社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
IT2005-24(2005-05)

A Study of Reliability Based Hybrid ARQ Scheme with Bitwise Posterior Probability Evaluation from Message Passing Algorithm

Daiki KOIZUMI, Naoto KOBAYASHI, Toshiyasu MATSUSHIMA, and Shigeichi HIRASAWA

Waseda University
3-4-1, Okubo, Shinjuku, Tokyo, 169-8555, JAPAN
E-mail: dkoizumi@matsu.mgmt.waseda.ac.jp

Abstract Reliability Based Hybrid ARQ (RBH-ARQ) is one of hybrid ARQ schemes with the modified decision feedback. In RBH-ARQ, the modified feedback is composed of both ACK/NAK signal and unreliable bit index which is evaluated from bitwise posterior probability. In the conventional RB-ARQ, the sender retransmits just unreliable information bits with no coding when unreliable bits are detected on the receiver and retransmission occurs. In the proposed RBH-ARQ, on the other hand, the sender retransmits not information bits but newly encoded parity bits corresponding to the unreliable information bits assuming systematic convolutional coding. Furthermore, the proposed scheme assumes message passing algorithm for Maximum A Posteriori probability (MAP) decoding on the receiver. The receiver puts received bits including retransmitted parity bits all together into our proposing probability model. As a result, better performance can be expected in the proposed RBH-ARQ since our probability model is similar to the celebrated decoding model of multiple Turbo codes. Finally, brief simulation results based on several message passing schedules in our algorithm would be shown.

Key words Reliability Based Hybrid ARQ, Bitwise MAP Decoding, Message Passing Algorithm

1 Introduction

Reliability Based Hybrid ARQ (RBH-ARQ)[4][5] is one of hybrid ARQ schemes with the modified decision feedback. In this scheme, the modified decision feedback is composed of both ACK/NAK signal and unreliable bit index. In order to get unreliable bit index from received sequence, bitwise decoding techniques are required. One of them is Maximum A Posteriori probability (MAP) decoding technique including BCJR algorithm[2] which calculates exact bitwise posterior probability of information bit under received sequence for convolutional codes. Moreover, the celebrated decoding algorithm of Turbo codes[1][3] utilizes parallel BCJR algorithm to calculate bitwise approximate posterior probability. RBH-ARQ normally utilizes such posterior probability to evaluate bitwise reliability of received information sequence. Unreliable bit index is determined by comparing this reliability (log ratio of binary posterior probabilities) and the pre-defined threshold. For example, if the log ratios of certain bits are proved to be less than one, then these bits are regarded as unreliable. After getting this unreliable bit index, the receiver sends back both NAK signal and unreliable bit index to the sender where the feedback channel is assumed to be noiseless as [4][5] do.

When the sender receives the modified decision feedback, the retransmission occurs. The sender then retransmits certain information about unreliable bits depending on the type of RBH-ARQ. In [4], the sender retransmits unreliable information bits with no coding. In another RBH-ARQ scheme[5], nonsystematic convolutional codes are taken and all bits corresponding to unreliable information bits are retransmitted.

Apart from these schemes, in our proposed RBH-ARQ scheme, the sender retransmits newly encoded single parity bit per an unreliable information bit for every retransmission where the half rate (for example) systematic convolutional codes are used in encoding. Moreover, the proposed scheme also assumes message passing algorithm for MAP decoding on the receiver. The receiver puts received bits including retransmitted parity bits all together into our proposing probability model. Since this model is similar to the celebrated decoding model of multiple Turbo codes[3], the better performance can be expected. We prepare several algorithms depending on message passing schedules on our decoding model in simulation and finally brief results are shown.

This paper is organized as follows. The next section 2 defines basic notations and RBH-ARQ model. Section 3 describes the conventional RBH-ARQ schemes in [4][5]. In section 4, we propose the improved RBH-ARQ scheme and show overview of decoding model of multiple Turbo codes. In section 5, we execute some simulations to analyze RBH-ARQ schemes. Last section 6 concludes our research.

2 Basic RBH-ARQ Model

First of all, we describe basic RBH-ARQ model[4][5]. Let $u_i \in \{0,1\}$, $(i = 1, 2, \cdots, M)$ be information bit sequence. In Figure 1, the encoder (the sender) takes each u_i and produces output x_i as the codeword. The sequence of x_i is then BPSK modulated and transmitted under the AWGN channel, assuming that the channel noise parameter is known to the decoder (the receiver).

The decoder beyond the channel takes the sequence of y_i where y_i is the receivedword and calculates bitwise posterior probability $p(u_i|y)$ where $y = y_1 y_2 \cdots y_M$. The error detector calculates the log ratio of (binary) bitwise posterior probabilities, i.e. $\log\{p(u_i = 0|y)/p(u_i = 1|y)\}$, from output of decoder and evaluates the reliability of each information bit. Taking pre-defined threshold λ, if $|\log\{p(u_i = 0|y)/p(u_i = 1|y)\}| < \lambda$ holds, the error detector regards ith bit as unreliable. After the whole sequence of y_i is processed, the error detector sends back both NAK signal and unreliable bit index through noiseless feedback channel. If $|\log\{p(u_i = 0|y)/p(u_i = 1|y)\}| > \lambda$ holds, on the other hand, the error detector sends back ACK signal under feedback channel and performs bitwise Maximum A Posteriori probability (MAP) decoding to estimate $\hat{u_i}$. Since NAK signal contains unreliable bit index, we regard it as not conventional decision feedback but *modified decision feedback.*

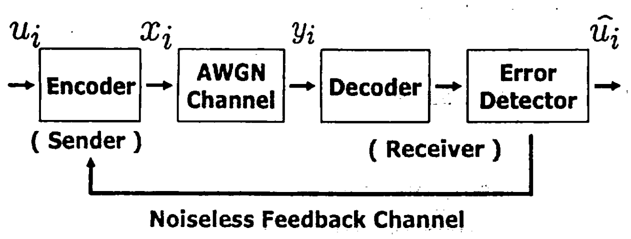

Figure 1: Basic RBH-ARQ Model

For the rest part of this paper, we shall simply call the encoder as the sender, both decoder and error detector as the receiver, respectively. If the modified decision feedback from the receiver contains NAK signal, the retransmission occurs at the sender. In this phase, the sender must retransmit certain information about unreliable bits to the receiver. The retransmitted information depends on the type of RBH-ARQ. The next section explains some types of them in detail.

3 The Conventional RBH-ARQ Schemes [4][5]

Suppose $j = 0, 1, \cdots$ is the number of retransmissions and let $x_i(j)$ be the jth retransmitted codeword, $y_i(j)$ be the corresponding receivedword, respectively, where i is bit index ($i = 1, 2, \cdots, M$) again. Additionally, let $y(j)$ denote the sequence: $y_1(j)y_2(j)\cdots y_M(j)$. In the conventional RBH-ARQ scheme of [4], the sender firstly (when $j = 0$) transmits sequence of Turbo codewords $x_i(0)$. The receiver receives the sequence of $y_i(0)$ through AWGN channel and calculates bitwise posterior probability $p(u_i|y(0))$. For bitwise decoding of Turbo codes, the algorithm described in [1] which is parallel version of BCJR algorithm[2] is well-known. By taking such posterior probability, bitwise reliability evaluation at the receiver can be defined as follows:

Definition 3.1 *(Unreliable Bit Detection)*
When the jth retransmission finishes, the receiver detects unreliable ith bit if the following holds:

$$L_i(j) \triangleq \left| \log \frac{p(u_i = 0|y(0), y(1), \cdots, y(j))}{p(u_i = 1|y(0), y(1), \cdots, y(j))} \right| < \lambda, \quad (1)$$

where λ is pre-defined threshold.

If equation (1) is satisfied, the receiver returns the modified decision feedback which consists of both NAK signal and unreliable bit index. Otherwise the receiver returns ACK signal. These signals are sent through feedback channel to the sender.

If the sender receives NAK signal, the retransmission occurs. Suppose $U(j) \subseteq \{1, 2, \cdots, M\}$ be a set of unreliable bit indices corresponding to the jth retransmission. The sender then retransmits unreliable information bit sequence u_i through AWGN channel where $i \in U(j)$. This means that $x_i(j) = u_i, \forall j > 0$ in [4]. The receiver calculates $L_i(j)$ in (1) and use $\sum_j L_i(j)$ for decoding. The whole retransmission procedure of [4] is shown in Figure 2.

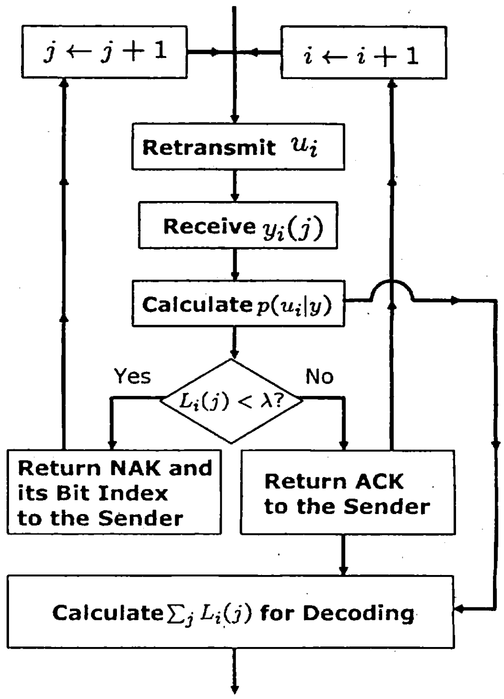

Figure 2: The Conventional RBH-ARQ Retransmission

In another RBH-ARQ scheme[5], the sender firstly transmits the sequence of non-systematic convolutional codeword of $x_i(0)$. As same as [4], the receiver takes the sequence of $y(0)$, performs BCJR algorithm to calculate $p(u_i|y(0))$, and evaluates bitwise reliability in (1). In this scheme, if the jth retransmission occurs, the sender retransmits not unreliable information bits $u_i, i \in U(j)$ but the whole codeword $x_i(j), i \in U(j)$ corresponding to the unreliable information bits. At the receiver, (the jth) BCJR algorithm is executed and $\sum_j L_i(j)$ is taken for decoding.

4 Proposed RBH-ARQ Scheme

4.1 Brief Procedure Description

In this section, we shall explain the proposed scheme with Figure 3. In our proposed scheme, the sender transmits systematic convolutional codes (half rate for example here) $x_i(0) = (x_{i,0}, x_{i,1})$ where $x_{i,0} = u_i$ and $x_{i,1}$ is parity bit for the first transmission: $j = 0$. The receiver beyond the AWGN channel takes the sequence of $y_i(0) = (y_{i,0}, y_{i,1})$, $(i = 1, 2, \cdots, M)$ and executes BCJR algorithm to calculate posterior probability $p(u_i|y(0))$ where $y(0) = y_1(0)y_2(0)\cdots y_M(0)$.

Nextly, the error detector evaluates bitwise reliability by (1) using $p(u_i|y)$ as [4][5] do. If unreliable bits are detected, the receiver sends back NAK signal as well as unreliable bit index through feedback channel. If the retransmission occurs, the sender newly encodes unreliable bit u_i and retransmits it as $x_i(j)$ to the receiver for the jth retransmission. In this phase, retransmitted bit index should be interleaved since we would make use of decoding algorithm of multiple Turbo codes later. Note that the conventional two schemes retransmit unreliable information bits and the all bits corresponding to the unreliable information bits, respectively.

Lastly, the receiver takes the sequence of $y_i(j)$, $i \in U(j)$ where $U(j)$ is the set of unreliable information bits for jth retransmission again. For every retransmission, the receiver de-interleaves bit index of $y_i(j)$ and executes decoding algorithm of multiple Turbo codes[3] for ith unreliable bit. This decoding strategy preserves sub-optimality in terms of calculating posterior probability even if the retransmission frequently occurs, whereas the one in conventional schemes does not.

4.2 Our Decoding Model Similar to that of Multiple Turbo Codes

This subsection explains the decoding model of multiple Turbo codes from which our proposed scheme is derived. In the following explanation, both interleaver and de-interleaver are abbreviated for the simplicity of notation. As same as most of reference books about Turbo codes, we shall basically follow the deriving process of BCJR algorithm[2].

Let $k = 1, 2, \cdots$ be the number of constituent codes where $k = j + 1, (j = 0, 1, \cdots)$ holds. By using k, let $x_i^k = (x_{i,0}, x_{i,k})$ be the constituent code and $y_i^k = (y_{i,0}, y_{i,k})$ be the constituent receivedword, respectively. In the decoding of multiple Turbo codes, the posterior probability of $p(u_i|y)$ is approximated by that of constituent code: $p(u_i|y^k)$ where $y^k = y_1^k y_2^k \cdots y_M^k$.

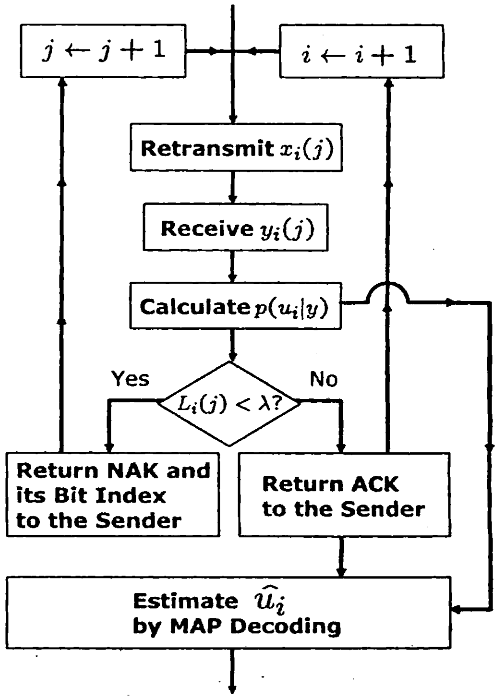

Figure 3: Proposed RBH-ARQ Retransmission

In terms of message passing algorithm on the graphical model such as Factor Graph, Bayesian Network etc. the constituent code can be represented by single subgraph which is subset of the whole graphical model. The output of posterior probability in the subgraph approximates the posterior probability of the multiple Turbo codes. To do this, exchanging extrinsic information between subgraphs is assumed. The above posterior probability then becomes the followings:

$$
\begin{aligned}
&p(u_i = a|y) \\
&\quad \approx p(u_i = a|y^k) && (2)\\
&\quad = p(u_i = a|y_1^k, y_2^k, \cdots, y_M^k) && (3)\\
&\quad = \frac{1}{p(y^k)} \sum_{(s_i, s_{i+1}) \in A} p(S_{i,k} = s_i, S_{i+1,k} = s_{i+1}, y^k), && (4)
\end{aligned}
$$

where $a \in \{0, 1\}$, $p(y^k) = p(y_1^k, y_2^k, \cdots, y_M^k)$, both $S_{i,k}$ and $S_{i+1,k}$ are possible states in trellis diagram, and A is a set of states in trellis diagram such that $u_i = a$.

According to [2], the following holds in (4):

$$
\begin{aligned}
&p(S_{i,k} = s_i, S_{i+1,k} = s_{i+1}, y^k) \\
&\quad = p(y_{t>i}^k|s_{i+1})p(s_{i+1}, y_i^k|s_i)p(s_i, y_{t<i}^k), \quad (5)
\end{aligned}
$$

where t is bit index in trellis diagram.

The first and third terms in RHS of (5) can be recursively calculated using the second term[2]. For the second term, the following transformation is possible:

$$p(s_{i+1}, y_i^k|s_i)$$

$$= \quad p(u_i = a)p(y_{i,0}|u_i)p(y_{i,k+1}|x_{i,k+1}). \qquad (6)$$

By substituting (4) by both (5) and (6), we have

$$\begin{aligned} & p(u_i = a|y_1^k, y_2^k, \cdots, y_M^k) \\ = & \frac{1}{p(y^k)} p(u_i = a) \\ & \times \sum_{(s_i, s_{i+1}) \in A} \left[p(y_{t>i}^k | s_{i+1}) \right. \\ & \left. \times \{ p(y_{i,0}|u_i) p(y_{i,k+1}|x_{i,k+1}) \} \, p(s_i, y_{t<i}^k) \right]. \end{aligned} \qquad (7)$$

In (7), $\sum_{(s_i,s_{i+1})\in A} \left[p(y_{t>i}^k|s_{i+1})\right.$ $\left.\{p(y_{i,0}|u_i)p(y_{i,k+1}|x_{i,k+1})\}p(s_i, y_{t<i}^k)\right]$ is frequently referred as *extrinsic information*. In the celebrated decoding algorithm of Turbo codes, this extrinsic information is normally exchanged among plural subgraphs. For the case of $k = 2$ (where $j = 1$), there exists two subgraphs and they would be shown in Figure 4 if we take Bayesian Network as a graphical model to express our decoding probability model which is similar to multiple Turbo codes. In Figure 4, note that $x_{i,1}$ as well as $y_{i,1}$ are missing since ith bit is reliable enough and the retransmission does not occur.

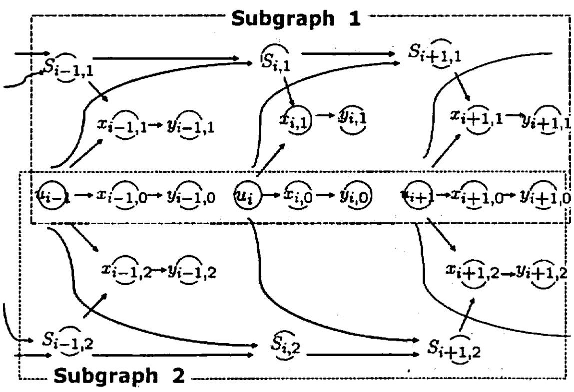

Figure 4: Example of Our Decoding Model which is Similar to that of Multiple Turbo Codes Expressed by Bayesian Network ($k = 2$).

For exchanging extrinsic information among subgraphs, we shall assume message passing algorithm on our probability model. Although there exists several message passing schedules on one given graph, they would be considered in the next section and we assume that full parallel messaging schedule for the simple explanation here. For kth single subgraph, outgoing and incoming messages are defined as the followings:

Definition 4.1 *(Outgoing Message from kth Subgraph) Outgoing message from kth subgraph, $M_{k\rightarrow}$, is equivalent to extrinsic information in equation (7):*

$$\begin{aligned} M_{k\rightarrow} \triangleq & \sum_{(s_i,s_{i+1})\in A} \left[p(y_{t>i}^k|s_{i+1})\right. \\ & \left.\{p(y_{i,0}|u_i)p(y_{i,k+1}|x_{i,k+1})\}p(s_i, y_{t<i}^k)\right]. \end{aligned} \qquad (8)$$

Definition 4.2 *(Incoming Message to kth Subgraph) Incoming message to kth subgraph, $M_{k\leftarrow}$, is defined as the product of extrinsic information of all subgraphs except for that of kth subgraph:*

$$M_{k\leftarrow} \triangleq \prod_{l=1,l\neq k}^{l=N} M_{l\rightarrow}. \qquad (9)$$

On the above definition of incoming message, the following examples can be typical cases:

- For $k = 2$, simply exchanging messages each other:

$$M_{1\leftarrow} = M_{2\rightarrow}, \qquad (10)$$
$$M_{2\leftarrow} = M_{1\rightarrow}. \qquad (11)$$

- For $k = 3$, the extension of the case of $k = 2$:

$$M_{1\leftarrow} = M_{2\rightarrow} \cdot M_{3\rightarrow}, \qquad (12)$$
$$M_{2\leftarrow} = M_{1\rightarrow} \cdot M_{3\rightarrow}, \qquad (13)$$
$$M_{3\leftarrow} = M_{1\rightarrow} \cdot M_{2\rightarrow}. \qquad (14)$$

Figure 5 shows examples of message passing among subgraphs for $j = 2, 3$. Figure 5 assumes full parallel message passing for simplicity, however, several message passing schedules can be considered:

- Full Parallel Schedule:

  For $k = 2$,
  Subgraph1 $\rightleftharpoons$ *Subgraph2.*

- Semi Parallel Schedule:

  For $k = 3$,
  Subgraph1 $\rightleftharpoons$ *Subgraph2* $\rightleftharpoons$ *Subgraph1* $\rightleftharpoons$ *Subgraph3* $\rightleftharpoons$ *Subgraph1* $\rightleftharpoons$ *Subgraph2*, $\cdots$, and so forth.

- Serial Schedule:

  For $k = 3$,
  Subgraph1 $\rightarrow$ *Subgraph2* $\rightarrow$ *Subgraph3* $\rightarrow$ *Subgraph1* $\rightarrow \cdots$.

These effects for RB-HRQ performance are examined in the next section by simulation.

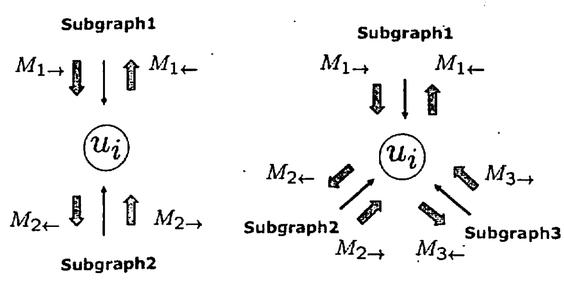

Figure 5: Example of Message Passing for $k = 2, 3$.

5 Simulation

To examine performance of the proposed RBH-ARQ, we took two RBH-ARQ schemes: one is the modified version of [4], the other is proposed one. Brief description of them is the followings:

The Conventional RB-AHQ Scheme:

- For the sequence of $x_i(j)$, the sender transmits half rate of the systematic convolutional codes.
- The sender retransmits *unreliable information bit* for every retransmission, i.e. $x_i(j) = u_i, \forall j > 0$.
- The receiver performs decoding by $\sum_j L_i(j)$.

Proposed RB-AHQ Scheme:

- For the sequence of $x_i(j)$, the sender transmits half rate of the systematic convolutional codes.
- The sender retransmits *single parity bit* corresponding to unreliable information bit for every retransmission, i.e. $x_i(j) = x_{i,k}, \forall j > 0$.
- Random interleaver is assumed for every retransmission.
- The receiver performs MAP decoding whose posterior probability is derived from message passing algorithm described in subsection 4.2.

5.1 Simulation1

In this simulation, we compare the performances of conventional and proposed schemes with the following conditions:

- Average Throughput value from 100 ARQ Procedure Executions is taken.
- RBH-ARQ Schemes: Conventional Scheme and Proposed Scheme with Full parallel Messaging Schedule described in the previous section.
- Length of Information Sequence: 1024.
- Constraint Length of Systematic Convolutional Codes: 3.
- Fixed E_s/N_0[dB]: 0.00.
- BER is fixed 7.0×10^{-3} by changing threshold λ about the bit reliability.
- Number of Turbo Iterations for Proposed Scheme: 10.

Table 6: Result of Simulation1

Messaging Schedule	Throughput	Retransmission
Conventional	0.468	3.11
Full Parallel	0.451	3.81

5.2 Simulation2

In this simulation, we compared performances of proposed schemes by changing their messaging schedules. The simulation conditions are the followings:

- Average Throughput value from 100 ARQ Procedure Executions is taken.
- Messaging Schedules of Decoding: Full parallel, Semi Parallel, and Serial Schedules described in the previous section.
- Length of Information Sequence: 1024.
- Constraint Length of Systematic Convolutional Codes: 4.
- Fixed E_s/N_0[dB]: -1.00.
- BER is fixed 7.0×10^{-3} by changing threshold λ about the bit reliability.
- Number of Turbo Iterations: 10.

With the above conditions, the following Table 7 is obtained.

Table 7: Result of Simulation2

Messaging Schedule	Throughput	Retransmission
Full Parallel	0.383	7.88
Semi Parallel	0.390	7.69
Serial	0.388	7.74

5.3 Discussion

In simulation1, we set relatively advantageous E_s/N_0 for the conventional scheme. From the Table 6, the performances of both schemes are almost same in terms of throughput as well as the number of retransmissions. We analyzed retransmission processes in detail and it turned out that the number of retransmissions reduced rapidly in the conventional scheme. In the conventional scheme, almost all bits are turned over to *reliable* for the second retransmission. Additionally, we should point out that the conventional scheme was extremely faster than the proposed one since its procedure is quite simple. This can be big advantage if the channel condition is relatively good.

In the proposed scheme, on the other hand, the number of unreliable bits did not decrease so rapidly. Only the half of them turned over to *reliable bit* for most of the second retransmission. Under the severe E_b/N_0 conditions, however, BER of the proposd scheme remained around 7.0×10^{-3} without chaning λ drastically. From this result, it is expected that the proposed scheme has better performance under the negative E_s/N_0 as same as the cereblated performance of the multiple Turbo codes

In simulation2, we set relatively low E_s/N_0 to clarify the differences of three messaging schedules. From the Table 7, however, the performances of them have no remarkable differences. According to the data of Turbo iteration processes, their trends of convergence (of posterior probabilities) were almost same. Hence the convergent values of them were also proved to be almost same.

6 Conclusion

In this paper, we proposed the improved RBH-ARQ scheme with both new retransmission procedure and several messasgeing schedules in decoding. For relatively good channel condition (typically $E_s/N_0 = 0.0$), the convensional and proposed scheme have almost same performance in simulation. But the complexity of calculations in the proposed scheme is extremely larger than the convensional scheme since it has Turbo-like iteration. This cost might be paid under the severe channel conditions such as negative E_s/N_0. For messaging schedules in the proposed scheme, no remarkable difference is obtained.

Acknowledgments

One of authors, D.KOIZUMI appreciates their helpful comments from T.NIINOMI in Kanagawa Institute of Technology and Y.UKITA in Yokohama College of Commerce.

This research is partially supported by Category (C) No.15560338 of Grant-in-Aid for Scientific Research, Japan Society for the Promotion of Science.

References

[1] C.Berrou, A.Glavieux and P.Thitimajshima, "Near Shannon Limit Error-correcting Coding and Decoding: Turbo-codes," In *Proceeding of the IEEE International Conference on Communications*, Vol.2, pp.1064-1070, May. 1993.

[2] L.R.Bahl, J.Cocke, F.Jelinek and J.Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate," *IEEE Transactions on Information Theory*, Vol.20, Issue.2, pp.284-287, Mar. 1974.

[3] D.Divsalar and F.Pollara, "Multiple Turbo Codes," In *Proceeding of the IEEE Military Communications Conference*, Vol.1, pp.279-285, Nov. 1995.

[4] H.Kim and J.M.Shea, "New Turbo-ARQ Techniques Based on Estimated Reliabilities," In *Proceeding of the IEEE Wireless Communications and Networking Conference*, Vol.2, pp.843-848, Mar. 2003.

[5] A.Roongta and J.M.Shea, "Reliability-based Hybrid ARQ using Convolutional Codes," In *Proceeding of the IEEE International Conference on Communications*, Vol.4, pp.2889-2893, May. 2003.