The Effect of Block Interleaving in an LDPC-Turbo Concatenated Code

Sang Hoon Lee and Eon Kyeong Joo

ABSTRACT—The effect of block interleaving in a low density parity check (LDPC)-turbo concatenated code is investigated in this letter. Soft decoding can be used in an LDPC code unlike the conventional Reed-Solomon (RS) code. Thus, an LDPC-turbo concatenated code can show better performance than the conventional RS-turbo concatenated code. Furthermore, the performance of an LDPC-turbo code can be improved by using a block interleaver between the LDPC and turbo code. The average number of iterations in LDPC decoding can also be reduced by a block interleaver.

Keywords—Block interleaver, turbo code, LDPC code, concatenated code.

I. Introduction

Turbo codes [1] show good performance due to iterative decoding. But it is difficult to get a very low bit error rate (BER) with only a turbo code because of its error floor phenomenon [2]. Concatenation with another code as an outer code is an alternative that improves the performance of a turbo code. The Reed-Solomon (RS) code has been used as an outer code with a block interleaver [3]. Conventional hard-decision decoding is used for the RS code. The turbo decoder output, however, is a soft log-likelihood ratio (LLR) value; thus, a soft decoding algorithm is efficient for the outer decoder using the soft output of the inner turbo decoder. The soft RS decoding algorithm was proposed in [4]; however, its computational complexity is very high as compared to hard RS decoding. For this reason, it is difficult to adopt the soft RS decoding algorithm in practice.

Gallager's LDPC code [5], which was rediscovered by Mackey [6], approaches near the Shannon limit with relatively low decoding complexity. The decoding algorithm uses soft reliability information and is very simple compared to the turbo code's maximum *a posteriori* (MAP) algorithm. Performance can be improved if an LDPC code is used as an outer code with an inner turbo code, owing to the soft decoding of both codes.

A very low BER will be required in future communication systems which should provide high-quality multimedia services, therefore, a more powerful error correcting code is needed. The LDPC-turbo code, which is composed of an outer LDPC code and an inner turbo code, may be a good choice to achieve a very low BER.

The concatenated LDPC-turbo code, as a scheme to improve the performance of turbo codes, was introduced by Y. Kou and others [7]. They used a long LDPC (65520, 61425) code and a turbo code which has Hamming (64, 57) codes with distance-4 as component codes; however, they did not use an interleaver between the LDPC and turbo codes. An interleaver is reported to be unnecessary if the LDPC code is concatenated with another LDPC code due to the randomness of its parity check matrix [8]. A 2-dimensional block interleaver, however, may be required if the LDPC code is concatenated with an inner turbo code, since concentrated errors may occur in some codewords in the output of a turbo decoder [9]. The number of codewords which have concentrated errors can be reduced by spreading the errors to other codewords by block interleaving. Therefore, the effect of block interleaving between the outer LDPC and the inner turbo code is investigated in this letter. The performance of the LDPC-turbo code is compared with the conventional RS-turbo code with the same block interleaver.

II. Serial Concatenation of the LDPC and Turbo Code with a Block Interleaver

A serially concatenated LDPC-turbo code is presented in Fig. 1.

Manuscript received Apr. 10, 2006; revised July 10, 2006.

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD) (R05-2004-000-10677-0).

Sang Hoon Lee (phone: +82 53 940 8642, email: generalsh@ee.knu.ac.kr) and Eon Kyeong Joo (email: ekjoo@ee.knu.ac.kr) are with the School of Electrical Engineering and Computer Science, Kyungpook National University, Daegu, Korea.

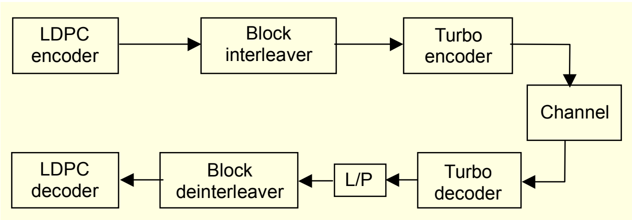

Fig. 1. Serial concatenation of the LDPC and turbo codes.

A block interleaver is used between the LDPC and the turbo encoder. This block interleaver has d rows and n columns. The encoded d codewords, each with a length of n bits, are written row-wise and read column-wise with depth d. This 2-dimensional block interleaver can improve the performance of the concatenated code. The number of codewords having a large number of errors can be reduced by spreading the concentrated errors in some codewords to other codewords by the 2-dimensional block interleaver [3]. The decoding performance can also be improved as depth d is increased.

The input k-bit sequence $\mathbf{x}=(x_1, x_2, \cdots, x_k)$ is encoded to an n-bit sequence $\mathbf{y}=(y_1, y_2, \cdots, y_n)$ by the LDPC encoder. Then, it is interleaved by the block interleaver and re-encoded to an l-bit sequence $\mathbf{z}=(z_1, z_2, \cdots, z_l)$ by a turbo encoder. The encoded bit sequence is modulated and transmitted to a channel. It is received as $\mathbf{z}^r=(z_1^r, z_2^r, \cdots, z_l^r)$ in the turbo decoder. The LLR of a bit y_i is obtained by (1) in the turbo decoder. The bit probability of y_i, as the initial value, however, is needed for the iterative decoding [6] in the LDPC decoder. The initial probability can be obtained by (2) in the L/P (LLR-to-probability) block in Fig. 1. Here, $p^1(y_i)$ is the probability that the bit y_i is 1.

$$L(y_i)=\ln\left\{\frac{\Pr(y_i=1\,|\,\mathbf{z}^r)}{\Pr(y_i=0\,|\,\mathbf{z}^r)}\right\}, \tag{1}$$

$$p^1(y_i)=1-\frac{1}{1+\exp\{L(y_i)\}}. \tag{2}$$

The LDPC decoder receives all initial probabilities of a codeword that are deinterleaved by the block deinterleaver; then $\hat{\mathbf{y}}$ is obtained by hard-decision of the probabilities. If the parity check condition is satisfied, that is $\mathbf{H}\hat{\mathbf{y}}^T=\mathbf{0}$, the decoding process is finished. Here, $\mathbf{H}$ is the parity check matrix of the LDPC code. Otherwise, the iterative LDPC decoding continues until the pre-determined maximum number of iterations. If the parity check condition, however, is satisfied at any iteration step, the whole decoding is finished. The decoded information bit sequence $\hat{\mathbf{x}}$ is obtained from the decoded codeword $\hat{\mathbf{y}}$.

III. The Performance of the LDPC-Turbo Code

In this section, the performance of the LDPC-turbo concatenated code is compared with that of a single LDPC and a turbo code. In addition, it is also compared with that of the RS-turbo concatenated code. The performance comparison is conducted under the same code rate and input frame size.

The turbo code of rate 1/3 is used as the inner code in both the LDPC-turbo and the RS-turbo code. It is composed of the recursive systematic convolutional code with the generator polynomial of $(7, 5)_8$. The number of iterations of the inner turbo code is set to 5 since the performance improvement is almost saturated after 5 iterations. The S-random interleaver [10] with the size of 2040 bits is used in the inner turbo code. An irregular LDPC (2040, 1784) code of rate 0.875 is used as the outer code in the LDPC-turbo code. The maximum number of iterations of the LDPC code is set at 100. The RS (255, 223) code is used as the outer code in the RS-turbo code to match the same code rate of 0.875 of the outer LDPC code. This RS code is composed of bytes, so 223 input bytes correspond to the same 1784 bits of the outer LDPC code. This 1784-bit input frame is used in the single LDPC and in the turbo code too. The overall code rate is 0.292 for both the LDPC-turbo and RS-turbo codes.

The code rates of the single turbo and LDPC codes are also set at the same value of 0.292. The turbo code of rate 0.292 with a 1784-bit input frame is obtained by puncturing the code of rate 1/4. The generating polynomial and the number of iterations of the single turbo code are the same as those of the inner turbo code. An irregular LDPC (6117, 1784) code of rate

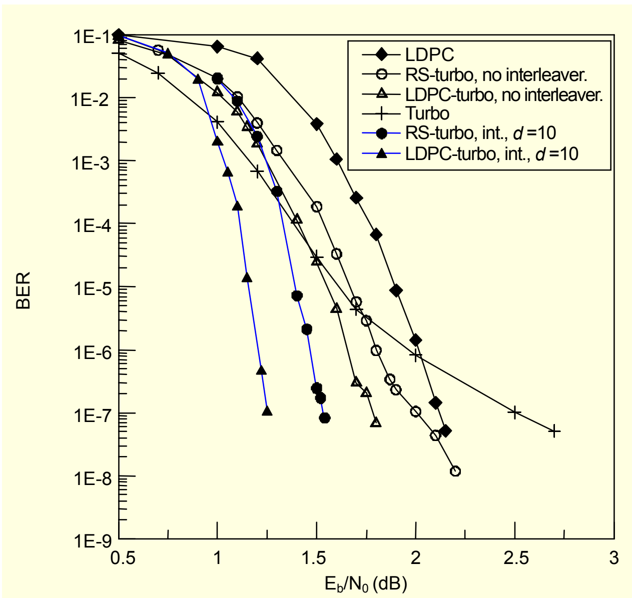

Fig. 2. Performance comparison of the LDPC, turbo, RS-turbo, and LDPC-turbo codes with the same code rate of 0.292 and 1784-bit input frame.

0.292 is used as the single LDPC code. The maximum number of iterations is also set at 100. An additive white Gaussian noise (AWGN) channel is assumed, and binary phase shift keying (BPSK) is used.

The results are shown in Fig. 2. The turbo code performs better than the LDPC code until 2.05 dB. It shows an error floor phenomenon, however, as E_b/N_0 increases. Thus, the LDPC code outperforms the turbo code over 2.05 dB. The LDPC-turbo code performs much better than the single LDPC code of the same code rate. The single LDPC code needs about 2.0 dB to achieve a BER of 10^{-6}, but the LDPC-turbo code needs 1.6 dB without the block interleaver and only 1.2 dB with the block interleaver. Here, the interleaver depth *d* is 10. The LDPC-turbo code without an interleaver also performs better than the single turbo code over 1.5 dB. The required E_b/N_0 to outperform the turbo code is reduced to 1.0 dB with the block interleaver. It is also found that the LDPC-turbo code performs better than the RS-turbo code with and without the block interleaver in the whole E_b/N_0 region. In addition, the difference in performance between the LDPC-turbo and the RS-turbo code is increased by the block interleaver. For example, the performance difference is about 0.2 dB without the interleaver and 0.4 dB with the block interleaver at a BER of 10^{-6}.

IV. The Effect of Block Interleaving

In Fig. 3, the relative performance of the LDPC-turbo and the RS-turbo codes is compared taking into account the various interleaver depths. The LDPC-turbo code performs better than the RS-turbo code with the same interleaver depth. The interleaving gain increases as the depth increases from 10 to 100.

A block interleaver with a large depth can spread out the concentrated errors effectively. This results in reduction in the number of codewords having a large number of errors at the output of the turbo decoder.

The codewords which have a large number of errors are checked at the output of the turbo decoder by hard-decision of LLRs in the LDPC-turbo decoder. The results are shown in Table 1. The number of codewords out of a total of 500 with more than 30-bit errors is observed. The codeword length *n* is 2040 bits. If a block interleaver is not used between the LDPC and the turbo code, 123 codewords having more than 30-bit errors are observed at 1.10 dB. This is reduced to 117, however, by the block interleaver with *d* = 10, and it is reduced further as *d* and E_b/N_0 are increased.

The LDPC decoder can receive codewords with a small number of errors from a large interleaver depth. A small number of errors in a codeword can lead to a reduction of the number of iterations in the outer LDPC decoding. The average number of iterations in the outer LDPC decoder is shown in Table 2. When an interleaver is not used, the average number

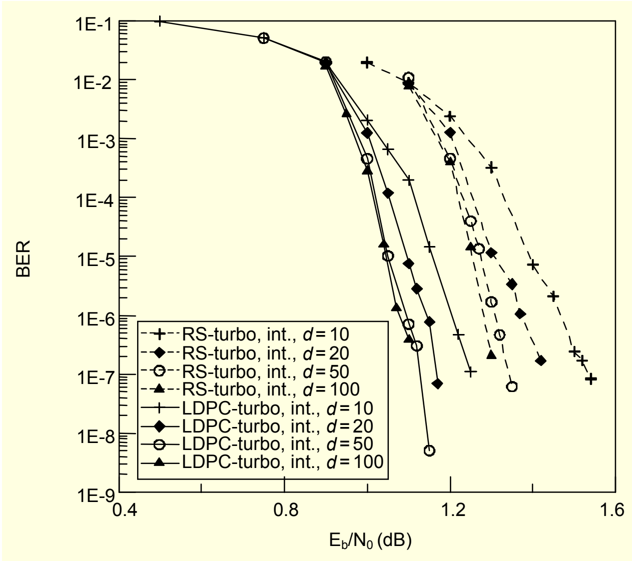

Fig. 3. Performance comparison of the LDPC-turbo and RS-turbo code with various interleaver depths.

Table 1. The number of codewords out of a total of 500 having more than 30-bit errors at the output of the turbo decoder in the LDPC-turbo code.

E_b/N_0	The number of codewords with more than 30-bit errors				
	No interleaver	Block interleaver			
		d = 10	*d* = 20	*d* = 50	*d* = 100
1.10 dB	123	117	63	22	22
1.15 dB	78	45	25	17	0
1.20 dB	38	22	0	0	0

Table 2. The average number of iterations of the LDPC decoding in the LDPC-turbo code.

E_b/N_0	The average number of iterations in the LDPC decoding				
	No interleaver	Block interleaver			
		d = 10	*d* = 20	*d* = 50	*d* = 100
1.10 dB	19.6	3.9	2.6	2.3	2.3
1.15 dB	11.0	2.1	1.9	1.7	1.5
1.20 dB	6.1	1.6	1.5	1.5	1.5

of iterations is 19.6 at 1.10 dB. This is reduced to 3.9 by a block interleaver with a depth *d* = 10. The number of iterations is reduced further as *d* and E_b/N_0 are increased.

V. Performance Analysis on Fading Environment.

In order to investigate the performance of the LDPC-turbo code in a mobile environment, a performance comparison is conducted on a Rayleigh fading channel based on a modified Jakes model [11]. The compared codes are the same as in the previous simulation. A source rate of 64 kbps and a carrier frequency of 2 GHz are assumed. A mobile speed of 60 km/h is considered. The selected block interleaver depth is 50. A channel interleaver is not used in this simulation to exclude the effect of a channel interleaver. The simulation results are shown in Fig. 4.

The LDPC-turbo code with a block interleaver shows the best performance in comparison with other codes. The RS-turbo code shows worse performance than the LDPC-turbo code with and without a block interleaver as in an AWGN channel. However, in some points this case differs from an AWGN channel. The single LDPC code shows better performance than the single turbo code unlike the case of an AWGN channel. This may be due to the interleaving effect in the parity-check matrix of LDPC code [12]. The LDPC-turbo code without a block interleaver needs 12.5 dB to achieve a BER of 10^{-6}, but it is reduced significantly to 5.9 dB by a block interleaver.

The average number of iterations in the concatenated LDPC code is investigated at 5.9 dB. Without a block interleaver, 20.4 iterations are required, but it is reduced to 3.3 iterations by the block interleaver in our simulation. Almost the same results are obtained at other mobile speeds. Thus, it can be said that a block interleaver between the LDPC and turbo codes leads not only to performance improvement but also to a reduction in the number of iterations in LDPC decoding.

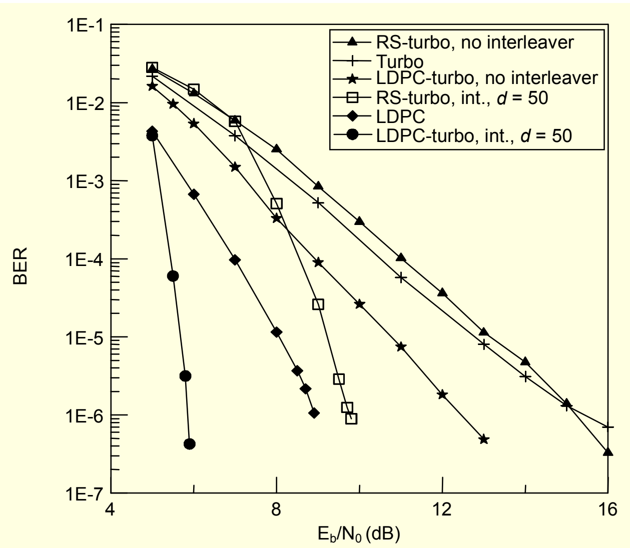

Fig. 4. Performance comparison of the LDPC-turbo code with other codes on fading channel.

VI. Conclusion

In this letter, the effect of block interleaving on the LDPC-turbo concatenated code was investigated. From the results of our investigation, we conclude that the LDPC-turbo code performs better than the single turbo code and the LDPC code; moreover, it outperforms the RS-turbo code. Our study demonstrates that performance can be improved with the use of a 2-dimensional block interleaver, since it can spread out the concentrated errors at the output of the turbo decoder effectively. We observed the number of iterations in the outer LDPC decoding to be very small in the low BER region. The number of iterations can be reduced further as the interleaver depth increases. Therefore, we conclude that the LDPC-turbo concatenated code with a block interleaver is suitable for future high-quality communication systems.

References

[1] C. Berrou, A. Glavieux, and P. Thitimasjshma, "Near Optimum Error Correcting Coding and Decoding: Turbo-Codes," *Proc. IEEE ICC'93*, Geneva, Switzerland, vol. 20, May 1993, pp. 1064-1070.

[2] L.C. Perez, J. Seghers, and D.J. Costello, Jr. "A Distance Spectrum Interpretation of Turbo Codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 6, Nov. 1996, pp. 1698-1709.

[3] M. Ferrari, F. Osnato, M. Siti, S. Valle, and S. Bellini, "Performance of Concatenated Reed-Solomon and Turbo Codes with Nonideal Interleaving," *IEEE GLOBECOM'01*, San Antonio, Texas, Nov. 2001, pp. 911-915.

[4] R. Koetter and A. Vardy, "Algebraic Soft-Decision Decoding of Reed-Solomon Codes," *Proc. IEEE ISIT 2000*, Sorrento, Italy, June 2000, p. 61.

[5] R.G. Gallager, "Low-Density Parity-Check Code," *IRE Trans. Inform. Theory*, vol. 8, Jan. 1962, pp. 21-28.

[6] D.J.C. Mackay and R.M. Neal, "Near Shannon Limit Performance of Low Density Parity Check Codes," *Electron. Lett.*, vol. 33, no. 6, Mar. 1997, pp. 457-458.

[7] Y. Kou, S. Lin, and M.P.C. Fossorier, "Low-Density Parity-Check Codes Based on Finite Geometrics: A Rediscovery and New Results," *IEEE Trans. Inform. Theory*, vol. 47, no. 7, Nov. 2001, pp. 2711-2736.

[8] H. Behairy and S.C. Chang, "Parallel Concatenated Gallager Codes for CDMA Applications," *Proc. IEEE MILCOM 2001*, Vienna, Virginia, vol. 2, Nov. 2001, pp. 1002-1006.

[9] O.Y. Takeshita, O.M. Collins, P.C. Massey, and D.J. Costello, "On the Frame-Error Rate of Concatenated Turbo Codes," *IEEE Trans. Commun.* vol. 49, no. 4, Apr. 2000, pp. 602-608.

[10] S. Dolinar and D. Divsalar, "Weight Distributions for Turbo Codes Using Random and Non-random Permutations," *TDA Progress Report*, JPL, Pasadena, CA, vol. 42, Aug. 1995, pp. 56-65.

[11] P. Dent, G. E. Bottomley, and T. Croft, "Jakes Fading Model Revisited," *IEE Electron. Lett.*, vol. 29, no. 13, June 1993, pp. 1162-1163.

[12] J. Hou, P.H. Siegel, and L.B. Milstein, "Performance Analysis and Code Optimization of Low Density Parity-Check Codes on Rayleigh Fading Channels," *IEEE J. Select. Areas Commun.*, vol. 19. no. 5, May 2001, pp. 924-934.