논문 14-39A-10-04 The Journal of Korea Information and Communications Society '14-10 Vol.39A No.10
http://dx.doi.org/10.7840/kics.2014.39A.10.585

이진 순환 부호를 쓰는 GLDPC 부호의 수평-수직 결합 직렬 복호

정 규 혁◆°

Combined Horizontal-Vertical Serial BP Decoding of GLDPC Codes with Binary Cyclic Codes

Kyuhyuk Chung◆°

ABSTRACT

It is well known that serial belief propagation (BP) decoding for low-density parity-check (LDPC) codes achieves faster convergence without any increase of decoding complexity per iteration and bit error rate (BER) performance loss than standard parallel BP (PBP) decoding. Serial BP (SBP) decoding, such as horizontal SBP (H-SBP) decoding or vertical SBP (V-SBP) decoding, updates check nodes or variable nodes faster than standard PBP decoding within a single iteration.

In this paper, we propose combined horizontal-vertical SBP (CHV-SBP) decoding. By the same reasoning, CHV-SBP decoding updates check nodes or variable nodes faster than SBP decoding within a serialized step in an iteration. CHV-SBP decoding achieves faster convergence than H-SBP or V-SBP decoding. We compare these decoding schemes in details. We also show in simulations that the convergence rate, in iterations, for CHV-SBP decoding is about $\frac{1}{6}$ of that for standard PBP decoding, while the convergence rate for SBP decoding is about $\frac{1}{2}$ of that for standard PBP decoding. In simulations, we use recently proposed generalized LDPC (GLDPC) codes with binary cyclic codes (BCC).

Key Words : BP decoding, LDPC codes, iterative decoding, GLDPC codes, convergence

Ⅰ. Introduction

Low-density parity-check (LDPC) codes[1,2], have received significant research attention. The LDPC codes can achieve the performance near the Shannon capacity on an additive white Gaussian noise (AWGN) channel[3,4]. The LDPC codes are usually decoded by the standard parallel belief propagation (BP) decoding[5]. In each iteration, the parallel BP (PBP) decoding scheme updates in parallel all the check nodes and then updates all the variable nodes iteratively. In order to take advantage of more reliable extrinsic messages updated within a single iteration, some serial BP (SBP) decoding schemes are introduced in [6-12]. The vertical SBP (V-SBP) decoding schemes are presented in [6-9]. On the other hand, the horizontal SBP (H-SBP) decoding schemes are addressed in [8, 9, 12]. These H-SBP and V-SBP decoding schemes can converge roughly twice as fast as the standard PBP decoding scheme

◆° First and Corresponding Author : Dankook University Department Of Software Science, khchung@dankook.ac.kr, 종신회원
논문번호 : KICS2014-09-354, Received September 18, 2014; Revised October 6, 2014; Accepted October 6, 2014

without any increase of decoding complexity per iteration and bit error rate (BER) performance loss.

In this paper, we present combined horizontal-vertical SBP (CHV-SBP) decoding. The H-SBP and V-SBP decoding schemes take advantage of more reliable extrinsic messages updated within a single iteration, so that they achieve faster convergence than the PBP decoding scheme. By the same reasoning, the CHV-SBP decoding scheme uses more reliable extrinsic messages updated within a single *serialized step* in an iteration. In result, the convergence rate, in iterations, for CHV-SBP decoding is about $\frac{1}{6}$ of that for standard PBP decoding, while the convergence rate for the H-SBP or V-SBP decoding schemesis about $\frac{1}{2}$ of that for standard PBP decoding. Obviously, the convergence rate for CHV-SBP decoding is about $\frac{1}{3}$ of that for H-SBP or V-SBP decoding. The CHV-SBP decoding scheme proposed in this paper is intended to speed up the H-SBP or V-SBP decoding schemes at no cost in complexity without BER performance loss. Thus, for simulations, we use recently proposed generalized LDPC (GLDPC) codes with binary cyclic codes (BCC)[13]. Various GLDPC codes achieve better performance at the cost of higher decoding complexity for moderate block lengths[14-17]. They are generated by replacing the parity check equations in a parity-check matrix of an LDPC code with many kinds of short length component codes. The examples of component codes are Hamming component codes[14], Reed-Muller component codes[15], and BCH and RS component codes[16]. Recently, GLDPC codes with BCC as component codes are reported[13]. The GLDPC codes with BCC have roughly the same decoding complexity as the LDPC codes, while keeping better performance than the LDPC codes like other GLDPC codes.

This paper is organized as follows. Section II describes reviews of the PBP, H-SBP, and V-SBP decoding schemes. In section III, we derive the proposed CHV-SBP decoding scheme. Section IV presents simulation results and discussions. Section V concludes the paper.

Ⅱ. Reviews of PBP, H-SBP, and V-SBP Decoding Schemes

In this section, first a review of the PBP decoding scheme is given and later reviews of the H-SBP and V-SBP decoding schemes are provided. Assume that a codeword block $\mathbf{c} \triangleq [c_0, c_1, \cdots, c_{N-1}]$ with a codeword block length N is transmitted over an AWGN channel with zero mean and variance $N_0/2$. Based on [2], we define an $M \times N$ parity-check matrix $\mathbf{H} \triangleq [H_{mn}]$ with the number of checks M, where $0 \le n \le N$-1, and $0 \le m \le M$-1. Note that in this paper, we label parity check code constraints in LDPC codes and various component code constraints in GLDPC codes as *checks* for simplicity and generalization without loss of generality. Let C_n be the log-likelihood ratio (LLR) of bit n, which is derived from the observed channel output y_n. Define $C_{mn}^{(i)}$ and $V_{mn}^{(i)}$ associated with the i^{th} iteration as the LLRs of bit n which are passed from check node m to bit node n and passed from bit node n to check node m, respectively. Let $V_n^{(i)}$ be the pseudo posterior LLR of bit n. We denote the set of bits that participate in check m by $\mathcal{N}(m) \triangleq \{n : H_{mn} = 1\}$ and the set of checks in which bit n participates by $\mathcal{M}(n) \triangleq \{m : H_{mn} = 1\}$. We also denote a set $\mathcal{N}(m)$ with bit n excluded by $\mathcal{N}(m) \backslash n$, and a set $\mathcal{M}(n)$ with check m excluded by $\mathcal{M}(n) \backslash m$. We use the operation notation Ψ for computing $C_{mn}^{(i)}$ for generalization. The operation Ψ can be analytical expressions or dynamic programming. This notation is helpful to applying the decoding schemes of LDPC codes to those of GLDPC codes and explaining clearly the SBP decoding schemes. For example, in LDPC codes, the tanh-rule is usually used, which is given by

$$\tanh \frac{C_{mn}^{(i+1)}}{2} = \prod_{n' \in \mathcal{N}(m) \backslash n} \tanh \frac{V_{mn'}^{(i)}}{2}. \qquad (1)$$

This is further simplified as

$$C_{mn}^{(i+1)} = 2\tanh^{-1}\left(\prod_{n'\in\mathcal{N}(m)\backslash n}\tanh\frac{V_{mn'}^{(i)}}{2}\right) = \log\frac{1+\prod_{n'\in\mathcal{N}(m)\backslash n}\tanh\frac{V_{mn'}^{(i)}}{2}}{1-\prod_{n'\in\mathcal{N}(m)\backslash n}\tanh\frac{V_{mn'}^{(i)}}{2}}. \quad (2)$$

Then the operation Ψ is defined by

$$\Psi_{n'\in\mathcal{N}(m)\backslash n}\left(V_{mn'}^{(i)}\right) \triangleq \log\frac{1+\prod_{n'\in\mathcal{N}(m)\backslash n}\tanh\frac{V_{mn'}^{(i)}}{2}}{1-\prod_{n'\in\mathcal{N}(m)\backslash n}\tanh\frac{V_{mn'}^{(i)}}{2}}. \quad (3)$$

For another example, in GLDPC codes with BCC, the operation Ψ is defined by using Bahl-Cocke-Jelinek-Raviv (BCJR) dynamic programming algorithms[17], which can be computed efficiently as in [13]. The standard PBP decoding scheme is carried out as follows:

■ PBP decoding scheme

Initialization:

Set i =1, and the maximum number of iterations to $I_{\max}$. For each m, n, set $V_{mn}^{(1)} = C_n$.

Horizontal step:

For $0 \le m \le M$-1 and each $n \in \mathcal{N}(m)$, compute

$$C_{mn}^{(i+1)} = \Psi_{n'\in\mathcal{N}(m)\backslash n}\left(V_{mn'}^{(i)}\right). \quad (4)$$

Vertical step:

For $0 \le m \le N$-1 and each $m \in \mathcal{M}(n)$, compute

$$V_{mn}^{(i+1)} = C_n + \sum_{m'\in\mathcal{M}(n)\backslash m} C_{m'n}^{(i+1)}, \quad (5)$$

$$V_n^{(i)} = C_n + \sum_{m'\in\mathcal{M}(n)} C_{m'n}^{(i+1)}. \quad (6)$$

Iteration stopping criterion:

Make hard decisions on $V_n^{(i)}$, and create a tentative codeword $\hat{\mathbf{c}} = [\hat{c}_0, \hat{c}_1, \cdots, \hat{c}_{N-1}]$.

If all the checks are satisfied with $\hat{\mathbf{c}}$ or $i = I_{\max}$, stop the decoding iteration with the decoded codeword $\hat{\mathbf{c}}$.

Otherwise, go to **Horizontal step** with $i = i + 1$. ■

Next, the V-SBP decoding scheme is reviewed, based on [6], [7]. In this paper, we introduce *serialized steps* in an iteration for clarifying the SBP decoding schemes and comparing various SBP decoding schemes. For the V-SBP decoding scheme, the serialized step is equal to n. However, the serialized step of the H-SBP decoding scheme is different from that of the V-SBP decoding scheme. This is considered later. For the V-SBP decoding scheme, interpret $V_{mn}^{(i)}$ as the LLRs of bit n which are passed from bit node n to check node m associated with the n^{th}serialized step in the i^{th} iteration. The V-SBPdecoding scheme is carried out as follows:

■ V-SBP decoding scheme

Initialization:

Set i =1, and the maximum number of iterations to $I_{\max}$. For each m, n, set $V_{mn}^{(1)} = C_n$.

Iteration:

For $0 \le m \le N$-1.

Serialized step n:

Horizontal step:

Each $m \in \mathcal{M}(n)$, compute

$$C_{mn}^{(i+1)} = \Psi_{n'\in\mathcal{N}(m)\backslash n}\left(V_{mn'}^{(i)}\right). \quad (7)$$

Vertical step:

Each $m \in \mathcal{M}(n)$, compute

$$V_{mn}^{(i)} = C_n + \sum_{m' \in \mathcal{M}(n)\backslash m} C_{m'n}^{(i+1)}, \tag{8}$$

$$V_{n}^{(i)} = C_n + \sum_{m' \in \mathcal{M}(n)} C_{m'n}^{(i+1)}. \tag{9}$$

Iteration stopping criterion:

Make hard decisions on $V_n^{(i)}$, and create a tentative codeword $\hat{\mathbf{c}} = [\hat{c}_0, \hat{c}_1, \cdots, \hat{c}_{N-1}]$.
If all the checks are satisfied with $\hat{\mathbf{c}}$ or $i = I_{\max}$, stop the decoding iteration with the decoded codeword $\hat{\mathbf{c}}$.

Otherwise, go to **Iteration** with $V_{mn}^{(i+1)} = V_{mn}^{(i)}$ and $i = i + 1$. ■

In the equation (8), $V_{mn}^{(i)}$ is used instead of $V_{mn}^{(i+1)}$. This simplifies considerably the equation (7).

Lastly, the H-SBP decoding scheme is reviewed, based on [10], [11]. For the H-SBP decoding scheme, the serialized step is equal to m and $V_{mn}^{(i)}$ is interpreted as the LLRs of bit n which are passed from bit node n to check node m associated with the m^{th} serialized step in the i^{th} iteration. The H-SBP decoding scheme is carried out as follows:

■ **H-SBP decoding scheme**

Initialization:

Set $i = 1$, and the maximum number of iterations to $I_{\max}$. For each n, set $V_n^{(1)} = C_n$. For each m, n, set $C_{mn}^{(1)} = 0$.

Iteration:

For $0 \le m \le M$-1.

Serialized step m:

Horizontal step:

Each $n \in \mathcal{N}(m)$, compute

$$V_{mn}^{(i)} = V_n^{(i)} - C_{mn}^{(i)}, \tag{10}$$

$$C_{mn}^{(i+1)} = \underset{n' \in \mathcal{N}(m)\backslash n}{\Psi} \left(V_{mn'}^{(i)} \right). \tag{11}$$

Vertical step:

Each $n \in \mathcal{N}(m)$, compute

$$V_n^{(i)} = V_{mn}^{(i)} + C_{mn}^{(i+1)}. \tag{12}$$

Iteration stopping criterion:

Make hard decisions on $V_n^{(i)}$, and create a tentative codeword $\hat{\mathbf{c}} = [\hat{c}_0, \hat{c}_1, \cdots, \hat{c}_{N-1}]$.

If all the checks are satisfied with $\hat{\mathbf{c}}$ or $i = I_{\max}$, stop the decoding iteration with the decoded codeword $\hat{\mathbf{c}}$.

Otherwise, go to **Iteration** with $V_n^{(i+1)} = V_n^{(i)}$ and $i = i + 1$. ■

Ⅲ. Derivation of CHV-SBP Decoding Schemes

The CHV-SBP decoding scheme combines the H-SBP and V-SBP decoding schemes. For the CHV-SBP decoding scheme, we define another set $\mathcal{M}^{\text{ordered}}(n) \triangleq \{m_s \mid m_0 < m_1 < m_2 < \cdots < m_{|\mathcal{M}(n)|-1} : H_{mn} = 1\}$. The H-SBP or V-SBP decoding schemes take advantage of more reliable extrinsic messages updated within a single iteration, so that they achieve faster convergence than the PBP decoding scheme. By the same reasoning, the CHV-SBP decoding scheme uses more reliable extrinsic messages updated in the m_s^{th} serialized step within the n^{th} serialized step in an iteration. For the CHV-SBP decoding scheme, $V_{m_s n}^{(i)}$ is interpreted as the LLRs of bit n which are passed from bit node n to check node m_s associated with the m_s^{th} serialized step within the n^{th} serialized step in the i^{th} iteration. The CHV-SBP decoding scheme is carried out as follows:

■ **CHV-SBP decoding scheme**

Initialization:

Set $i = 1$, and the maximum number of iterations to $I_{\max}$. For each n, set $V_n^{(1)} = C_n$. For each m, n, set $C_{mn}^{(1)} = 0$.

Iteration:

For $0 \le m \le N$-1.

Vertical serialized step n:

$s = 0$

Horizontal serialized step m_s:

If $s = |\mathcal{M}(n)|$, go to **Vertical serialized step** n with $n = n + 1$.

For $m_s \in \mathcal{M}^{\text{ordered}}(n)$.

$m = m_s$.

If all the $C_{mn}^{(i+1)}$ are already computed for each $n \in \mathcal{N}(m)$, go to **Horizontal serialized step m_s** with $s = s + 1$.

Otherwise continue.

Horizontal step:

Each $n_H \in \mathcal{N}(m)$, compute

$$V_{mn_H}^{(i)} = V_{n_H}^{(i)} - C_{mn_H}^{(i)}, \tag{13}$$

$$C_{mn_H}^{(i+1)} = \Psi_{n'_H \in \mathcal{N}(m) \backslash n_H} \left(V_{mn'_H}^{(i)} \right). \tag{14}$$

Vertical step:

Each $n_H \in \mathcal{N}(m)$, compute

$$V_{n_H}^{(i)} = V_{mn_H}^{(i)} + C_{mn_H}^{(i+1)}. \tag{15}$$

Iteration stopping criterion:

Make hard decisions on $V_n^{(i)}$, and create a tentative codeword $\hat{\mathbf{c}} = [\hat{c}_0, \hat{c}_1, \cdots, \hat{c}_{N-1}]$.

If all the checks are satisfied with $\hat{\mathbf{c}}$ or $i = I_{\max}$, stop the decoding iteration with the decoded codeword $\hat{\mathbf{c}}$.

Otherwise, go to **Iteration** with $V_n^{(i+1)} = V_n^{(i)}$ and $i = i + 1$. ■

After four decoding schemes are explained, we mention one thing very carefully for decoding complexity. When we consider decoding complexity, we do not include complexity reduction techniques on hardware implementations. Thus, since the only differences of four decoding schemes are scheduling, the minimum complexity for an iteration is assumed to be the same for all decoding schemes.

Ⅳ. Simulation Results and Discussions

We consider the GLDPC codes with BCC, which were simulated in [13]. Specifically, the codeword block length N is ten thousand and the code rate is 1/2. The maximum allowable number of iterations is $I_{\max} = 100$, for all the decoding schemes. Note that we did not optimize $I_{\max} = 100$ and just observed negligible performance improvemence when $I_{\max} > 100$. The BERs for a binary input AWGN channel are shown as a function of E_b / N_0 (in decibels). Two Things are intensively considered.

The one thing is that we show, in simulations, the convergence rates, in iterations, for the PBP, V-SBP, H-SBP, and CHV-SBP decoding schemes. In Fig. 1, the BER performances of 10, 20, 60 iterations for the PBP decoding schemes are shown. This figure also includes the BER performances of 10, 30 iterations for the V-SBP and H-SBP decoding schemes and the BER performance of 10 iterations for the CHV-SBP decoding scheme. As shown in Fig. 1, the convergence rates of V-SBP, H-SBP, and CHV-SBP decoding are about $\frac{1}{2}$, $\frac{1}{2}$, and $\frac{1}{6}$ of that for PBP decoding, respectively. Therefore,we conjecture that the convergence rate, in iterations, for CHV-SBP

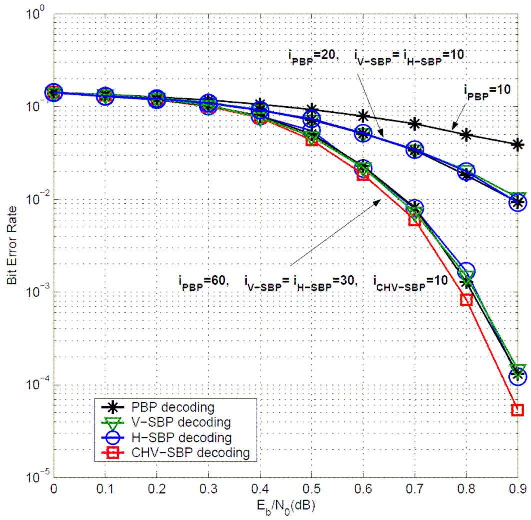

Fig. 1. Comparisons of BERs by PBP, V-SBP, H-SBP and proposed CHV-SBP decoding schemes. The BERs for a binary input AWGN channel are shown as a function of E_b/N_0 (in decibels). (Note that BER curves are given at specified numbers of iterations.)

decoding is about $\frac{1}{6}$ of that for standard PBP decoding. On the other hand, thorough experimental and mathematical studies indicate that the convergence rate, in iterations, for the V-SBP and H-SBP decoding schemes is about $\frac{1}{2}$ of that for the PBP decoding scheme, independent of LDPC and GLDPC codes. Thus, this $\frac{1}{2}$ factor is not the new result, just presented as for comparison purpose. Only the $\frac{1}{6}$ factor is the contribution of this paper. Before we mention the other thing, note that in Fig. 1, I_{max} does not matter because all the BER curves are shown at specific numbers of iterations.

The other thing is the comparisons of average numbers I_{ave} of iterations for the converged BER performance of the four decoding schemes. With the same I_{max} = 100 for all the decoding schemes, in Fig. 2, we plot the four BER curves at $i = I_{max}$ = 100. Fig. 2 shows the approximately converged BER performances. This means that with the same I_{max} = 100, even though average numbers I_{ave} of iterations are significantly different for the four decoding schemes, the BER performances almost reach the maximum performance of the simulated GLDPC code at each SNRs with BP decoding. Then in Fig. 3, we calculate average numbers I_{ave} of iterations of the four decoding schemes. One point should be mentioned. This is that in the low signal-to-noise (SNR) region, up to 0.7 dB, the $\frac{1}{2}$ factor of the convergence rate for V-SBP and H-SBP decoding does not hold. In this low SNR region, BERs are greater than 10^{-3} and in turn the dominant iteration stopping criterion is $i = I_{max}$. In result, the convergence rate factor is greater that the $\frac{1}{2}$ factor. Therefore, the meaningful SNR region is the high SNR region, greater than or equal to 0.8 dB. In high SNR region, the $\frac{1}{2}$ factor roughly holds. By the same reasoning, the convergence rate for CHV-SBP decoding is about $\frac{1}{6}$ of that for standard PBP decoding in the high SNR region, greater than or equal to 0.8 dB. For example, since $1/6 \simeq 0.167$ and $1/7 \simeq 0.142$, based on I_{ave}(CHV-PBP) / I_{ave}(PBP) = 6.5 / 38.5 = 0.168 at 0.8 dB and 4.45 / 31 = 0.143 at 0.9 dB in Fig. 3, we could approximate the

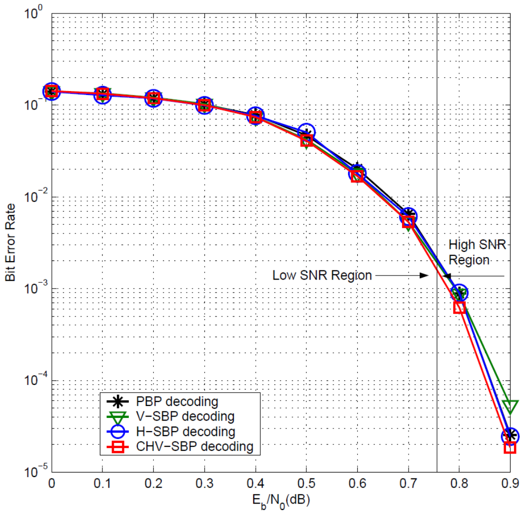

Fig. 2. Comparisons of BERs by PBP, V-SBP, H-SBP and proposed CHV-SBP decoding schemes. The BERs for a binary input AWGN channel areshown as a function of E_b/N_0 (in decibels). (Note that all the BER curves are given at $i = I_{max} = 100$.)

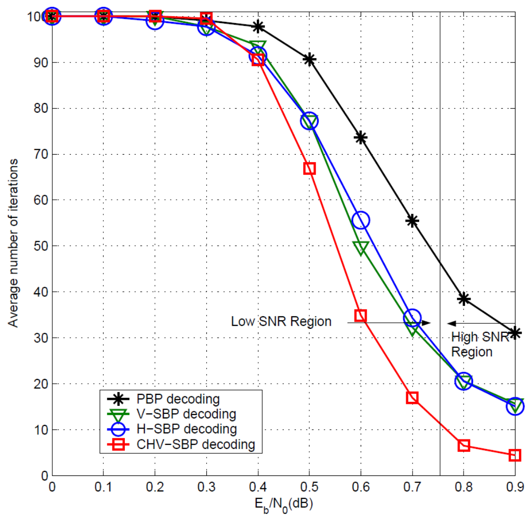

Fig. 3. Comparison of average numbers of iterarions for PBP, V-SBP, H-SBP and proposed CHV-SBP decoding schemes. The average numbers of iterarions are shown as a function of E_b/N_0 (in decibels). (Note that all the curves are simulated with $I_{max} = 100$.)

convergence rate for CHV-SBP decoding as about $\frac{1}{6}$ of that for standard PBP decoding in the high SNR region. This second analysis confirms the first analysis one more time.

V. Conclusion

In this paper, we proposed the CHV-SBP decoding scheme. In simulations, we used recently proposed GLDPC codes with BCC. The CHV-SBP decoding scheme updates check nodes or variable nodes faster than V-SBP and H-SBP decoding within a serialized step in an iteration. In result, CHV-SBP decoding achieves faster convergence than V-SBP and H-SBP decoding. We compared these decoding schemes in details. We also showed in simulations that the convergence rate, in iterations, for CHV-SBP decoding is about $\frac{1}{6}$ of that for standard PBP decoding, while the convergence rate for V-SBP or H-SBP decoding is about $\frac{1}{2}$ of that for standard PBP decoding. In future research, analytical and mathematical studies of the convergence rate for CHV-SBP decoding will be meaningful and interesting.

References

[1] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory,* vol. IT-8, pp. 21-28, Jan. 1962.

[2] D. J. C. MacKay, "Good error correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 399-431, Mar. 1999.

[3] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of capacity-approaching low-density parity-check codes," *IEEE Trans. Inf. Theory,* vol. 47, no. 2, pp. 619-637, 2001.

[4] T. J. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 599-618, 2001.

[5] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference,* San Mateo, Morgan Kaufmann, CA, 1988.

[6] J. Zhang and M. Fossorier, "Shuffled iterative decoding," *IEEE Trans. Commun.*, vol. 53, pp. 209-213, Feb. 2005.

[7] J. Zhang, Y. Wang, and M. Fossorier, "Replica shuffled iterative decoding," in *Proc. IEEE Int. Symp. Inf. Theory*, pp. 454-458, Adelaide, Australia, Sept. 2005.

[8] X. Liu, J. Cai, and L. Wu, "Improved decoding algorithm of serial belief propagation with a stop updating criterion for LDPC codes and applications in patterned media storage," *IEEE Trans. Magnetics*, vol. 49, no. 2, pp. 829-836, Feb. 2013.

[9] Y. Ueng, B. Yang, C. Yang, H. Lee, and J. Yang, "An efficient multi-standard LDPC decoder design using hardware-friendly shuffled decoding," *IEEE Trans. Circuits and Systems I*, vol. 60, no. 3, pp. 743-756, Mar. 2013.

[10] D. Hocevar, "A reduced complexity decoder architecture via layered decoding of LDPC codes," in *Proc. Signal Processing Systems (SIPS)*, pp. 107-112, Austin, Texas, USA, Oct. 2004.

[11] M. M. Mansour and N. R. Shanbhag, "High-throughput LDPC decoders," *IEEE Trans. Very Large Scale Integration Systems(VLSI)*, vol. 11, pp. 976-996, 2003.

[12] Y. Ueng, C. Leong, C. Yang, C. Cheng, K. Liao, and S. Chen, "An efficient layered decoding architecture for nonbinary QC-LDPC codes," *IEEE Trans. Circuits and Systems I*, vol. 59, no. 2, pp. 385-398, Feb. 2012.

[13] K. Chung, "Generalised low-density parity-check codes with binary cyclic codes as component codes," *IET commun.* vol. 6, no. 12, pp. 1710-1715, 2012.

[14] M. Lentmaier and K. Zigangirov, "On generalized low-density paritycheck codes based on Hamming component codes," *IEEE Commun. Lett.*, vol. 3, no. 8, pp. 248-250, Aug. 1999.

[15] I. Djordjevic, L. Xu, T. Wang, and M. Cvijetic, "GLDPC codes with Reed-Muller component codes suitable for optical communications," *IEEE Commun. Lett.*, vol. 12, no. 9, pp. 684-686, Sept. 2008.

[16] N. Miladinovic and M. Fossorier, "Generalized LDPC codes with Reed-Solomon and BCH codes as component codes for binary channels," in *Proc. IEEE GLOBECOM*, p. 6, St. Louis, Missouri, USA, Nov. 2005.

[17] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory,* vol. 20, pp. 284-287, 1974.

정 규 혁 (Kyuhyuk Chung)

1997년 2월 : 성균관대학교 전자공학과 졸업

1998년 12월 : Univ. of Southern California 전기공학 석사

2003년 12월 : Univ. of Southern California 전기공학 박사

1999년 8월~2000년 5월 : 미국 Integrated Device Technology, Inc., Member of Technical Staff

2001년 5월~2002년 5월 : 미국 TrellisWare Technology, Inc., Senior Engineer

2004년 2월~2005년 8월 : LG전자 이동통신기술연구소 표준화그룹 선임연구원

2005년 9월~현재 : 단국대학교 공과대학 소프트웨어학과 부교수

<관심분야> Channel coding, Iterative detection, LDPC codes, Turbo codes, Near-channel capacity approaching codes