certain degree of confidence, or performance, unnecessary decoding operations would be avoided. Such techniques are called early-stopping criteria. Several of them have been presented in literature and they aim to stop the iteration process when a certain confidence threshold is achieved. The choice of this threshold results then in a trade-off between average number of iterations and performance.

In cellular standards such as 3GPP-LTE, the block size and coding rate of a code block are not static. Therefore, benefits can be obtained with a threshold adapting to the code block characteristics.

In this paper we propose an adaptive approach of reduced overhead based on the code block size and coding rate that selects the best threshold for two state-of-the-art early stopping techniques We compute the achievable gain in average number of iterations compared to a fixed-threshold and a fixed-iterations (classical) approach.

This paper is organized as follows. Section 2 shows the previous work and motivates the proposed approach. Section 3 describes our adaptive-threshold approach. Section 4 shows the results of our adaptive approach in a 3GPP-LTE system. Finally Section 5 draws the conclusions.

2. PREVIOUS WORK IN LITERATURE

An exhaustive classification of early-stopping criteria can be found in [4], dividing them into three groups: those based on soft decisions [5][6], hard decisions [7], and Cyclic Redundancy Check (CRC)[5].

In [8] several of the previous criteria are proposed for a UMTS turbo decoder. Many of them pose a considerable overhead due to their computational complexity and high memory requirements. The best criteria suited for implementation are the minimum Log-Likelihood Ratio (LLR), the Sign Difference Ratio (SDR), and the Sum-reliability criteria, offering simplicity, low-memory requirements, and good performance [8]. Although the last criterion offers some reduction in the number of iterations over all SNR values compared to a classical approach, it requires more iterations than the first two for high SNRs.

 ISSN 2076-1465

The SDR criterion consists in comparing the number of bits presenting a sign difference between a priori and extrinsic information. Decoding is stopped when this number is a fraction of the code block size or when we have reached the maximum number of iterations. It does not require storage from previous iterations.

The LLR criterion consists in stopping the decoding once the absolute value of all the output bit LLRs are above a threshold or the maximum number of iterations is reached. It only requires limited storage and almost no extra computation.

To our knowledge, these criteria have been scarcely assessed in a practical system. For example, in [9] some LLR thresholds were tested for an UMTS system using a fixed block size and coding rate. In [10] a two-level early-stopping algorithm for LTE was proposed, but it is based on the computationally-expensive CRC checksum and no soft/hard decision approach was considered.

A CRC stopping rule could achieve almost the same performance than the genie-solution. However it results in more computation than SDR or LLR [5]. Also, there are some implementation-related problems. For instance, the CRC depends on the input bits ordering, so it cannot be calculated on-the-fly, which introduces certain latency. In contrast, LLR and SDR criteria can be done on-the-fly, with very low storage and computation overhead [8]. LLR and SDR do not require memory storage of previous iterations. In fact, as soon as the number of bits with a sign difference (SDR) or any of the LLR values reach the chosen threshold, the iterations can be stopped. This can bring large benefits in terms of latency and power consumption.

Due to the low implementation complexity and memory requirements, we assess in this paper the performance of the LLR and the SDR early-stopping criteria for the LTE standard. Most of the papers on early-stopping criteria focus on comparing different methods and selecting a fixed threshold. Alternatively, in this paper we propose an adaptive early-stopping threshold for those two methods.

Our approach is compared with two references. The first one is the classical worst-case design with a predefined fixed number of iterations. The second one is a "genie-based" approach that assumes complete knowledge of the transmitted bits and stops the decoding in the minimum number of iterations required to successfully decode a code block. It provides a bound on the minimum number of iterations.

3. ADAPTIVE THRESHOLD APPROACH

For a given SNR, a strongly-coded block (lower coding rate) needs less iterations in average to reach the same performance achieved with more iterations over a weakly-coded block. In addition, large code blocks have more error correction potential than small ones because of the extra information available in a large code block. This can be seen in Figure 1 and Figure 2 for a "genie" implementation with 6 as maximum number of iterations. Furthermore, this reduction in the number of iterations could be quite substantial depending on the code block error rate (BLER) that we want to achieve (Figure 3). However, we cannot achieve the same

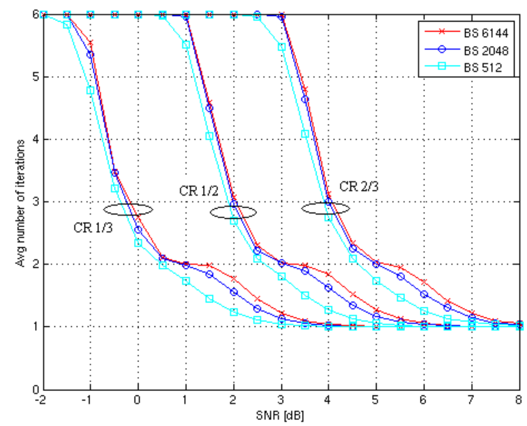

Figure 1 – Average number of turbo iterations for different block size (BS) and coding rates (CR) with genie early stopping.

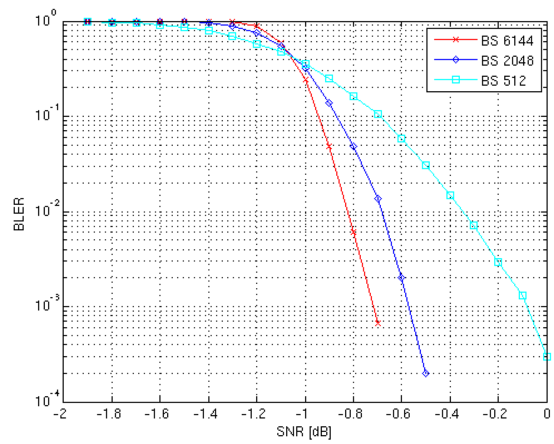

Figure 2 – BLER for different block sizes (BS) and coding rate 1/3 with genie early stopping.

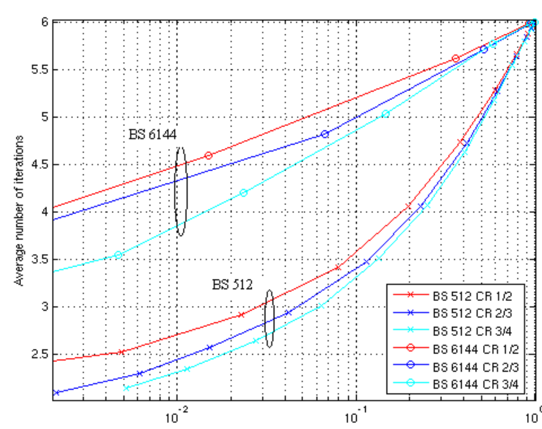

Figure 3 – Average number of turbo iterations as a function of BLER with genie early stopping.

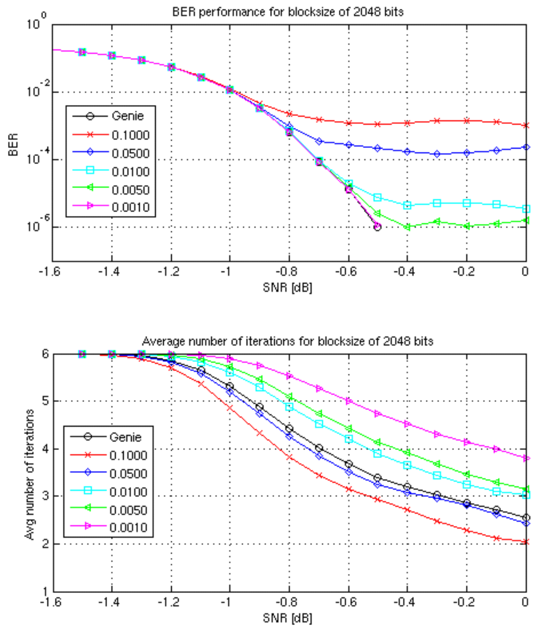

Figure 4 – SDR BER performance and average number of iterations, block size 2048 bits

average number of iterations than the "genie" approach without a penalty.

Any early-stopping criteria shows degradation from the "genie" solution at a certain bit error rate (BER). This happens when the criterion is not able to guarantee a higher degree of confidence, therefore the BER curve separates from the "genie" as can be seen in Figure 4 for SDR criterion. As the threshold becomes smaller, the degradation comes at a lower BER, yet more iterations are needed to satisfy this threshold. There are even some thresholds that can achieve less iterations in average than the "genie", but the price to pay in performance is very high. Following the system performance bounds, we define a region of no degradation from the optimal solution. This helps us to select the threshold that achieves the lowest amount of iterations while still fulfilling the system constraints. A practical implementation of such an adaptive-threshold approach is simple from a look-up table.

Again, depending on the block size and coding rate, a code block can reach a higher degree of confidence with fewer iterations. Hence, a fixed-threshold approach is not optimal with systems that use code blocks with variable characteristics. For example, in LTE standard, block sizes can vary from 40 to 6144 bits with different coding rates (15 possible channel quality indicator or CQI values) [2][11]. On one hand, the block size choice depends on the input bit sequence length. On the other hand, the code rate choice depends on the performance achieved by the decoder and on the number of bits needed for radio resource assignment. If the decoder achieved a good performance in terms of BLER, a higher modulation and coding rate scheme (link adaptation) could be used.

Also from Figure 4, we notice that the amount of iterations that we could save depends on the input SNR, a parameter difficult to estimate precisely in a real implementation. Hence, we have to select a relevant SNR working region. The LTE standard specifies a minimum performance. If it increases, the link adaptation mechanism is likely to switch to a faster mode to increase capacity. We select then a range of 3dB as working region starting from the point of minimum performance. This permits us to quantify the gains of the selected threshold, while avoiding the knowledge of the SNR.

4. RESULTS

Simulations were performed using a standard-compliant turbo decoder in an AWGN channel with 6 iterations as a maximum. Three different block sizes are considered corresponding to a maximum, an average, and a small code block: 6144, 2048, and 512 bits, and 5 different coding rates: 1/3, 1/2, 2/3, 3/4, and 7/8. With each possible combination of block size and coding rate, several thresholds were

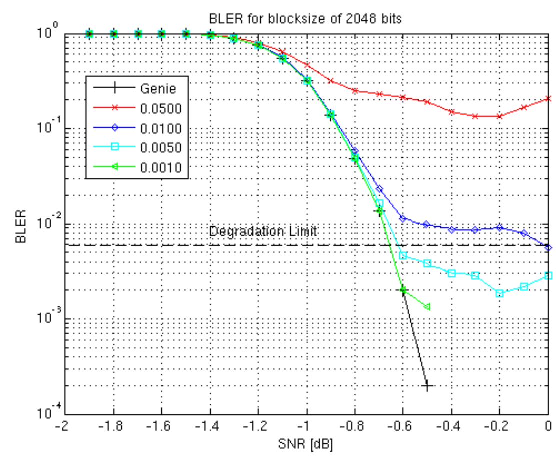

Figure 5 – SDR BLER performance, block size 2048 bits, 1/3

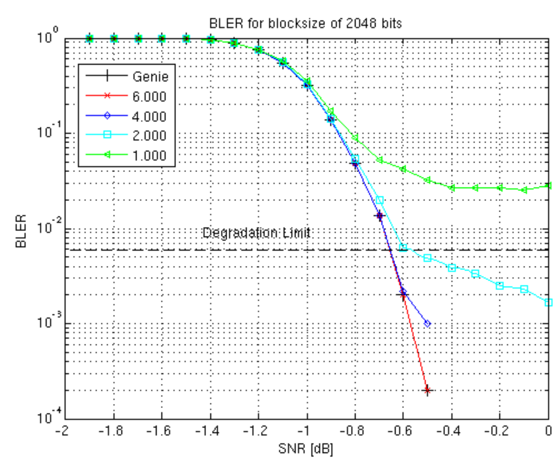

Figure 6 – LLR BLER performance, block size 2048 bits, 1/3 coding rate.

tested for the two considered criteria. The tested thresholds for SDR are 0.01, 0.005, 0.0025, 0.001, 0.00075, 0.0005, 0.00025, 0.0001; and for LLR, 6, 4, 3, 2, 1, 0.5, 0.1, 0.05, 0.01. The SDR threshold represents the fraction of code block bits changing over one iteration, while the LLR threshold is the value above which all output LLRs should be.

In LTE, the medium access control (MAC) layer operates on the unit of a transport block (TB). As such, retransmissions are based on the correct decoding of the entire TB, which could be too large to be processed as a whole by the encoder. For practical purposes, TBs are divided into a number of code blocks that are independently encoded by a 8-state, 1/3 mother code rate turbo encoder. LTE allows the use of a different block size and coding rate for groups of code blocks belonging to a single TB depending on the channel feedback reported by the terminal [11].

Still, a suitable number of bits need to be generated for radio resource assignment. This is done by puncturing or repeating the bits of the mother code rate to generate a desired number of bits. This is equivalent to increasing or decreasing the code rate.

Based on the received signal quality, the UE feeds back for which scheme it can receive a TB with a transport block error (TBLER) probability of 0.1[11]. However, one erroneous code block is enough to retransmit the whole TB. Since many code blocks form one TB, the BLER should be much lower than the TBLER.

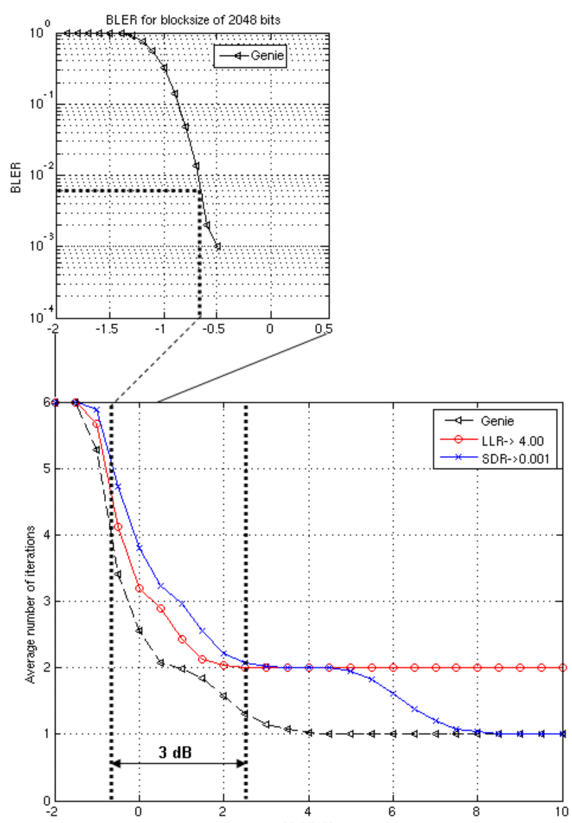

Figure 7 – Average number of iterations of SDR and LLR, block size 2048 bits and 1/3 coding rate.

Based on a TBLER of 0.1, we derive a maximum allowed BLER. For a 20 MHz bandwidth, the maximum TB size can be of roughly more than 100,000 bits corresponding to 17 code blocks per TB [2], so we can allow 1 code block out of 170 (BLER ~ 0.006) to be incorrect in the worst case. In order to avoid degradation up to a TBLER of 0.1, the selected threshold should offer the same performance of the optimal solution for a target BLER of 0.006 with the smallest amount of iterations in average.

The BLER performance of some of the considered thresholds can be seen in Figure 5 and Figure 6 for SDR and LLR criteria, respectively. The target BLER below which we can allow degradation is indicated with a dotted line (degradation limit). For SDR, the smaller (tighter) the threshold, the smaller the BLER that results in performance degradation; while for LLR, a higher value represents a tighter threshold. For SDR only threshold 0.001 satisfies the degradation constraint, and for LLR both thresholds 6 and 4 do, but 4 achieves fewer iterations in average.

The lower graph of Figure 7 shows the average number of iterations for the selected SDR and LLR thresholds satisfying the target BLER with the fewest average iterations for a 2048 block size with 1/3 coding rate. The criterion offering the least amount of average iterations depends on the input SNR. In this case, LLR is better than SDR for low values of SNR, and SDR is better for high values. Both solutions are less than one iteration away from the optimal solution for any SNR value.

Therefore, our adaptive approach cannot be based just on selecting a certain threshold and criteria. We use then a 3dB working region as explained in Section 2 starting from a BLER of 0.006 (upper graph of Figure 7). In case of working at a higher SNR, the decoder can provide even further savings in average number of iterations.

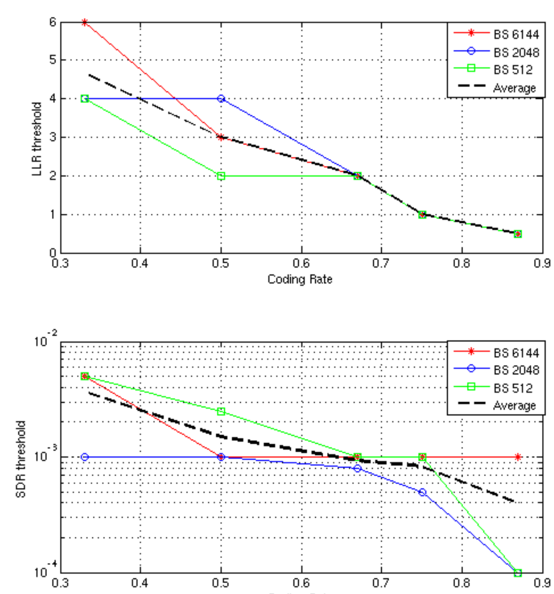

Figure 8 – Threshold selection for LLR and SDR according to block size (BS) and coding rate (CR).

The selected thresholds respecting the LTE performance bounds for both LLR and SDR criteria are presented in Figure 8. With respect to different block sizes, there is not an important tendency, however, we observe that the tightness of the threshold increases with the coding rate for SDR, while it decreases for LLR.

This could be explained by the coding process. The punctured parity bits at the encoding side result in more sign changes between a priori and extrinsic information at the decoding side. Therefore SDR needs tighter thresholds for higher coding rates to obtain the target BLER performance. However, in the case of LLR, looser thresholds are needed for higher coding rates because without parity information a small increase in the LLR values guarantees a good confidence level.

For each of the selected thresholds, we compute the average number of iterations of the adaptive and fixed threshold approaches and the “genie” implementation in the selected SNR working region (3dB starting from the BLER bound). The fixed threshold selected for both criteria is a threshold that offers no degradation at BLER 0.006 for all the block sizes and coding rates.

In the best case, with an adaptive threshold in both criteria we can save up to 2.3 iterations in average iterations (38%) for LLR, and up to 1 iteration (16%) for SDR with respect to a fixed-threshold.

Compared to a classical approach our gains are higher than 38% for most cases without the need of knowing the input SNR. LLR adaptive-threshold proves to save more iterations than SDR adaptive-threshold for most coding rates.

The results for a block size of 2048 are presented in Figure 9 showing the average iterations obtained by the adaptive LLR and SDR thresholds compared with a fixed and a “genie” approach.

By combining both adaptive-threshold approaches, we can achieve high gains for every coding rate. This approach is at most one iteration away from the “genie” solution with no performance degradation under the system constraints at any coding rate and with a small implementation overhead. However, already an adaptive-threshold approach for a single criterion could bring substantial gains compared to a fixed-threshold approach.

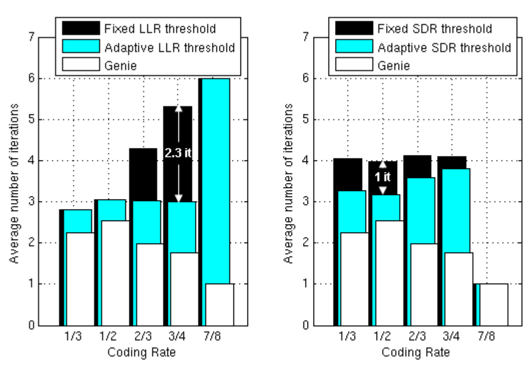

Figure 9 – Gain in average number of iterations, block size 2048

5. CONCLUSIONS

In this paper, we propose an adaptive approach to select the best confidence threshold between SDR and LLR early-stopping criteria based on the code block size and the coding rate. Our solution does not need knowledge of the input SNR of the turbo decoder. Also, it is less than one iteration away from the optimal solution with no degradation under the LTE minimum performance constraints.

We are able to achieve gains up to 38% for LLR and 16% for SDR with respect to a fixed-threshold approach and higher than 38% compared to a no-early-stopping solution. By combining both criteria we could achieve high gains for every coding rate and block size.

Our analysis also shows that already an adaptive-threshold approach for a single criterion achieves substantial gains. In this case, LLR adaptive-threshold saves more iterations than SDR for most coding rates and block sizes.

A future study involves the simulation results for frequency selective fading channels using different modulation schemes and multiple antennas.

REFERENCES

[1] 3GPP TS 25.912, “Feasibility study for evolved Universal Terrestrial Radio Access (UTRA) and Universal Terrestrial Radio Access Network (UTRAN)”, v9.0.0, September 2009.

[2] 3GPP TS 36.212, “Multiplexing and Channel Coding”, v9.0.0, December, 2009.

[3] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes”, *Proceedings of IEEE International Conference on Communications (ICC)*, (Geneve, Switzerland), May 1993.

[4] A. Matache, S. Dolinar, and F. Pollara. “Stopping Rules for Turbo Decoders,” *JPL TMO Progress Report*, no. 42-142, pp. 1-22, Aug. 2000.

[5] Y. Wu, B.D.Woerner, and W.J.Ebel, “A Simple Stopping Criterion for Turbo Decoding”, *IEEE Communications Letters*, vol. 40, pp. 258-260, Aug. 2000.

[6] J. Hagenauer, E.Offer, and L. Papke, “Iterative decoding of binary block and convolutional codes”, *IEEE Trans. Inform. Theory*, vol. 42, pp. 429-445, Mar. 1996.

[7] R.Y.Shao, S.Lin, and M.P.C.Fossorier, “Two simple stopping criteria for turbo decoding”, *IEEE Trans.Commun.*, vol. 47, pp. 1117-1120, Aug. 1999.

[8] F. Gilbert, F. Kienle, and N. When. “Low Complexity Stopping Criteria for UMTS Turbo-Decoders,” *Vehicular Technology Conference*, vol. 4, pp. 2376- 2380, Apr. 2003.

[9] M. C. Valenti and J. Sun, "The UMTS Turbo Code and an Efficient Decoder Implementation Suitable for Software-Defined Radios", *International Journal of Wireless Information Networks*, Oct. 2001.

[10] J.F.Cheng, “Two-Level Early Stopping Algorithm for LTE Turbo Decoding”, *IEEE IEEE 68th Vehicular Technology Conference.*, Sep. 2008.

[11] 3GPP TS 36.213, “Physical Layer Procedures”, v9.0.0, December, 2009.