Design of Swap Interleaver without Edge Effect in CRC-turbo Concatenated Code

Byung Gil Lee*, Sang Jae Bae**, and Eon Kyeong Joo**

*Samsung Electronics Co. Ltd., Kyungpook, Korea
**Kyungpook National University, Daegu, Korea

Abstract - **Turbo code can achieve good error performance by iterative decoding, but more iterations result in additional computational complexity and delay. CRC-turbo concatenated code is known to be the most efficient method to reduce the number of iterations. However the performance may be degraded by the edge effect in this scheme like the conventional turbo code without CRC. A method to eliminate the edge effect is proposed by adopting D-parameter to the conventional swap interleaver in this paper. As results of simulations, the edge effect is shown to be successfully eliminated by using the new interleaver designed with D-parameter.**

I. INTRODUCTION

TURBO code can achieve good error performance by iterative decoding, but more iterations result in additional computational complexity and delay [1,2]. Thus, various methods to reduce the number of iterations have been proposed. CRC-turbo concatenated code is known to be the most efficient method among them [3].

They incorporate CRC code before turbo code and frame error is detected by CRC code at each iteration step. If no error is detected, iteration is stopped. Otherwise, it is continued.

However the error performance may be degraded by the edge effect [4] in this scheme. If the bits located near the end of an information frame are also mapped near the end of a frame after interleaving, these might result in high bit error rate (BER). This is because the weight-1 input sequences where a bit '1' is located near the end of a frame generate relatively small free distance in turbo code [5]. This is one major reason that the constituent encoder is forced to return to all-zero state by appending tail bits. But this method cannot eliminate edge effect completely.

A swap interleaver [6,7] has been proposed recently which shows better performance than s-random interleaver [8] for turbo code.

B. G. Lee is with the Mobile Communication Division, Samsung Electronics Co. Ltd., 94-1, Imsoo-Dong, Kumi, Kyungpook 730-350, Korea.

S. J. Bae and E. K. Joo are with the School of Electronic and Electrical Engineering, Kyungpook National University, Daegu 702-701, Korea. Phone: +82 53 950 5542, Fax: +82 53 950 5505, E-mail: ekjoo@ee.knu.ac.kr

D-parameter is adopted in a conventional swap interleaver to remove the possible edge effect of CRC-turbo concatenated code in this paper. The error performance and average number of iterations are analyzed by computer simulations.

II. INTERLEAVER DESIGN

A. D-parameter

The error performance of turbo code using iterative decoding depends largely on the structure of the interleaver used because the interleaver affects the distance properties of the codewords. That is, the performance of a turbo code depends on how effectively the input data sequences that produce low encoded weights at the output of first constituent encoder are matched with the permutations of the same data sequences that yield higher encoded weights at the other constituent encoder. Therefore considerable efforts have been made to obtain such a good interleaver, and some successful results have been obtained [6-8].

Generally the non-self-terminating sequences have little effect on the performance of turbo code because they accumulate high encoded weights until trellis is artificially terminated at the end of frame [8]. However the non-self-terminating sequences near the end of frame may generate codeword with low weight after turbo encoding like self-terminating sequences. That is, a bit '1' near the end of an information frame may accumulate relatively low weights at the first constituent encoder. The position of the bit '1' may be also near the end of frame after interleaving in this case. Then the resulting overall coded sequence still has low weight.

Thus not only self-terminating input sequences but also non-self-terminating input sequences close to the end of interleaver affect the error performance of turbo code significantly. Therefore D-parameter is adopted in this paper to avoid this bad situation, which is called the edge effect.

D-parameter is simply defined as the number of bit '0's

after a bit '1' to obtain the predetermined minimum free distance. Following is an example of non-self-terminating weight-1 input sequence. At the rate 1/2 recursive systematic convolutional (RSC) encoder with the generator polynomial $(7,5)_8$, if input sequence is $(00\cdots00100(000)^\delta)$ then the resulting output parity sequence is $(00\cdots00111(011)^\delta)$. That is, the output codeword of weight $4+2\delta$ (one for input sequence and $3+2\delta$ for parity sequence) is generated by the weight-1 input sequence with a bit '1' and $2+3\delta$ consecutive '0's. Here the weights of tail bits and the parity bits generated by the tail are not counted for simple calculation.

D-parameter should be larger than 17 to obtain the weight of codeword larger than 14 in this case, where δ is 5. Also the weight of codewords in the case of other non-self-terminating input sequences with weight-2 or more increases in the ratio nearly the same as weight-1 non-self-terminating input sequences.

B.   Design procedure of interleaver with D-parameter

The design procedure of swap interleaver with the proposed D-parameter is depicted in Fig. 1, where n_1 and n_2 represent the positions of data in the interleaver, $\pi(n_1)$ and $\pi(n_2)$ are data in positions n_1 and n_2, respectively. N is the size of interleaver and i represents the number of swapping.

Design procedure can be described as follows. First, a block interleaver with equal or similar size of span and depth is generated [6]. Then two arbitrarily and randomly chosen values in the block interleaver are swapped. S-parameter is checked after swapping as in the conventional s-random interleaver [8]. That is, bits located within the distance of the predetermined s-parameter S from nth input bit before interleaving should be separated at the distance of at least S from $\pi(n)$ after interleaving.

Also D-parameter is checked as follows after s-parameter is checked. The separation between the nth input bit and the end of frame is N-n. The separation between the interleaved position of nth input bit, $\pi(n)$ and the end of frame is N-$\pi(n)$ after interleaving. Sum of these should be made larger than the predetermined D-parameter D to obtain codewords with the required minimum free distance. This can be represented as follows;

$$(N-n_1)+(N-\pi(n_1)) > D$$
$$(N-n_2)+(N-\pi(n_2)) > D \qquad (1)$$

If any one of these two conditions is not satisfied, the swapped two values should be restored to their original positions. If all of these two conditions are satisfied, we proceed to the next iteration process. The design procedure is completed after these swapping processes have been sufficiently iterated, e.g. 100 times the frame length.
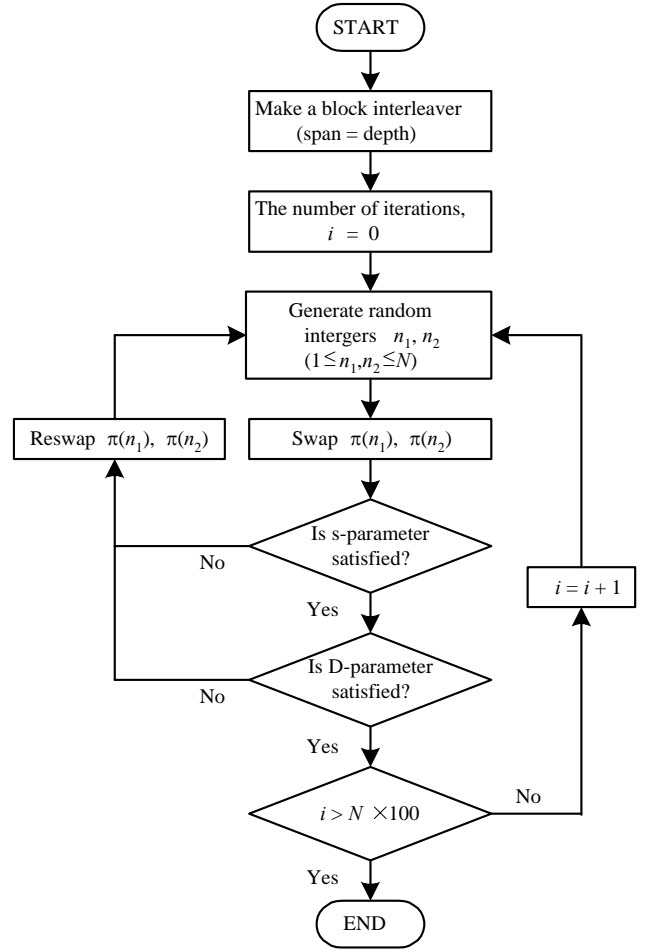

Fig. 1 Design procedure of swap interleaver with the proposed D-parameter

III. PERFORMANCE ANALYSIS

Simulation was performed for the 4-state rate-1/3 turbo code with the generator polynomial of $(7,5)_8$. The size of interleaver is 1296, and additive white Gaussian noise (AWGN) channel is assumed. The 16-bit CRC code with generator polynomial, $x^{16}+x^{15}+x^2+1$ is concatenated with the turbo code. The automatic iteration-stopping algorithm with maximum iteration of 8 is used for the decoding of CRC-turbo concatenated code. Trellis termination scheme is applied at the first constituent encoder. The same tail bits as those used at the first constituent encoder are used at the second encoder. The MAP (maximum *a posteriori*) algorithm is used for decoding [9-11]. The number of iterations at the conventional turbo code without CRC is set to 8.

The simulation results of CRC-turbo concatenated code are depicted in Fig. 2 and 3 where the frame and bit error rate versus the E_b/N_0 at various D are shown. E_b is bit energy and N_0 is noise density as usual. Here D=0

corresponds to the worst case of edge effect in the conventional interleaver.

It can be observed that FER decreases rapidly as E_b/N_0 increases when D-parameter is sufficiently large. When D is 0, worst error performance is observed by the edge effect. However the edge effect is eliminated almost completely when D is larger than 15.
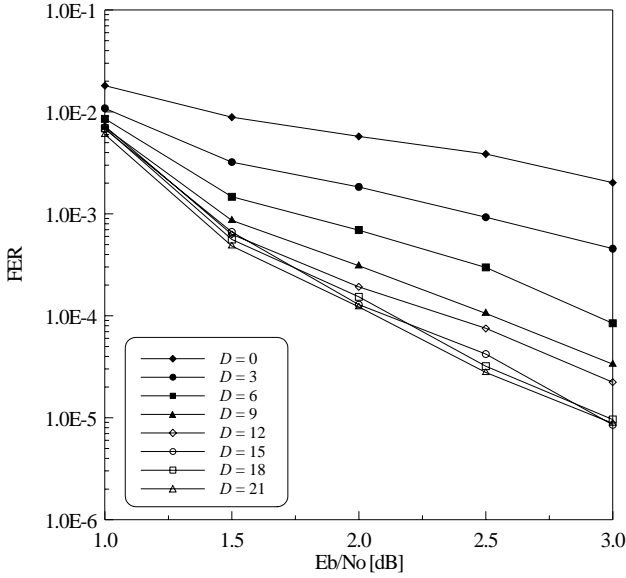

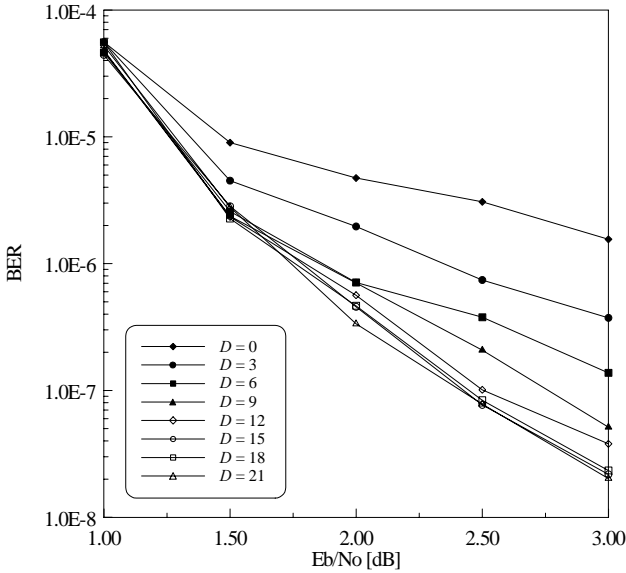Fig. 2 Frame error rate of CRC-turbo concatenated code.

Fig. 3 Bit error rate of CRC-turbo concatenated code.

Also BER of CRC-turbo concatenated code is similar to FER. BER is greatly decreased as D increases. However BER is not decreased any more like FER if D is larger than 15.

Error distribution according to various D-parameters is shown in Table 1, where ε_i represents the ratio of information bit errors to total errors and ε_c represents CRC bit errors to total errors respectively. The smaller D, the more errors are concentrated at CRC bits. Furthermore almost all errors appear in CRC bits at high E_b/N_0. However the errors in CRC bits are significantly reduced when D is large.

TABLE I
ERROR DISTRIBUTION ACCORDING TO D (%)

D / E_b/N_0 [dB]	0		6		12		18	
	ε_i	ε_c	ε_i	ε_c	ε_i	ε_c	ε_i	ε_c
1.0	82	18	96	4	97	3	98	2
1.5	28	72	73	27	96	4	98	2
2.0	8	92	58	42	97	3	96	4
2.5	4	96	41	59	93	7	88	12
3.0	1	99	35	65	91	9	100	0

The average number of iterations according to D is shown in Table 2. The average number of iterations of the proposed interleaver is reduced slightly compared to that of the conventional interleaver with the edge effect. If it is compared to the conventional turbo code without CRC where the number of iteration is fixed to 8, the average number of iterations is significantly reduced.

TABLE II
THE AVERAGE NUMBER OF ITERATIONS

D / E_b/N_0 [dB]	0	6	12	18	Turbo code without CRC
1.0	3.69	3.64	3.64	3.64	8
1.5	2.67	2.63	2.63	2.63	8
2.0	2.15	2.12	2.11	2.11	8
2.5	1.96	1.94	1.94	1.94	8
3.0	1.62	1.61	1.61	1.61	8

IV. CONCLUSIONS

A method to design a swap interleaver to eliminate the edge effect in CRC-turbo concatenated code is proposed and the performance is analyzed by computer simulation in this paper. As results of simulations, frame and bit error rate can be reduced significantly by adopting D-parameter, which is determined by the required minimum free distance of a code. In addition, the average number of iterations is also reduced slightly.

Furthermore, CRC error detection system coupled with automatic repeat request (ARQ) can be used with turbo

code [12]. If the proposed interleaver is used in this hybrid ARQ system the number of retransmissions is expected to be reduced due to the greatly reduced FER.

REFERENCES

[1] C. Berrou, A. Glavieux, and P. Thitimasjshima, "Near Shannon limit error-correcting coding and decoding: turbo-codes," *Proc. IEEE Int. Conf. Commun.*, Geneva, Switzerland, vol. 20, pp. 1064-1070, May 1993.

[2] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261-1271, Oct. 1996.

[3] A. Shibutani, H. Suda, F. Adachi, "Reducing average number of turbo decoding iterations," *Electron. Lett.,* vol. 35, no. 9, pp. 701-702, 1999.

[4] P. Robertson, "Illuminating the structure of parallel concatenated recursive systematic (turbo) codes," *Proc. IEEE Globecom,* San Francisco, USA, pp. 1298-1303, Dec. 1994.

[5] L.C. Perez, J. Seghers, and D.J. Costello, "A distance spectrum interpretation of turbo codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 6, pp. 1698-1709, Nov. 1996.

[6] B.G. Lee, S.J. Bae, S.G. Kang, and E.K. Joo, "Design of swap interleaver for turbo codes," *Electron Lett.,* vol. 35, no. 22, pp. 1939-1940, Oct. 1999.

[7] B.G. Lee, S.J. Bae, C.K. Jeong, and E.K. Joo, "Performance Analysis of Swap Interleaver for Turbo Codes," *Proc. ICT 2001*, vol. 2, pp. 381-384, Bucuresti, Romania, June 2001.

[8] S. Dolinar and D. Divsalar, "Weight distributions for turbo codes using random and nonrandom permutations," *TDA Progress report* 42-122, JPL, pp. 56-65, Aug. 1995.

[9] L.R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear code for minizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. 20, no. 2, pp. 284-287, Mar. 1974.

[10] S.S. Pietrobon and A.S. Barbulescu, "A simplification of the modified Bahl decoding algorithm for systematic convolutional codes," *Proc. IEEE ISITA*, Sydney, Australia, pp. 1073-1077, Nov. 1994.

[11] P. Robertson, E. Villerun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in log domain," *Proc. IEEE Int. Conf. Commun.*, Seattle, USA, pp. 1009-1013, June 1995.

[12] K.R. Narayanan and G.L. Stüber, "A novel ARQ technique using the turbo coding principle," *IEEE Commun. Lett.,* vol. 1. no. 2, pp.49-51, 1997.