ADAPTIVE ITERATION SCHEME OF TURBO CODE USING HYSTERESIS CONTROL

Chih-Hao WU*, Kenichi ITO**, Yung-Liang HUANG*, Takuro SATO***

(Received October 29, 2004)

Turbo code, because of its remarkable coding performance, will be popular for the next generation wireless communication systems. This paper assesses the performance of an adaptive iteration algorithm for turbo decoding in AWGN. Numerical results are presented to demonstrate the feasibility of the algorithm..

Key words: Turbo decoding, Adaptive iterations, Hysteresis control

1.INTRODUCTION

Turbo coding [1] [2] architectures lie at the heart of all third-generation (3G) wireless standards, including UMTS [3] [4] and CDMA2000 [5]. These types of coding are called for because they allow systems to meet the tough bit-error-rate (BER) requirement and low signal-to-noise ratios (SNR) placed on emerging 3G designs.

The optimum decoding of turbo codes is the maximum likelihood-decoding algorithm applied to the turbo code trellis structure [6]. However, due to the interleaver embedded in the turbo encoder, the turbo code trellis will have an extremely large number of states. This fact makes the maximum likelihood decoding process almost impossible to implement in practice for large interleaver sizes. A more practical approach is an iterative decoding approach by which the maximum likelihood decoding algorithm is applied to the elementary convolutional /block codes of the turbo code. This iterative process raises the question of how close its performance is to the performance of the optimum decoding process. In [1], it was shown that turbo codes are near–optimal codes. Simulation results based on an iterative decoding approach were within 0.7 dB of the sphere–packing lower bound for various code rates and an information block size from 100 to 10,000 bits. Therefore, this iterative technique is a very efficient way to decode turbo codes and to achieve performance close to the theoretical limits.

However, calling for turbo coding architectures and actually implementing them are worlds apart. Turbo codes deliver excellent BER and SNR performance. But these achievements come at the cost of intense computing requirements. At a high data rate, the turbo decoder alone may consume more power than the rest of the base band transceiver [7] [8].

* Liteon Technology Wireless Research and Design Department

** Department of Information and Electronics Engineering, Associate Professor

*** Waseda University, Professor

In the last decade, both mobile communications and multimedia communications have experienced unequaled rapid growth and commercial success. Naturally, the great albeit separate successes in these areas fuel the old vision of ubiquitous multimedia communication enabling the user to communicate from anywhere at any time, transmitting and receiving any type of data. The convergence of mobile and multimedia venues is now underway. In the future, video and image transmission will become very important for cellular systems and wireless LAN systems. The wireless environment is in general a highly volatile one, with bit error rates potentially varying over orders of magnitude even within a single session as a user moves about or environmental conditions change. Because of this, it is worthwhile and in fact extremely important to consider a system that is adaptive to environmental conditions. This algorithm is suitable for video and image transmission, because the quality is not dependent on the environment or the distance between the base station (Access point) and the terminal.

Our algorithm therefore has an inherent characteristic that it maintains a constant received signal quality. In addition to its lower power consumption, the proposed algorithm has the advantage of constant quality suitable for image transmission.

2. ADAPTIVE ITERATIONS ALGORITHM

The adaptive iteration algorithm is shown in Figure 1. Consider for example the case in which the present iteration states $State_l$. The E_b/N_0 should meet as:

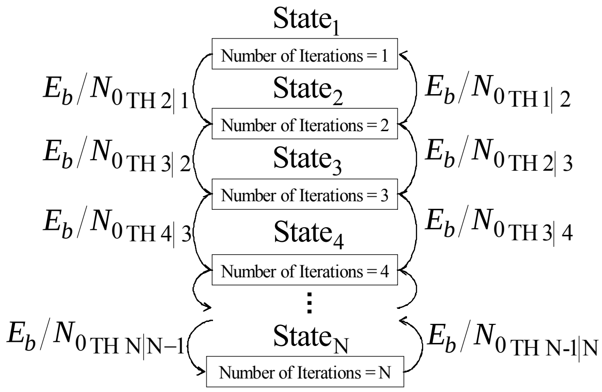
$$E_b/N_{0TH\ l+1|l} < E_b/N_0 < E_b/N_{0TH\ l-1|l}$$

$E_b/N_{0TH\ l-1|l}$ represents the E_b/N_0 threshold for iteration state transits from $State_l$ to $State_{l-1}$ .Once the next data slot signal has been received, we calculate the E_b/N_0 of the received slot, as follows:

If $E_b/N_0 < E_b/N_{0\ TH\ l+1|l}$ then $State_l \rightarrow State_{l+1}$

If $E_b/N_0 > E_b/N_{0\ TH\ l-1|l}$ then $State_l \rightarrow State_{l-1}$.

In the traditional algorithm, the threshold E_b/N_{0TH} is a constant value and $E_b/N_{0\ TH\ l|l+1}$ is equal to $E_b/N_{0\ TH\ l+1|l}$. In this paper, we integrate hysteresis Δ into the constant E_b/N_{0TH} to reduce the transitions of the iteration state. Figure 2 shows the Transfer Iteration State with hysteresis.



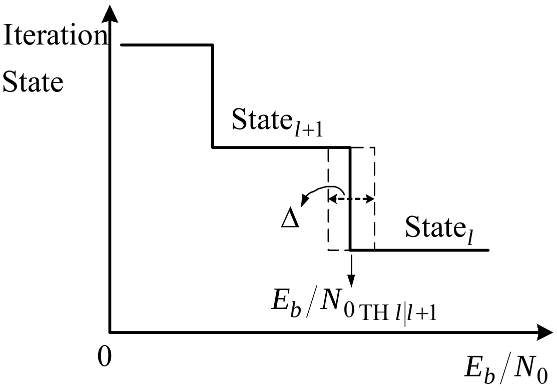


Fig. 1 Adaptive Iteration Algorithm.

Fig. 2 Transfer Iteration State from $State_{l+1}$ to $State_l$.

3. BER PERFORMANCE EVALUATION

3.1 System model

The received E_b/N_0 is as modeled as follows:

$$x(t) = x + n \qquad (1)$$

where n represents E_b/N_0 estimation error and is modeled as zero mean-independent with Gaussian process with variance σ^2.

Figure 3 shows the BER performance for Turbo decoding in AWGN channel with no E_b/N_0 estimation error. The conditioned BER with different iteration numbers is shown in Table 1. Table 1 shows the minimum E_b/N_0 to achieve conditioned BER with fixed iteration number. For example, in condition $BER = 10^{-3}$, to achieve $BER < 10^{-3}$, we decide the $E_b/N_{0TH\,3|4} = 1.815$. If $E_b/N_{0TH\,3|4} > 1.815$, then the number of iterations changes from 4 to 3. As above, $E_b/N_{0TH\,2|3} = 2.3$, $E_b/N_{0TH\,1|2} = 3.7$. We assume $A_4 = 1.72$, then

$$E_b/N_{0TH\,3|4} = 1.815 = \frac{A_4 + A_3}{2} \Rightarrow A_3 = 1.91$$

$$E_b/N_{0TH\,2|3} = 2.3 = \frac{A_3 + A_2}{2} \Rightarrow A_2 = 2.69$$

$$E_b/N_{0TH\,1|2} = 3.7 = \frac{A_2 + A_1}{2} \Rightarrow A_1 = 4.71 \qquad (2).$$

The Δ shown in simulation is assumed as:

$$\Delta = hysteresis \;*\, (A_l - A_{l+1}) \qquad (3)$$

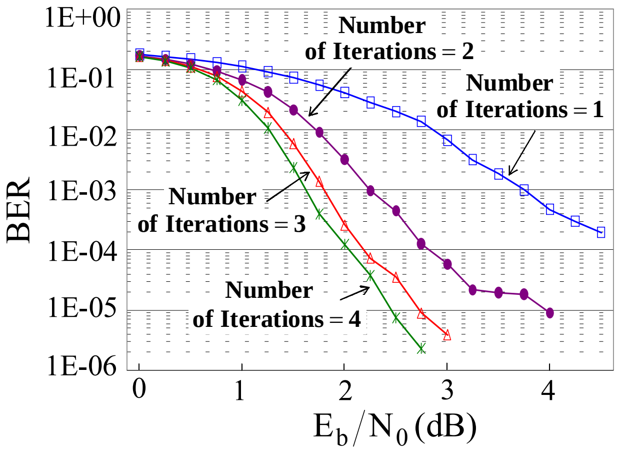


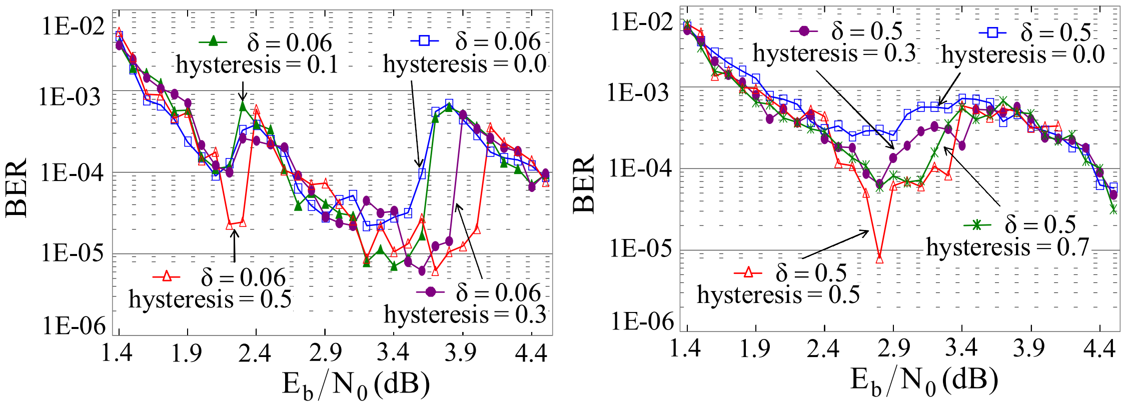
Fig. 3 Turbo decoding in AWGN channel, interleaver number=760, coding rate =1/3.

Table 1: Number of iterations in condition BER

$E_b/N_0\,(dB)$	Number of iterations			
BER	N=1	N=2	N=3	N=4
0.01	2.8	1.75	1.375	1.25
0.001	3.7	2.3	1.815	1.625

3.2 Simulation Results

Figures 4 shows that the adaptive iterations result in condition $BER = 10^{-3}$ with different E_b/N_0 estimation errors and hystereses. Based on the simulate results, we find that lower E_b/N_0 estimation error achieves better BER. The higher hysteresis will achieve better BER performance. As the E_b/N_0 estimation error increases, the performance of BER degrades. In Figure 4(a), there are two BER notches, one between 1.9 to 2.4 and one between 3.4 to 3.9, because there are different iterations of the state transition threshold $E_b/N_{0TH\,2|3} = 2.3$, $E_b/N_{0TH\,1|2} = 3.7$ in these respective areas. As E_b/N_0 estimation error increases, as shown in Figures 4(b), the BER curves become more flat than that shown in Figure 4(a). The notch phenomenon is not obvious, because the iteration stat is easy to transfer from one to the other with high E_b/N_0 measurement error.



(a) E_b/N_0 estimation error σ=0.06 (b) E_b/N_0 estimation error σ=0.5

Fig. 4 Adaptive iterations results.

4. ITERATION STATE TRANSITION EVALUATION

In order to evaluate the number of transition under dynamic variation of E_b/N_0 caused by change of communication environment, we investigated the effect of hysteresis under AWGN channel.

4.1 System model

We assumed that the E_b/N_0 value changed linearly from A_{l+1} to A_l due to change in the communication environment; the change rate of the E_b/N_0 value was constant. The E_b/N_0 value $x(k)$ is modeled as follows:

$$x(k) = A_{l+1} + k \cdot \alpha + n \qquad (4),$$

where k denotes a sample number, α represents the change rate, and n represents the E_b/N_0 estimation error and is modeled as zero mean-independent with Gaussian process with variance σ^2 . The term $k \cdot \alpha$ in the equation (4) means a variation in quantity of E_b/N_0 from A_{l+1}, measured at interval k .

Let $P_t(k)$ denote the probability that there is a transition at interval k . $P_{l+1|l}(k)$ denotes the probability of transition from A_l to A_{l+1}, and that of vice versa for the inverse transition is indicated by $P_{l|l+1}(k)$. Then, if $P_{Al}(k)$ and $P_{Al}(k)$ denote the probability that the present iteration state is characterized by $State_l$ and $State_{l+1}$ respectively, the following recursive relations hold:

$$\begin{aligned} Pt(k) &= P_{Al}(k-1)P_{l+1|l}(k) + P_{Al+1}(k-1)P_{l|l+1}(k) \\ P_{Al}(k) &= P_{Al}(k-1)(1-P_{l+1|l}(k)) + P_{Al+1}(k-1)P_{l|l+1}(k) \\ P_{Al+1}(k) &= P_{Al}(k-1)P_{l+1|l}(k) + P_{Al+1}(k-1)(1-P_{l|l+1}(k)) \end{aligned} \qquad (5),$$

k =1, …, $\dfrac{A_{l+1}}{\alpha}$, $P_{Al}(0)$ =1, and $P_{Al+1}(0)$ =0 as initial values.

The algorithm considered performs a transition to the adjacent state if the following condition is met: the measured E_b/N_0 falls below $\Delta_{l+1|l} = \dfrac{A_l + A_{l+1} - \Delta}{2}$ or exceeds $\Delta_{l|l+1} = \dfrac{A_l + A_{l+1} + \Delta}{2}$. $P_{l+1|l}(k)$ And $P_{l|l+1}(k)$ can be evaluated by

$$P_{l+1|l}(k) = P\{State_{l+1}(k) \,|\, State_l(k-1)\} \cong P\{x(k) < \Delta_{l+1|l} \,|\, x(k-1) > \Delta_{l+1|l}\} \qquad (6),$$

$$P_{l|l+1}(k) = P\{State_l(k) \,|\, State_{l+1}(k-1)\} \cong P\{x(k) > \Delta_{l|l+1} \,|\, x(k-1) < \Delta_{l|l+1}\} \qquad (7).$$

The following criteria may be used to assess the performance of the transition algorithm:

1) Average number of transitions $N_t = \sum_k P_t(k)$ (9)

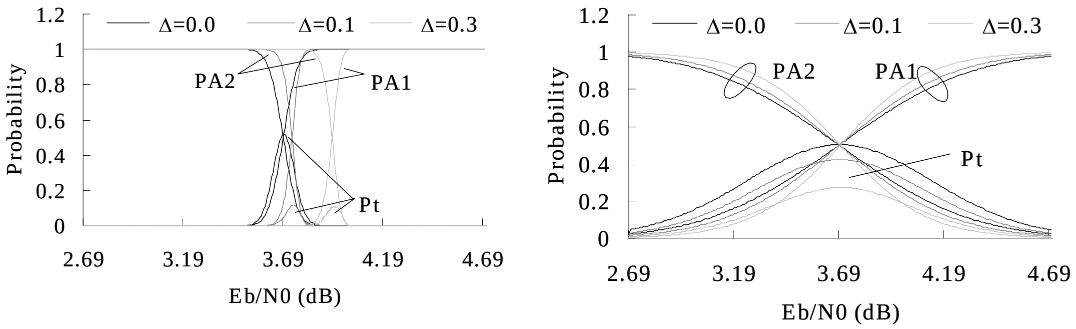
2) Average transition E_b/N_0 $E_b/N_{0\,t} = \sum_k k \cdot P_t(k)/N_t$ (10).

4.2 Numerical Results

For numerical evaluation, A_1, A_2, A_3, and A_4 are assumed to be 3.7, 2.3, 1.815, and 1.625 dB, respectively, on condition that $BER = 10^{-3}$. The inclination α is assumed to be 0.01.

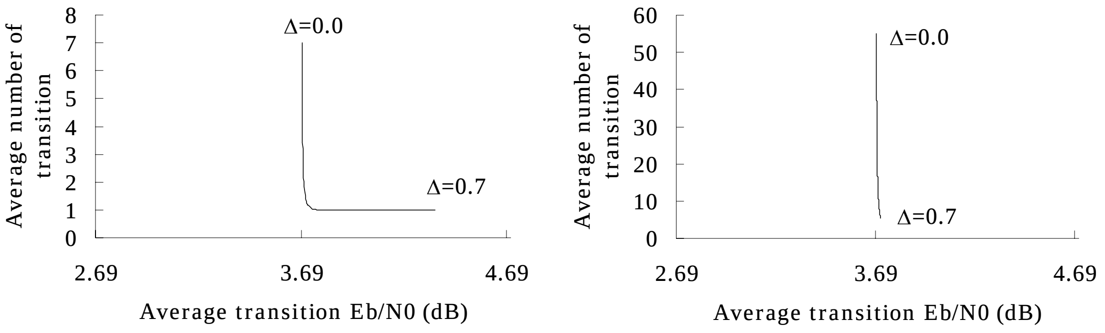
Figure 5 shows the probability of assignment to $State_1$ and $State_2$ and the probability of transition as the threshold level Δ is increased from 0 to 0.3 dB when the standard deviation of the E_b/N_0 estimation error is σ =0.06 and 0.5dB. From figure 5(a), in the case that E_b/N_0 estimation error is relatively small, as the threshold level Δ increases, the intersections of the probabilities $P_{A1}(k)$ and $P_{A2}(k)$ shift away from the state boundary, resulting in a magnitude reduction as well as a right shift from the center in $P_t(k)$. When E_b/N_0 estimation error is relatively large, transition probability $P_t(k)$ is even wider, and even if the threshold level Δ increases, the magnitude of the probability only reduces and the probability doesn't shift much.

Figure 6 shows the tradeoff curves between average number of transitions and average transition E_b/N_0 for increasing the threshold level (0.0-0.7 dB) at the standard deviation of the E_b/N_0 estimation error of σ =0.06 and 0.5 dB. Increasing the threshold level Δ can reduce the average number of transitions. This figure also shows how the E_b/N_0 estimate accuracy affects the performance of the algorithm. As the E_b/N_0 estimation error σ is increased, average number of transitions is extremely increased, that is, this means that the iteration state switches frequently.



(a) σ =0.06 (b) σ =0.5

Fig. 5 Probability of assignment to $State_1$ and $State_2$, and probability of transition.



(a) σ =0.06 (b) σ =0.5

Fig. 6 Average number of transitions versus average transition E_b/N_0 ; $State_1$ through $State_2$.

5. DISCUSSION

Figure 4 demonstrates the BER vs. E_b/N_0 under different E_b/N_0 estimation errors. Simulation results show some BER notches. If we want to have a flat BER curve, we need to select an applicable hysteresis. As shown in Figures 4, hysteresis=0.3 is suitable to obtain a smooth BER curve. As shown in Figure 6, in the case of hysteresis=0.3, the average number of transitions is also sufficiently small.

In Figure 4(b) it can be seen that when we increase the hysteresis for adaptive iterations in Turbo decoding, the performance of BER will increase. The effects of increasing the hysteresis are the same in Figure 4(a). Also, as shown in Figure 6, a large hysteresis level can remove unnecessary transitions. The performance in large hysteresis will be better than that in in small hysteresis. However, in a large hysteresis situation, the decoder has more opportunity to stay in a state of high iteration. The drawback of increased hysteresis is that the decoder utilizes more power in high average iterations; power consumption is proportional to the number of iterations. Specifically, designers must consider a tradeoff between BER and power consumption. Although we simulated our algorithm in the AWGN channel, this algorithm is also suitable for a slow fading environment. As results of the analysis, low noise environments give the clinical variation for the adaptive iteration algorithm. In our analysis, low noise environments provided sufficient clinical variation for the adaptive iteration algorithm and the hysteresis easily controlled the BER. This conclusion is based on analysis results of transition probability at a condition of noise σ =0.06. With regard to the result of noise σ =0.06, shown in Figure 5 (a), the probability curve is critical. However, the curve of the adaptive iteration algorithm becomes broad when noise σ =0.5. This means that under this condition, hysteresis was not effective for the control of BER performance; this result was obtained by analysis of transition probability when noise σ =0.5, as shown in Figure 5 (b). That is to say, when noise is small, adaptive iteration is more effective. When noise is large, BER performance is broad but hysteresis is not effective. This result shows the influence of the average number of transitions. When the noise is small, the average number of transitions becomes once suddenly. When noise is large, the average number of transitions cannot be reduced.

6. CONCLUSION

We studied the performance of an adaptive algorithm based on the E_b/N_0 measurement and the effect of the hysteresis in an AWGN environment in order to verify the feasibility of the algorithm in practical application. With this approach, the receiving stations determine for themselves the number of decoding iterations they need, based on the E_b/N_0 measurement. Moreover, this kind of adaptive iteration does not require signaling between sender and receiver. In this way, a considerable amount of decoding power can be saved, and low power consumption is essential for portable applications. As a consequence, the idea of adaptive decoding of turbo codes shows interesting potential for future usage in indoor environments. The hysteresis will affect the transition number of iterations, in that it influences the average number of iterations and the performance of BER. A suitable hysteresis must be selected in order to achieve good balance between power and performance.

ACKNOWLEDGMENTS

This research was fully supported by Lite-on Technology, Taiwan. I would like to thank Mr. Robert Lai and Mr. Harrison Chen for providing encouragement and discussion support during the research.

REFERENCES

[1] C. Berrrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: turbo-codes,” ICC 1993, Geneva, Switzerland, pp.1064-1070, May 1993.

[2] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, “A soft-input soft-output Maximum A Posteriori (MAP) module to decode parallel and serial concatenated codes”, TDA progress report 42-127, November 15, 1996.

[3] 3GPP TS 25.212: “3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Multiplexing and channel coding (FDD)”.

[4] Harri Holma and Antti Toskala, “WCDMA for UMTS”, 2002.

[5] Interim V&V Text for CDMA-2000 Physical Layer (Revision 8.3), TIA, 1999.

[6] A. S. Barbulescu and S. S. Pietrobon, “Turbo codes: A tutorial on a new class of powerful error correcting coding schemes: Part 2: Decoder design and performance”, Journal of Electrical and Electronics Engineering, Australia, vol. 19, pp. 143-152, Sept. 1999.

[7] John Dielissen, Jef van Meerbergen, and Marco Bekooij, “Power efficient layered turbo decoder processor implementation”, accepted on 2001, Munich.

[8] F. Gilbert et al., “Low Power Implementation of a Turbo-Decoder on Programmable Architectures”. ASP-DAC, 2001.