A Novel Stop Criterion for Turbo Decoding

Student Member Hua LIN (Chiba University)
Student Member Xiaoqiu WANG (Chiba University)
Member Jianming LU (Chiba University)
Non-member Takashi YAHAGI (Chiba University)

Iterative decoding is a key feature of turbo code and each decoding results in additional computations and decoding delay. In this paper, we present a simple and effective criterion for stopping the iteration process in turbo decoding with a negligible degradation of the error performance. Simulation results show the effectiveness of the proposed dynamic decoding method.

Keywords: turbo code, stop criterion, iterative decoding

1. Introduction

Iterative decoding is a key feature of turbo code and each decoding results in additional computations and decoding delay. As the decoding approaches the performance limit of a given turbo code, any further iteration results in very little improvement. Usually each frame is decoded for M preassigned iterations and it is set with the worst corrupted frame in mind. Most frames need fewer iterations to converge. It would reduce the average computation substantially without performance degradation if the decoder terminated the iterations for each individual frame immediately after the bits are correctly estimated. It is natural to use BER as the stop criterion because the BER decreases with the increase of iterations. However, in practical transmission system, the BER cannot be directly obtained at the receiver. One stop criterion has been proposed by Hagenauer et al. to monitor the cross entropy between the set of information bit reliabilities for the current iteration and the previous iteration. When the change in this cross entropy falls below a threshold, the iterative decoding process is terminated. The basic idea is that iterative decoding should stop when the decoder's reliability values are stable. This criterion is known as CE criterion.

In this paper, we focus on the probability of the bit correct ratio at the end of decoder. The iterative decoding process will be stopped when this probability value is large enough. Simulation results show that the proposed method is effective and is easily realized.

2. The Structure of Turbo Code

A two dimensional iterative decoder is shown in Fig. 1. Assuming BPSK transmission over an AWGN channel, let u be the information sequence and y be the received sequence. At the ith iteration, let \( L^{(i)}_m(u_k) \) and \( \bar{L}^{(i)}_m(u_k) \) denote the log-likelihood-ratio (LLR) and the extrinsic value of the estimated information bits \( u_k \) delivered by decoder \( m \), respectively, with \( m = 1, 2 \). It is shown in (2) that

\[
L^{(i)}_1(u_k) = L^{(i-1)}_2(\hat{u}_k) + \frac{2}{\sigma^2} y_{k,1} + L^{(i)}_1(\hat{u}_k) \tag{1}
\]

\[
L^{(i)}_2(u_k) = L^{(i)}_1(\hat{u}_k) + \frac{2}{\sigma^2} y_{k,1} + L^{(i)}_2(\hat{u}_k) \tag{2}
\]

where \((2/\sigma^2)y_{k,1}\) is the channel soft value. Based on the log-likelihood ratio of \( \hat{u}_k \) from (2), the estimation of bit \( u_k \) is derived by

\[
\hat{u}_k = \text{sign}(L^{(i)}_1(\hat{u}_k)) + 1)/2 \tag{3}
\]

The turbo decoding process is often performed by applying SOVA (soft output Viterbi algorithm) or MAP (maximum a posteriori) algorithm in an iterative way between modules DECI and DEC2. The extrinsic information of each module is reused as a priori information for the other module. After a certain number of iterations, the system performance will approach the limit.

3. The Proposed Stop Criterion for Turbo Decoding

The usage of stop criterion can reduce the number of iterations by terminating the decoding process when no more gain is expected. In this paper, we present a stop criterion based on the probability of bit correct ratio (BCR) at the decoder output.

From (2), we know that the decoding algorithm is based on log-likelihood ratios, which are defined as

\[
L^{(i)}_2(\hat{u}_k) = \log \left( \frac{P(\hat{u}_k = 1)}{P(\hat{u}_k = 0)} \right) \tag{4}
\]
Then the probability that \( u_k \) equals 0 or 1 is derived according to equation (4).

\[
P(\hat{u}_k = 1) = \frac{\exp(L_2^{(i)}(\hat{u}_k))}{1 + \exp(L_2^{(i)}(\hat{u}_k))} \\
P(\hat{u}_k = 0) = \frac{1}{1 + \exp(L_2^{(i)}(\hat{u}_k))} \quad \cdots \quad (5)
\]

Therefore, the probability \( P(\hat{u}_k = u_k) \) is estimated by

\[
P(\hat{u}_k = u_k) = \frac{\exp(L_2^{(i)}(\hat{u}_k))^u_k}{1 + \exp(L_2^{(i)}(\hat{u}_k))} \quad \cdots \quad (6)
\]

In order to analyze the probability of bit correct ratio for simplicity, we take the logarithm of both sides of equation (6), then the log-probability of bit correct rate is expressed as

\[
\log(P(\hat{u}_k = u_k)) = L_2^{(i)}(\hat{u}_k) - \log(1 + \exp(L_2^{(i)}(\hat{u}_k))) \quad \cdots \quad (7)
\]

If the entire data block is taken into consideration, then \( P_{\text{correct}}(i) \) is expressed as the probability that a block of length \( F \) is correct after the \( i \)th iteration and it is estimated by

\[
\log(P_{\text{correct}}(i)) = \sum_{k=1}^{F} [L_2^{(i)}(\hat{u}_k) - \log(1 + \exp(L_2^{(i)}(\hat{u}_k)))] \quad (8)
\]

From the analysis of the turbo decoding process, we conclude that \( \log(P_{\text{correct}}(i)) \) is increased with the increment of iteration numbers until no more gain is expected. Now, let us look at the subsequent iterations \( (i-1) \) and \( (i) \), where the deviation of \( \log(P_{\text{correct}}(i)) \) is defined as

\[
T(i) = \Delta \log(P_{\text{correct}}(i)) = \log(P_{\text{correct}}(i)) - \log(P_{\text{correct}}(i - 1)) \quad \cdots \quad (9)
\]

Therefore, we could use the criterion

\[
\left| \frac{T(i)}{\log(P_{\text{correct}}(i - 1))} \right| < \text{threshold} \quad \cdots \quad (10)
\]

as a stop criterion for the iterations. Simulation results have shown that a threshold value of 8\% is appropriate to stop the iterations. The benefit of this stop criterion will be shown in the next section.

### 4. Simulation Results and Analysis

Simulations for turbo code using the proposed dynamic iterations are presented in this section. The turbo code scheme is executed in BPSK transmission system, and the maximum iteration number is set to 15. Generator matrix is \((7, 5)\) (expressed in octal form), code rate is \(1/2\) (punctured). The adopted decoding algorithm is maximum a posteriori probability (MAP) \((3)\), and the selected information frame size is 1000 and 10000. Table 1 and Table 2 show the performance evaluation of the proposed turbo decoding algorithm using dynamic iterations (maximum iteration number is 15), compared with the traditional method \((3)\) using fixed iterations when the transmitted frame size is 1000 and 10000, respectively. \( \text{Iter.} \) is the average value of the proposed dynamic iteration numbers. From simulation results, we find that most decoding stages stop early when the BER performance reaches limit and this will greatly increase the efficiency of the turbo code system. Moreover, we also find that the average iteration number decreases fast with a negligible degradation of the error performance when the SNR is increased.

### 5. Conclusion

In this paper, a novel stop criterion for turbo decoding algorithm has been presented. It changes the traditional fixed iterative decoding stages into dynamic iterative stages with about the same error performance as the traditional method because the stop criterion effectively allows the decoder to spend more iterations on “tough” data frames, and fewer iterations on “easy” ones, thus reducing the computational complexity. Simulation results have shown the effectiveness of the proposed method.

(Manuscript received December 20, 2000)

### References


---

<table>
<thead>
<tr>
<th>Table 1. Performance evaluation of the proposed stop criterion (Frame size=1000)</th>
<th>Table 2. Performance evaluation of the proposed stop criterion (Frame size=10000)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BER</strong></td>
<td><strong>Iter.</strong></td>
</tr>
<tr>
<td><strong>E_b/N_0 (dB)</strong></td>
<td><strong>Proposed (Max iter. =15)</strong></td>
</tr>
<tr>
<td>1</td>
<td>1.31 x 10^-2</td>
</tr>
<tr>
<td>1.5</td>
<td>1.06 x 10^-3</td>
</tr>
<tr>
<td>2</td>
<td>8.48 x 10^-5</td>
</tr>
<tr>
<td>2.5</td>
<td>2.80 x 10^-5</td>
</tr>
</tbody>
</table>