Construction of Optimal Concatenated Zigzag Codes Using Density Evolution with a Gaussian Approximation

Song-Nam Hong* ***Associate Member***, Dong-Joon Shin**° ***Lifelong Member***

ABSTRACT

Capacity-approaching codes using iterative decoding have been the main subject of research activities during past decade. Especially, LDPC codes show the best asymptotic performance and density evolution has been used as a powerful technique to analyze and design good LDPC codes. In this paper, we apply density evolution with a Gaussian approximation to the concatenated zigzag (CZZ) codes by considering both flooding and two-way schedulings. Based on this density evolution analysis, the threshold values are computed for various CZZ codes and the optimal structure of CZZ codes for various code rates are obtained. Also, simulation results are provided to confirm the analytical results.

Key Words : LDPC codes, CZZ codes, Density evolution, Scheduling, Threshold.

Ⅰ. Introduction

Low-density parity-check (LDPC) codes were introduced by Gallager[1] and rediscovered by Mackay and Neal[2]. Richardson, Shokrollahi and Urbanke[3] showed that irregular LDPC codes could have better performance on the AWGN channel than turbo codes, which was nearly close to the Shannon limit. However, the encoding complexity of LDPC codes is quadratic in the block length, which results in slow encoding. Therefore, efficient encoding algorithm has been studied[4] and various fast encodable codes such as irregular repeat accumulate (IRA) codes, concatenated tree (CT) codes and concatenated zigzag (CZZ) codes were proposed[5-7].

CZZ codes were proposed by Ping[7], which use zigzag codes having simple tree structure without cycle. The error correcting capability of zigzag code is very weak but it is very useful as a component code to construct concatenated codes. CZZ codes have lower decoding complexity than turbo codes, faster convergence speed than LDPC codes due to the two-way scheduling. Also, they are linear-time encodable.

We denote a CZZ code having Nc zigzag codes and dc check node degree by (Nc, dc)-CZZ code. The overall code rate is (dc - 2)/(dc - 2 +Nc) and more detailed structure can be found in [7]. For a fixed code rate, CZZ code can have various structures by changing the number of zigzag codes Nc and the corresponding check node degree dc. Therefore, it is important to find the optimal (Nc, dc)-CZZ code for a given code rate. In this paper, the density evolution analysis of CZZ codes considering two-way scheduling is performed and compared with that for flooding scheduling. Based on density evolution analysis, we confirm that the performance does not depend on the scheduling. The threshold values are computed for various CZZ codes and the optimal CZZ codes for various code rates are obtained.

Ⅱ. Analysis of CZZ Codes Using Density Evolution with a Gaussian Approximation

Richardson and Urbanke demonstrated that the

※ 본 연구는 삼성전자 SKYPASS4G 과제 지원 및 관리로 수행되었습니다.
* 삼성전자 통신연구소 (sn7955.hong@samsung.com)
** 한양대학교 전자통신컴퓨터 공학부 (djshin@hanyang.ac.kr), (° : 교신저자)
논문번호 : KICS2006-01-013, 접수일자 : 2006년 1월 6일

average asymptotic behavior of LDPC decoder using sum-product algorithm could be numerically estimated by using an algorithm called density evolution[9]. Chung, Richardson, and Urbanke have shown that the messages passed in the sum-product decoding on the AWGN channel can be well approximated by Gaussian random variables[10], that allows us to easily understand the behavior of the decoder. In this section, we apply density evolution with a Gaussian approximation to see the average asymptotic behavior of (Nc, dc)-CZZ decoder using sum-product algorithm on two-way scheduling. In this case, the messages sent on edges represent a posteriori densities on the bit associated with the incident variable node and it can be compactly represented by log-likelihood ratio (LLR). These LLR messages are updated at each iteration in the sum-product decoding[9]. For the simplicity of analysis, it is assumed that all-0 codeword is transmitted[9].

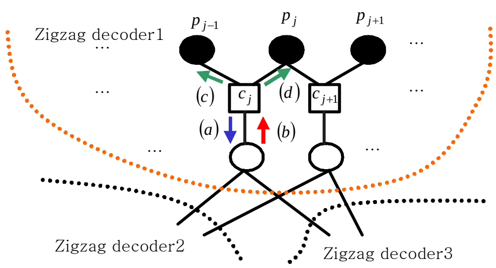


Fig.1. Message flows in CZZ decoder using two-way scheduling when Nc=3.

From now on, we will derive the recursion equations for various (Nc, dc)-CZZ codes with two-way scheduling. Let m_{u_0} be the mean of message from the channel at variable node and Nch is the number of check nodes in each zigzag code. Then, $m_{u_0} = 2/\sigma^2$ since all-0 codeword is transmitted. If two information nodes are connected to the same check node, the means of messages from information nodes to the check node are the same. Therefore, we do not distinguish the information nodes connected to the same check node and only consider Nch information nodes. We assume that the zigzag interleaver randomizes the information bits perfectly and the number of information bits is sufficiently large. The density evolution analysis for two-way scheduling is done by two levels considering messages passed between zigzag decoders and messages passed in each zigzag decoder.

First, we consider the messages passed between zigzag decoders. $m_{I^{(l,k)}(j)}$ and $m_{v^{(l,k)}(j)}$ denote the means of messages from the jth check node of the kth zigzag decoder to information node at the lth iteration and vice versa. $m_{I^{(l,k)}(j)}$ and $m_{v^{(l,k)}(j)}$ are corresponding to (a) and (b) in Fig. 1, respectively. Let $I^{(l,k)}$ be the message from randomly selected check node to information node. Then, the probability of $I^{(l,k)} = I^{(l,k)}(j)$ is equal to $1/N_{ch}$ for all j. Therefore, $I^{(l,k)}$ has the following Gaussian mixture probability density $f_{I^{(l,k)}(j)}(\bullet)$.

$$f_{I^{(l,k)}}(x) = \frac{1}{N_{ch}} \sum_{j=1}^{N_{ch}} f_{I^{(l,k)}(j)}(x), k=1,2,\cdots,N_c$$

where $f_{I^{(l,k)}(j)}(\bullet)$ means the Gaussian probability density function with mean $m_{I^{(l,k)}(j)}$ and variance $2m_{I^{(l,k)}(j)}$. The message $m_{v^{(l+1,k)}}$ by

$$v^{(l+1,k)} = \sum_{n=1, n\neq k}^{N_c} I^{(l,n)}, k=1,2,\cdots,N_c$$

$v^{(l+1,k)}$ is the sum of $I^{(l,n)}$ which can be assumed Gaussian since $I^{(l,n)}(j)$ has the almost same mean value for all j. Therefore, $v^{(l+1,k)}$ is well approximated by Gaussian random variable and only the mean and variance are needed, which can be calculated as follows.

$$\begin{aligned} E[v^{(l+1,k)}] &= \sum_{n=1, n\neq k}^{N_c} E[I^{(l,n)}] \\ &= (N_c - 1) E[I^{(l,1)}] \\ &= (N_c - 1) \frac{1}{N_{ch}} \sum_{j=1}^{N_{ch}} m_{I^{(l,1)}}(j), \\ k &= 1,2,\cdots,N_c \end{aligned}$$

where n is fixed by 1 since $E[I^{(l,n)}]$ does not depend on n.

$$VAR[v^{(l+1,k)}] = (N_c - 1)\, VAR[I^{(l,1)}] = (N_c - 1)\frac{1}{N_{ch}}\sum_{j=1}^{N_{ch}} 2m_{I^{(l,1)}}(j), \quad k = 1, 2, \cdots, N_c.$$

Since $v^{(l+1,k)}$ also satisfies the symmetry condition[3], we only need to track the mean.

Second, we consider the messages passed in each zigzag decoder. We denote the means of messages from the jth check node (cj in Fig. 1) to the left-parity node (pj-1 in Fig. 1) and the right-parity node (pj in Fig. 1) at the lth iteration by $m_{L^{(l)}(j)}$ and $m_{R^{(l)}(j)}$, respectively. These are corresponding to (c) and (d) in Fig. 1, which are only used in each zigzag decoder and not passed to any other zigzag t decoder. Therefore, the mark to denote the zigzag decoder can be omitted.

Two-way scheduling used in each zigzag decoder is divided into three levels[7]. Three recursion equations obtained by density evolution are derived by considering three levels separately as follows.

$$m_{L^{(l)}(j)} = \phi^{-1}(1 - [1 - \phi(m_{u_0} + (N_c - 1) \cdot \frac{1}{N_{ch}}\sum_{j=1}^{N_{ch}} m_{I^{(l-1)}(j)})]^{d_c - 2}[1 - \phi(m_{u_0} + m_{L^{(l)}(j+1)})]) \quad j = 1, 2, \cdots, N_{ch}, \tag{1}$$

$$m_{R^{(l)}(j)} = \phi^{-1}(1 - [1 - \phi(m_{u_0} + (N_c - 1) \cdot \frac{1}{N_{ch}}\sum_{j=1}^{N_{ch}} m_{I^{(l-1)}(j)})]^{d_c - 2}[1 - \phi(m_{u_0} + m_{R^{(l)}(j-1)})]) \quad j = N_{ch}, N_{ch} - 1, \cdots, 1, \tag{2}$$

$$m_{I^{(l)}(j)} = \phi^{-1}(1 - [1 - \phi(m_{u_0} + (N_c - 1) \cdot \frac{1}{N_{ch}}\sum_{j=1}^{N_{ch}} m_{I^{(l-1)}(j)})]^{d_c - 3}[1 - \phi(m_{u_0} + m_{R^{(l)}(j-1)})] \cdot [1 - \phi(m_{u_0} + m_{L^{(l)}(j)})]), \quad j = 1, 2, \cdots, N_{ch}, \tag{3}$$

where $\phi(\cdot)$ is defined in[10].

In (3), the superscript k of $I^{(l,k)}(j)$ is omitted since it does not depend on k. The mean of message at information node can be tracked by recursively computing the mean value by using (1), (2) and (3). The number of iterations is sufficiently large enough to determine whether the error can be corrected for the given channel noise level and the threshold σ^* is obtained. The values of σ^* for various CZZ codes are calculated in the next section.

Ⅲ. Optimal Concatenated Zigzag Codes

Since CZZ code can have various structures for a fixed code rate, it is important to find the optimal (Nc, dc)-CZZ code. In most of the mobile communication systems, the code rate of mother error-correcting code is usually 1/n and by modifying this code (e.g. using puncturing), higher rates can be achieved. Therefore, we only consider the (Nc, dc)-CZZ codes of rate 1/n. Using (1), (2) and (3), we can easily compute the threshold values for various (Nc, dc)-CZZ codes using two-way scheduling. The threshold values of various CZZ codes are given in Table I. The overall code rate R of CZZ code is $R = 1/(N_c r + 1)$ where r is the code rate of zigzag code and $r = 1/(d_c - 2)$. Therefore, we get $N_c = (1/R - 1)(d_c - 2)$ and for the code rate 1/n, $N_c = (n-1)(d_c - 2)$ should be satisfied.

From Table I, we can see that CZZ codes having four zigzag codes (Nc=4) show the best performance for code rates 1/2, 1/3 and 1/5. For the rate 1/4, the optimal number of component codes is 6 and note that Nc=4 is impossible for this rate. The CZZ codes having the minimum number of zigzag codes are optimal for the rates less than 1/5. If the number of zigzag codes increases, weaker zigzag codes have to be used and at some point concatenating more zigzag codes with weaker error correcting capability gives worse overall performance. To verify the analytical results in Table I, they are compared using the simulation results. For the simulations, 1000-bit information frames, BPSK modulation, Sum-product algorithm, random interleavers and AWGN channel are assumed. 20 and 60 iterations are used for two-way and flooding schedulings since the performance gain is negligible as the iteration numbers become larger these numbers. However, if the number of information bits increases, the iteration number can be increased to get better performance.

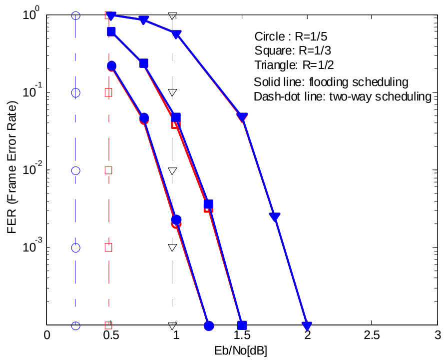


Fig.2. Comparison of CZZ codes when different schedulings are used

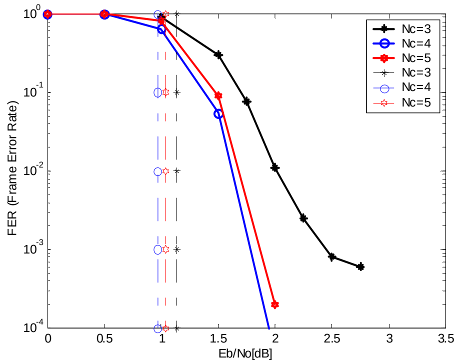
Fig.2 compares the simulation results of the optimal CZZ codes with rates 1/2, 1/3 and 1/5 when using two-way and flooding schedulings, respectively. This simulation results are well matched to the analytical results in Table I. Although the simulation results are about 1dB far from the analytical bound at FER=10-4because of the short codeword length, the differences among simulation curves are well matched to those for the analytical results. Therefore, we can conclude that the performance of CZZ code does not depend on which scheduling is used. When many interleavers are used to construct a CZZ code of low code rate, it may be efficient to use flooding scheduling since it has smaller decoding delay compared with the two-way scheduling. However, the penalty is the big memory requirement for the sparse parity-check matrix.

The simulation results in Fig. 3 also confirm the analytical results in Table I. As the number of zigzag codes increases, the simulation results become farther from those analytically predicted by density evolution. The reason is that it is difficult to construct many interleavers which are perfectly random to each other and, as a result,

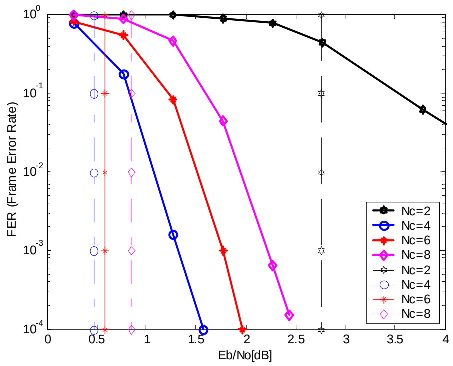
TABLE I. THRESHOLD VALUES FOR VARIOUS (Nc, dc)-CZZ CODES

Rate	σ^*		Nc	dc
	Flooding scheduling	Two-way scheduling		
1/2	0.877	0.877	3	5
	0.894	0.895	4	6
	0.887	0.888	5	7
	0.856	0.857	6	8
1/3	0.891	0.891	2	3
	1.159	1.160	4	4
	1.145	1.146	6	5
	1.110	1.111	8	6
1/4	1.322	1.322	3	3
	1.357	1.358	6	4
	1.305	1.306	9	5
	1.249	1.251	12	6
1/5	1.541	1.542	4	3
	1.502	1.503	8	4
	1.431	1.433	12	5
1/6	1.702	1.703	5	3
	1.623	1.626	10	4
	1.541	1.544	15	5
1/7	1.835	1.837	6	3
	1.729	1.733	12	4
	1.641	1.645	18	5
1/8	1.950	1.952	7	3
	1.826	1.830	14	4
	1.735	1.840	21	5

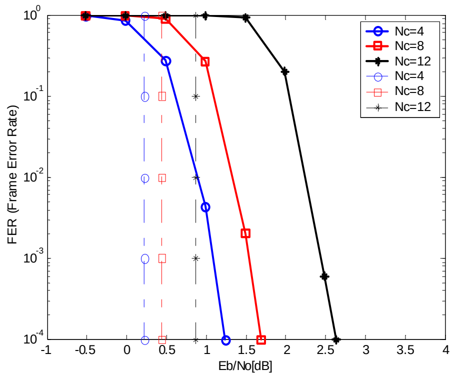
short cycles appear in Tanner graph. However, the performance trend for various Nc's is well explaining the analytical result. For regular LDPC codes, the number of edges connected to the variable nodes should be larger than two to achieve good performance and avoid error floor [1, 9]. From Fig.3 (a) and (b),we can see that the number of edges connected to the information nodes (i.e. the number of zigzag codes) should be larger than three to avoid error floor and it is confirmed that Nc = 4 is optimal for rates 1/2, 1/3 and 1/5.



(a) Code rate= 1/2



(b) Code rate= 1/3



(c) Code rate= 1/5

Fig.3. Comparison of various CZZ codes for a fixed rate (Dash-dot line: Threshold values, Solid line: Simulation result).

Ⅳ. Conclusions

In this paper, we applied density evolution with a Gaussian approximation to CZZ codes when two-way scheduling is used and derive the new recursion equations. By using these equations, we obtained the threshold values for various CZZ codes and based on those results, the threshold values of various CZZ codes are calculated. Since the thresholds of flooding and two-way schedulings are the same, it is confirmed that the asymptotic performance of CZZ codes does not depend on the scheduling scheme. Also, the optimal structures of CZZ codes are derived for various code rates. As the future work, it may be an interesting problem to design the zigzag interleavers such that the short cycles can be avoided.

References

〔1〕 R. G. Gallager, “Low-density parity-check codes,” IRE Trans. Inform. Theory, pp. 21-28, Jan. 1962.

〔2〕 D. J. C. MacKay and R. M. Neal, “Near Shannon limit performance of low desnsity parity check codes,” IEE Electon. Lett., pp. 1645-1646, Aug. 1996.

〔3〕 T. J. Richardson, A. Shokrollahi, and R. L. Urbanke, “Design of capacity-approaching irregular low-density parity-check codes,” IEEE Trans. Inform. Theory, pp. 619-637, Feb. 2001.

〔4〕 T. J. Richardson and R. L. Urbanke, “Efficient encoding of low-density parity-check codes,” IEEE Trans. Inform. Theory, pp. 638 -656, Feb. 2001.

〔5〕 H. Jin, A. Khandekar, and R. McEliece, “Irregular repeat-accumulate codes,” pp. 1-8 in Proc. 2nd International Symposium on Turbo Codes and Related Topics, Brest, France, Sept. 2000.

〔6〕 L. Ping and K. Y. Wu, “Concatenated tree codes: a low-complexity, high-performance approach,” IEEE Trans. Inform. Theory, pp. 791-799, Feb. 2001.

〔7〕 L. Ping, X. L. Huang, and N. Phamdo, “Zigzag codes and concatenated zigzag codes,” IEEE Trans. Inform. Theory, pp. 800-807, Feb. 2001.

〔8〕 F. R. Kschischang and B. J. Frey, “Iterative decoding of compound codes by probability propagation in graphical models,” IEEE J. Select.

Areas Commun., pp. 219-230, Feb. 1998.
〔9〕 T. J. Richardson and R. L. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," IEEE Trans. Inform. Theory, pp. 599-618, Feb. 2001.
〔10〕 S. Y. Chung, T. J. Richardson, and R. L. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," IEEE Trans. Inform. Theory, pp. 657-670, Feb. 2001.

홍 송 남 (Song-Nam Hong) 준회원

2003년 2월 : 한양대학교 전자전기컴퓨터공학부 학사
2005년 2월 : 한양대학교 전자통신전파공학과 석사
2005년 3월~현재 : 삼성전자 통신연구소 연구원
<주관심분야> 디지털통신, 오류정정부호

신 동 준 (Dong-Joon Shin) 종신회원

1990년 2월 : 서울대학교 전자공학 공학사
1991년 12월 : Northwestern University, 전기공학 석사
1998년 12월 : University of Southern California, 전기공학과 공학박사
1999년 1월~1999년 4월 : Research Associate(USC)
1999년 4월~2000년 8월 : Hughes Network Systems, MTS
2000년 9월~현재 : 한양대학교 전자통신컴퓨터공학부 부교수
<주관심분야> 디지털통신, 오류정정부호, 이산수학, 시퀀스