New Message-Passing Decoding Algorithm of LDPC Codes by Partitioning Check Nodes[1]

Sunghwan Kim* O, Min-Ho Jang*, Jong-Seon No*, Song-Nam Hong†, and Dong-Joon Shin†
*School of Electrical Engineering and Computer Science, Seoul National University
†Division of Electrical and Computer Engineering Hanyang University
{nodoubt, mhjang}@ccl.snu.ac.kr, jsno@snu.ac.kr, sunny795@ccrl.hanyang.ac.kr, djshin@hanyang.ac.kr

Abstract

In this paper, we propose a new sequential message-passing decoding algorithm of low-density parity-check (LDPC) codes by partitioning check nodes. This new decoding algorithm shows better bit error rate (BER) performance than that of the conventional message-passing decoding algorithm [3],[7], especially for small number of iterations. Analytical results tell us that as the number of partitioned subsets of check nodes increases, the BER performance becomes better. We also derive the recursive equations for mean values of messages at variable nodes by using density evolution with Gaussian approximation. From these equations, the mean values at each iteration of the new decoding algorithm can be obtained. Simulation results also confirm the analytical results.

1. Introduction

In 1996, low-density parity-check (LDPC) codes, originally invented by Gallager [1], were rediscovered by MacKay and Neal [6]. Since then, LDPC codes have been the main research topic in error-control coding area because these codes show the capacity-approaching performance with practical complexity. Compared with turbo codes, they have lower decoding complexity due to the iterative decoding algorithm (or message-passing algorithm) based on the sum-product algorithm, but slower decoding convergence speed.

Recently, there has been a great deal of efforts on implementing LDPC decoder. In general, hardware implementation of LDPC decoder uses parallel processing. However, if the decoder cannot be implemented in the fully parallel processing mode, sequential decoding approach has to be used. An efficient sequential decoding algorithm and the realizable hardware implementation of LDPC decoders are introduced in [8], where messages between each variable node and its neighbors are sequentially updated without partitioning check nodes. In the conventional fully parallel message-passing decoding algorithm, many iterations (50 or more) are required to achieve the desired performance, which results in high decoding complexity. For reducing the number of iterations to achieve the desired performance, we investigate an efficient sequential message-passing decoding algorithm of LDPC codes. For the fixed bit error rate (BER), the number of iterations in the new decoding algorithm is substantially reduced in comparison with fully parallel decoding algorithm. Therefore, it can be especially useful for wireless communication systems which require low complexity decoding algorithm.

The new sequential message-passing decoding algorithm can be briefly explained as follows. First, partition the check nodes of an LDPC code into several subsets appropriately. Then, this LDPC code can be described by the interconnected several subgraphs, each of which consists of a subset of check nodes and the connected variable nodes. The decoding can be performed by applying the message-passing decoding algorithm to each subgraph in the sequential order. This algorithm can be efficiently adopted in the LDPC decoder when the fully parallel processing mode cannot be implemented. In one iteration, the complexity of the new decoding algorithm is the same as that of the conventional fully parallel decoding algorithm, but the convergence speed of the new decoding algorithm is faster.

2. A New Sequential Message-Passing Decoding Algorithm

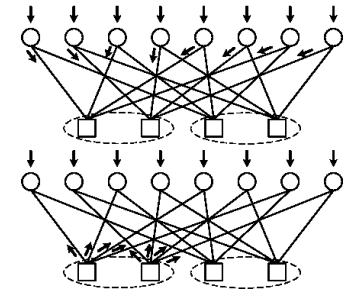
First, conventional message-passing decoding algorithm of (d_v, d_c) regular LDPC codes will be reviewed as follows [3]. A check node (or a variable node) receives messages from its d_c neighbors (or its d_v neighbors and its corresponding channel output), processes the messages, and passes the updated messages back to its neighbors. Each output message of a variable or a check node is a function of all incoming messages to the node except for the incoming message on the edge

[1]This work was supported in part by BK21 and ITRC program of the Korean Ministry of Information and Communications.

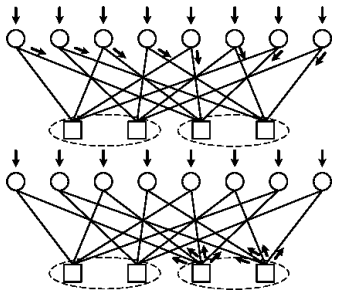
where the output message will be sent.

Now, we introduce the new sequential message-passing decoding algorithm. Each iteration in the conventional iterative decoding algorithm consists of two steps. The first step contains calculating messages at all variable nodes and sending them to all check nodes and the second step contains calculating messages at all check nodes and sending them to all variable nodes. These operations are performed simultaneously.

In a new sequential message-passing decoding algorithm, we assume that the check nodes are partitioned into p subsets. The messages from variable nodes to the check nodes in the first subset are updated and then the messages from the check nodes in the first subset to their neighboring variable nodes are updated, which corresponds to one iteration for the first subset of check nodes. This decoding procedure is sequentially applied to the remaining $p-1$ subsets of check nodes. One iteration in the new decoding algorithm means the above sequential message updating and passing for all variable nodes and all subsets of the check nodes. Thus, it is clear that the amount of computation for one iteration in the new decoding algorithm is the same as that of the conventional decoding algorithm.



(a)



(b)

Fig. 1: One iteration in the new decoding algorithm of a $(2, 4)$ regular LDPC code with length 8 and $p = 2$, (a) Message passing for the first check node subset, (b) Message passing for the second check node subset.

Fig. 1 shows the new decoding procedures for a $(2, 4)$ regular LDPC code of length 8 when $p = 2$. Here, circles and squares stand for variable nodes and check nodes, respectively. The messages received from the channel are represented by the arrows at the top of the circles.

In Fig. 1 (b), the messages from variable nodes to the check nodes in the second subset are updated by using the messages to the variable nodes, which were already updated from the first check node subset as in Fig. 1 (a). In the conventional message-passing decoding algorithm, messages of all variable and check nodes at the l-th iteration are updated by using the messages updated at the $(l-1)$-th updated messages and passed to their neighbors. Let S_i, $1 \le i \le p$, be the i-th subset of check nodes. During the l-th iteration in the new decoding algorithm, the l-th updated messages from $S_1, S_2, \cdots, S_{i-1}$ and the $(l-1)$-th updated messages from the remaining subsets are used for the l-th message updating between the variable nodes and the check nodes in S_i. This is why the new decoding algorithm can give the faster convergence speed.

3. Analysis by Density Evolution with Gaussian Approximation

Density evolution with Gaussian approximation is based on approximating the probability densities of messages as Gaussians or Guassian mixtures [3]. Since this is easier to analyze and computationally faster than the conventional density evolution, it is a useful method for investigating the behavior of the message-passing decoding algorithm. We will only consider (d_v, d_c) regular LDPC codes. For irregular LDPC codes, the similar analysis can be used. Density evolution with Gaussian approximation in conventional decoding algorithm is shown in [3]. Let m_u and m_v be the means of u and v, respectively. And let $\phi(x)$ be a function defined as

$$\phi(x) = \begin{cases} 1 - \frac{1}{\sqrt{4\pi m_u}} \int_R \tanh \frac{u}{2} e^{-\frac{(u-m_u)^2}{4m_u}} du, & \text{if } x > 0 \\ 1, & \text{if } x = 0. \end{cases} \quad (1)$$

A. Density Evolution with Gaussian Approximation for Random Partitioning

We assume that the edges between the variable nodes and p partitioned subsets of the check nodes are randomly connected. We call it *random partitioning*. Let S_i, $1 \le i \le p$, be the i-th subset of check nodes and u_{S_i} be the message from a check node in S_i to a variable node. The number of variable nodes for each type of connections from a variable node to p subsets may vary according to the code structure, but we will assume that these numbers are constant since the partitioning is considered as random.

Let $(a_1, a_2, \cdots, a_p)$ be the distribution of edges from a variable node to p subsets, where a_i denotes the number of edges connected from the variable node to S_i. Then, for all $(a_1, a_2, \cdots, a_p)$, the means of messages from the variable node to S_i in the conventional message-passing decoding algorithm can be expressed

as

$$m_{u_0} + (a_i - 1)m_{u_{S_i}}^{(l-1)} + \sum_{\substack{j=1 \\ j\neq i}}^{p} a_j m_{u_{S_j}}^{(l-1)}.$$

By considering the sequential decoding from S_1 to S_p in the new sequential message-passing decoding algorithm, the above equation should be modified as

$$m_{u_0}+(a_i-1)m_{u_{S_i}}^{(l-1)}+\sum_{j=1}^{i-1} a_j m_{u_{S_j}}^{(l)} + \sum_{j=i+1}^{p} a_j m_{u_{S_j}}^{(l-1)}. \quad (2)$$

The probability that the messages with mean value in (2) is passed to S_i can be derived as

$$\frac{a_i}{\sum_{j=1}^{d_v} j \cdot_{p-1} \mathrm{H}_{d_v-j}}. \quad (3)$$

Then recursive equation for the means of messages of check node in S_i can be expressed as

$$\begin{aligned} m_{u_{S_i}}^{(l)} = \phi^{-1}(1 - [1 - \sum_{(a_1,\cdots,a_p)} & \frac{a_i}{\sum_{j=1}^{d_v} j \cdot_{p-1} \mathrm{H}_{d_v-j}} \\ & \times\phi(m_{u_0} + (a_i - 1)m_{u_{S_i}}^{(l-1)} \\ & + \sum_{j=1}^{i-1} a_j m_{u_{S_j}}^{(l)} + \sum_{j=i+1}^{p} a_j m_{u_{S_j}}^{(l-1)})]^{d_c-1}). \quad (4) \end{aligned}$$

B. Density Evolution with Gaussian Approximation for Uniform Partitioning

Next, we consider the case that the edges from a variable node are distributed among subsets $S_1, S_2, \cdots, S_p$ as uniformly as possible. We call it *uniform partitioning*.

1) Case of $p < d_v$

Let d_v be $b \cdot p + r$, where b is an positive integer and r, $0 \le r \le p-1$ is a nonnegative integer. Then, for all $(a_1, a_2, \cdots, a_p)$, where a_i is b or $b+1$, the means of messages from the variable node to S_i in the new message-passing decoding algorithm can be expressed as

$$m_{u_0}+(a_i-1)m_{u_{S_i}}^{(l-1)}+\sum_{j=1}^{i-1} a_j m_{u_{S_j}}^{(l)} + \sum_{j=i+1}^{p} a_j m_{u_{S_j}}^{(l-1)}. \quad (5)$$

The probability that the messages with the mean value in (5) is passed to S_i can be derived as

$$\frac{a_i}{b \times_{p-1} \mathrm{C}_r + (b+1) \times_{p-1} \mathrm{C}_{r-1}}. \quad (6)$$

The recursive equation for the means of messages of check node in S_i can be expressed as

$$\begin{aligned} m_{u_{S_i}}^{(l)} = \phi^{-1}(1 - [1 - \sum_{(a_1,\cdots,a_p)} & \frac{a_i}{b \times_{p-1} \mathrm{C}_r + (b+1) \times_{p-1} \mathrm{C}_{r-1}} \\ & \times\phi(m_{u_0} + \sum_{j=1}^{i-1} a_j m_{u_{S_j}}^{(l)} + \sum_{j=i+1}^{p} a_j m_{u_{S_j}}^{(l-1)}]^{d_c-1}). \end{aligned}$$

2) Case of $p \ge d_v$

In this case, the number of different types of connections from a variable node to p subsets becomes ${}_p\mathrm{C}_{d_v}$. Assume that at least one edge from a variable node is connected to S_i. Then, for all $(a_1, a_2, \cdots, a_p)$, where a_i is zero or one, the means of messages from the variable node to S_i in the new message-passing decoding algorithm can be expressed as

$$m_{u_0} + \sum_{j=1}^{i-1} a_j m_{u_{S_j}}^{(l)} + \sum_{j=i+1}^{p} a_j m_{u_{S_j}}^{(l-1)}. \quad (7)$$

The probability that the messages with mean value in (7) is passed to S_i can be derived as $\frac{1}{{}_{p-1}\mathrm{C}_{d_v-1}}$.

The recursive equation for the means of messages in the check nodes of S_i can be expressed as

$$\begin{aligned} m_{u_{S_i}}^{(l)} = \phi^{-1}(1 - [1 - \sum_{(a_1,\cdots,a_p)} & \frac{1}{{}_{p-1}\mathrm{C}_{d_v-1}} \\ & \times\phi(m_{u_0} + \sum_{j=1}^{i-1} a_j m_{u_{S_j}}^{(l)} + \sum_{j=i+1}^{p} a_j m_{u_{S_j}}^{(l-1)}]^{d_c-1}). \end{aligned}$$

C. Density Evolution When $p = 2$

As an example, we will partition the check nodes of a $(3, 6)$ regular LDPC code into S_1 and S_2 and derive the recursive equations for means of messages as in (4). Also a good partition method of check nodes is explained.

First, we assume that the edges between the variable nodes and the check nodes in S_1 and S_2 are randomly connected. In this case, there are ${}_2\mathrm{H}_3 = 4$ different types of connections between the variable nodes and two subsets of check nodes. The means of messages from the variable node to the check nodes in S_1 can be given as one of $m_{u_0} + m_{u_{S_1}} + m_{u_{S_2}}$, $m_{u_0} + 2m_{u_{S_1}}$, and $m_{u_0} + 2m_{u_{S_2}}$ with probabilities $2/6$, $3/6$, and $1/6$, respectively. Similarly, the means of messages from the variable node to the check nodes in S_2 are also given as one of $m_{u_0} + m_{u_{S_1}} + m_{u_{S_2}}$, $m_{u_0} + 2m_{u_{S_2}}$, and $m_{u_0} + 2m_{u_{S_1}}$ and with probabilities $2/6$, $3/6$, and $1/6$, respectively. Then the recursive equations for the means of messages from the check nodes in S_1 and S_2 to the variable nodes are given as follows;

$$\begin{aligned} m_{u_{S_1}}^{(l)} = \phi^{-1}(1 - [1 - & \frac{3}{6}\phi(m_{u_0} \\ & + 2m_{u_{S_1}}^{(l-1)}) - \frac{1}{6}\phi(m_{u_0} + 2m_{u_{S_2}}^{(l-1)}) \\ & - \frac{2}{6}\phi(m_{u_0} + m_{u_{S_1}}^{(l-1)} + m_{u_{S_2}}^{(l-1)})]^5) \\ m_{u_{S_2}}^{(l)} = \phi^{-1}(1 - [1 - & \frac{3}{6}\phi(m_{u_0} \\ & + 2m_{u_{S_2}}^{(l-1)}) - \frac{1}{6}\phi(m_{u_0} + 2m_{u_{S_1}}^{(l-1)}) \\ & - \frac{2}{6}\phi(m_{u_0} + m_{u_{S_1}}^{(l)} + m_{u_{S_2}}^{(l-1)})]^5). \end{aligned}$$

Next, we consider the case that the edges from a variable node are distributed among subsets

$S_1, S_2, \cdots, S_p$ as uniformly as possible. For the above example, it means that among three edges from the variable node, two edges are connected to S_1 and one edge is connected to S_2, and vice versa.

The means of messages from the variable nodes to the check nodes in S_1 can be $m_{u_0} + m_{u_{S_1}} + m_{u_{S_2}}$ and $m_{u_0} + 2m_{u_{S_2}}$ with probabilities 2/3 and 1/3, respectively. The means of messages from the variable nodes to the check nodes in S_2 can also be $m_{u_0} + m_{u_{S_1}} + m_{u_{S_2}}$ and $m_{u_0} + 2m_{u_{S_1}}$ with probabilities 2/3 and 1/3, respectively.

Then the recursive equations considering subsets S_1 and S_2 are given as follows;

$$\begin{aligned} m_{u_{S_1}}^{(l)} &= \phi^{-1}(1-[1-\frac{1}{3}\phi(m_{u_0}+2m_{u_{S_2}}^{(l-1)}) \\ &\quad -\frac{2}{3}\phi(m_{u_0}+m_{u_{S_1}}^{(l-1)}+m_{u_{S_2}}^{(l-1)})]^5) \\ m_{u_{S_2}}^{(l)} &= \phi^{-1}(1-[1-\frac{1}{3}\phi(m_{u_0}+2m_{u_{S_1}}^{(l)}) \\ &\quad -\frac{2}{3}\phi(m_{u_0}+m_{u_{S_1}}^{(l)}+m_{u_{S_2}}^{(l-1)})]^5). \end{aligned}$$

Equation (1) can be simplified as $\phi(x) \approx e^{-0.4527x^{0.86}+0.0218}$ in [3]. Then the threshold value for the message-passing decoding algorithm of the (3,6) regular LDPC codes is given as 0.8747 which is also the threshold value for new decoding algorithm. Under the AWGN (additive white Gaussian noise) channel with the standard deviation values 0.83 or 0.87 which are less than the above threshold value, the mean values at each iteration for (3,6) regular LDPC code are obtained as in Fig. 2. The number of iterations is denoted by I.

In Fig. 2, R and U stand for random partitioning and uniform partitioning, respectively. It is shown in Fig. 2 that partitioning the check nodes increases the convergence speed of the message-passing decoding algorithm of LDPC codes. It is also shown that the uniform partitioning is better than the random partitioning in terms of convergence speed of LDPC decoding. As the channel is becoming worse, the convergence speed gain of the new decoding algorithm is getting larger.

4. Simulation Results

Simulation for the proposed decoding algorithm of LDPC codes with $p = 1, 3$, and 4 is done in the AWGN channel. Fig. 3 shows the BER performance of irregular LDPC code with length 1000 and rate 1/2, which was constructed by optimizing the degree distribution [3] with restricting the maximum degree of variable nodes to 8. Fig. 4 shows the BER performance of a (3, 6) regular LDPC code with length 1000 and rate 1/2, which is randomly constructed. In Fig. 3 and 4, the new decoding algorithm with random partitioning is used.

From the simulation results in Fig. 3 and Fig. 4, the BER performance of our proposed decoding algorithm is better than that of the conventional decoding

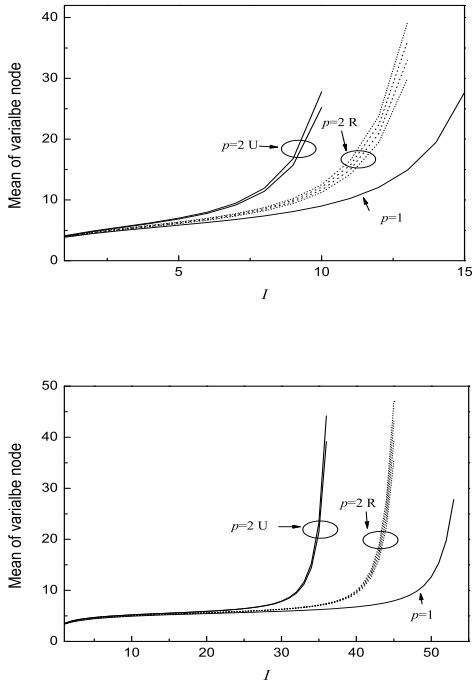


Fig. 2: Density evolutions of (3, 6) regular LDPC codes for $\sigma = 0.83$ (above) and 0.87 (below).

algorithm, especially for $I = 5$ and 10. However, for $I = 50$, the BER performance improvement decreases since both decoding methods achieve enough iteration gain. Note that both decoding algorithm have the same threshold values.

Fig. 5 and 6 show the performance improvement of our proposed decoding algorithm with uniform partitioning for a (3, 6) regular QC LDPC code with length 4092 and rate 1/2, which is constructed by computer search.

5. Conclusions

The proposed sequential message-passing decoding

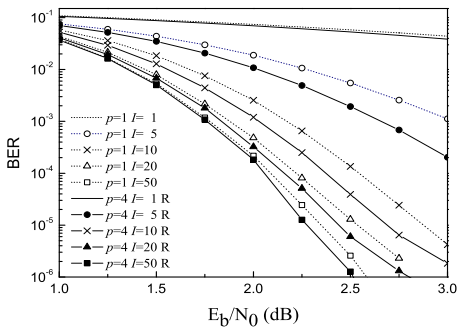


Fig. 3: BER performance with random partitioning when an irregular LDPC code with length 1000 and rate 1/2 is used.

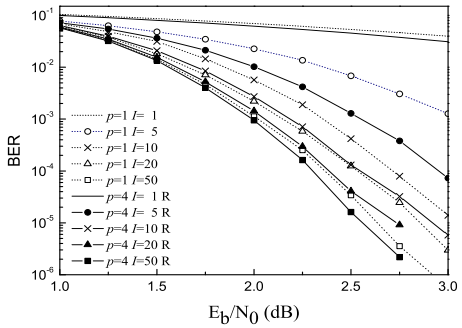


Fig. 4: BER performance with random partitioning when a (3, 6) regular LDPC code with length 1000 and rate 1/2 is used.

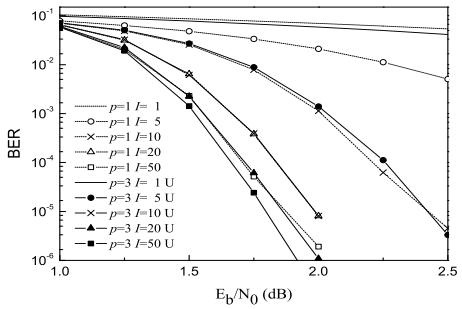


Fig. 5: BER performance with partitioning and no partitioning of a (3, 6) regular QC LDPC code with length 4092 and rate 1/2.

algorithm outperforms the conventional decoding algorithm for the small number of iterations. This means that the proposed algorithm improves the convergence speed without an increase of the decoding complexity. We investigated the reason of fast convergence of our proposed decoding algorithm from analysis of density evolution with Gaussian approximation. Moreover, the proposed algorithm can be applied to any code represented by Tanner graph. Therefore the new decoding algorithm of LDPC codes can be used to implement the practical decoder.

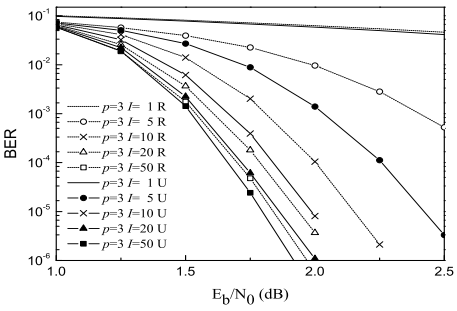


Fig. 6: BER performance with uniform and random partitioning of a (3, 6) regular QC LDPC code with length 4092 and rate 1/2.

References

[1] R. G. Gallager, *Low-density parity-check codes.* Cambridge, MA: MIT Press 1963.

[2] R. Tanner, "A recursive approach to low complexity code," *IEEE Trans. Inform. Theory*, vol. IT-27, pp. 533-547, Sept. 1981.

[3] S.-Y. Chung, *On the construction of some capacity-approaching coding schemes.* PhD thesis, MIT, Sept. 2000.

[4] M. Fossorier, "Quasi-cyclic low-density parity-check codes from circulant permutation matrices," *IEEE Trans. Inform. Theory*, vol. 50, no. 8, pp. 1788-1793, Aug. 2004.

[5] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. IT-47, no.2, pp. 533-547, Feb. 2001.

[6] D. J. C. Mackay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *IEEE Electron. Lett.*, vol. 32, no. 18, pp. 1645-1646, Aug. 1996.

[7] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 599-618, Feb. 2001.

[8] M. Cocco, J. Dielissen, M. Heijligers, A. Hekstra, and J. Huisken, "A scalable architecture for LDPC decoding," in *Proc. DATE'04*, pp. 88-93, Feb. 2004.

[9] T. Richardson, M. Shokrollahi, and R. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 619-637, Feb. 2001.