A Low-Complexity Adaptive Extended Min-Sum Algorithm for Non-Binary LDPC Codes

Min-Ho Kim[1], Kyeong-Bin Park[2], and Ki-Seok Chung[*]

Department of Electronics and Computer Engineering

Hanyang University

Seoul, South Korea

e-mail: [1]minhokim1000@hanyang.ac.kr, [2]lay1523@hanyang.ac.kr, and [*]kchung@hanyang.ac.kr

***Abstract*— The extended min-sum algorithm (EMS) for decoding non-binary low density parity check (NB-LDPC) codes reduces the decoding complexity by truncating the message vector by retaining only the most reliable symbols. However, the EMS algorithm does not consider that the noise of the received codeword is gradually reduced as the iteration count goes up. In this paper, we propose a low-complexity adaptive EMS algorithm, called threshold-based EMS (TB-EMS). The TB-EMS algorithm has a simple adaptive rule to calculate the new message vector length compared to the A-EMS. The proposed algorithm selects one of two message vector lengths. Experimental results show that the proposed algorithm reduces the decoding complexity with minimal performance degradation compared with the EMS algorithm. Further, the decoding performance of the TB-EMS is better than A-EMS.**

Keywords-Non-Binary Low Density Parity Check codes; Extended Min-Sum algorithm; Low-complexity decoding; Message truncation rule

I. INTRODUCTION

Non-binary low density parity check (NB-LDPC) codes over Galois field GF(q) (q>2) can provide a better error correction performance than binary LDPC codes, but the decoding complexity is significantly increased [1], [2]. For example, in the case of q-ary sum product algorithm (QSPA), which is one of the decoding algorithms, the computational complexity of a check node update operation is $O(q^2)$ [3]. Various types for low complexity decoding algorithms have been proposed [4], [5], [6], [7]. Among them, to reduce the complexity, the extended min-sum algorithm (EMS) retains only the most reliable n_m (<q) symbols among q symbols of the message vector [7]. As a result, the computational complexity of the check node update process is greatly reduced to $O(n_m^2)$.

During the decoding process, as the number of iterations increases, the noise level of the received codeword is gradually reduced, and the message vector length can be decreasing accordingly. To take advantage of this characteristics, adaptive EMS algorithms that vary the number of retained messages are proposed [8], [9], [10].

In [8], an adaptive EMS algorithm called A-EMS on a codeword basis was proposed. To reduce the decoding complexity, the effective message vector length is calculated by check node error rate ($CNER$), which represents the reliability of check nodes. The A-EMS algorithm calculates the truncation size as follows.

$$n_a^k = A + C \times (\Delta CNER_k)^2 \tag{1}$$

n_a^k represents the adaptive message vector length to be used in the $(k+1)^{th}$ iteration and A is the least message vector length. C and $\Delta CNER_k$ are predetermined parameters, and the k is the number of iterations. However, calculating the new message vector length at every iteration incurs additional computational complexity because it involves computing a square and a multiplication.

In [9], another adaptive algorithm called two-length EMS (TL-EMS) on a message basis was proposed. It uses the reliability difference between the most and the second most reliable symbols among incoming messages from variable nodes when updating the check node. It selects the effective message vector length from the two predetermined message vector lengths. Also, when updating a variable node, they follow their own adaptive rules to truncate the message vector coming from each check node. However, the process of calculating the message vector length for each input node requires additional operations when updating the check node and variable node.

In this paper, we propose a low-complexity adaptive EMS algorithm. The TB-EMS algorithm has a simple adaptive rule than A-EMS algorithm. The proposed algorithm selects one of the two predetermined effective message vector lengths by comparing the $CNER$ with the check node error rate threshold ($CNER_{th}$). The $CNER_{th}$ is determined based on the error correction capability. Unlike A-EMS, the proposed algorithm employs only the two message vector lengths, one of which will be selected based a simple comparison instead of some complex operation. Experimental results show that the proposed algorithm reduces the decoding complexity with minimal performance degradation compared with the EMS algorithm.

II. BACKGROUNDS

A. *Non-binary Low Density Parity Check Codes*

NB-LDPC codes are defined by an M×N sparse parity check matrix H [11], [12] where M is the parity symbol length, and (N–M) is the information symbol length. The code rate R is (N–M)/N. The H matrix can be represented by a Tanner graph, which is a bipartite graph consisting of two node sets [13]. There are N variable nodes and M check nodes in the Tanner graph. If element $h_{i,j}$ of the H matrix has a non-zero value, the i^{th} check node ($i \in \{1, 2, 3, \ldots, M\}$) and the j^{th} variable node ($j \in \{1, 2, 3, \ldots, N\}$) in the corresponding Tanner graph are connected to each other. Each element $h_{i,j}$ has a value out of q symbols. The number of check nodes connected with a variable node is called d_v, and the number of variable nodes neighboring a check node is called d_c. When d_v and d_c have fixed values, such codes are called (d_v, d_c) regular NB-LDPC codes.

B. *Extended Min-Sum Algorithm*

Algorithm 1 describes the decoding steps of the EMS algorithm. Here are some notations used in Algorithm 1.

- v_j: the j^{th} variable node and c_i: the i^{th} check node
- $U_{j,i}(a)$: v_j to c_i message associated to symbol a.
- $V_{i,j}(a)$: c_i to v_j message associated to a.
- $H_c(i)$: set of variable nodes connected to c_i.
- $H_v(j)$: set of check nodes connected to v_j.
- $R_j(a)$: *a priori* information of v_j associated to a.
- $R_j^{post}(a)$: *a posteriori* information of v_j associated to a.
- $L(i)$: set of symbols satisfying c_i's parity check equation (2).

$$\sum_{j \in H_c(i)} h_{ij} a_j = 0 \tag{2}$$

where a_j means a symbol sent from v_j to c_i.
- $L(i|a_j=a)$: subset of $L(i)$ when $a_j = a$.

Algorithm 1. Decoding steps of EMS algorithm

Input: $R_j(a)$, k_{max}

Initialization:

Set $U_{j,i}(a) = R_j(a)$ and $k = 0$.

Iteration:

Check node update:

$$V_{i,j}(a) = \min_{\substack{(a_{j'})_{j' \in H(i)} \\ \in L(i|a_j = a)}} \left[\sum_{j' \in H_c(i) \setminus \{j\}} U_{j',i}(a_{j'}) \right]$$

Variable node update:

$$\bar{U}_{j,i}(a) = R_j(a) + \sum_{i' \in H_v(j) \setminus \{i\}} V_{i',j}(a)$$

$$U_{j,i}(a) = \bar{U}_{j,i}(a) - \bar{U}_{j,i}(0)$$

Post processing:

$$R_j^{Post}(a) = R_j(a) + \sum_{i' \in H_v(j)} V_{i',j}(a)$$

Tentative decision:

$$Z_j = arg\{min_{a \in GF(q)} R_j^{Post}(a)\}$$

Syndrome check

(if syndrome $S = Z \times H^T = 0$, return Z)

if $k = k_{max}$, decoding failure.

$k = k+1$, move to the check node update

Output: $\boldsymbol{Z_j}$

The EMS decoding algorithm largely consists of five steps: **Initialization**, **Check node update**, **Variable node update**, **Post processing** and **Tentative decision** as shown in Algorithm 1. All operations are based on the message vector of each node. The message vector used in the update process is composed of a reliability measure corresponding to the symbol. The measure of reliability is expressed as a log likelihood ratio (LLR) which is computed as (3).

$$R_j(a) = \log \frac{P(y_j = 0|x_j)}{P(y_j = a|x_j)}, a \in GF(q) \tag{3}$$

where x_j denotes the j^{th} symbol of the transmitted codeword, and y_j denotes the j^{th} symbol of the received codeword. The variable node is initialized using $R_j(a)$.

III. Proposed Algorithm

A. *Decoding Process*

The proposed algorithm is a low-complexity adaptive EMS algorithm called threshold-based EMS (TB-EMS) that selects the message vector length for the next decoding process by comparing the *CNER* value with a predetermined threshold value denoted by $CNER_{th}$.

The *CNER* values reveal a consistent tendency that is proportional to bit error rate (BER) [8]. Therefore, we may use the *CNER* value for adjusting the message vector length. The *CNER* can be calculated as shown in (4).

$$CNER = \frac{\#\ of\ unsatisfied\ check\ nodes}{\#\ of\ all\ check\ nodes} \tag{4}$$

The *CNER* value in (4) represents the ratio of *the number of unsatisfied check nodes* among *all check nodes*, where *the unsatisfied check node* means a check node whose syndrome check is failed. The syndrome check is performed with the most reliable symbols of each check node output.

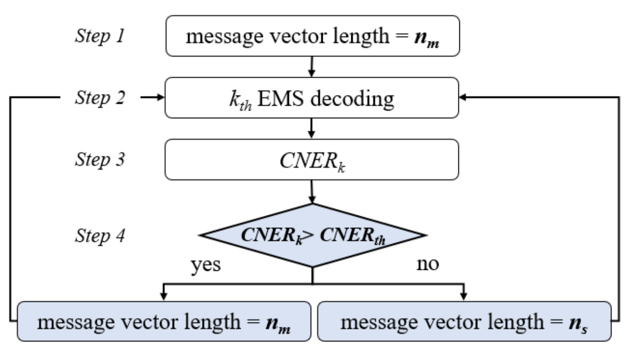

Figure 1. Decoding flow of the proposed algorithm

The flowchart of the proposed algorithm is shown in Fig.1. First, in *Step 1*, the proposed algorithm initializes the message truncation size to n_m. In *Step 2*, the decoding process of the EMS algorithm is performed at every iteration. After the k^{th} iteration, the $CNER_k$ value is calculated using (4) in *Step 3*. In *Step 4*, $CNER_{th}$ and $CNER_k$ are compared. When $CNER_k$ is smaller than $CNER_{th}$, a smaller message size is used for the next decoding process. The truncation rule of the proposed algorithm is as follows.

$$n_a{}^k = \begin{cases} n_m\ (CNER_k \geq CNER_{th}) \\ n_s\ (CNER_k < CNER_{th}) \end{cases} \text{ where } (n_s < n_m) \quad (5)$$

Unlike the A-EMS algorithm, the proposed algorithm employs only the two message vector lengths, one of which will be selected by a simple comparison operation as in (5).

B. *Determine the $CNER_{th}$*

In the proposed low-complexity decoding algorithm, determining $CNER_{th}$ appropriately is very crucial to achieve a good frame error rate (*FER*) performance. With respect to various combinations of the two vector lengths, we measure the *FER* performance and the average message size with respect to various *CNER* values to determine the best $CNER_{th}$ as shown in Fig. 2.

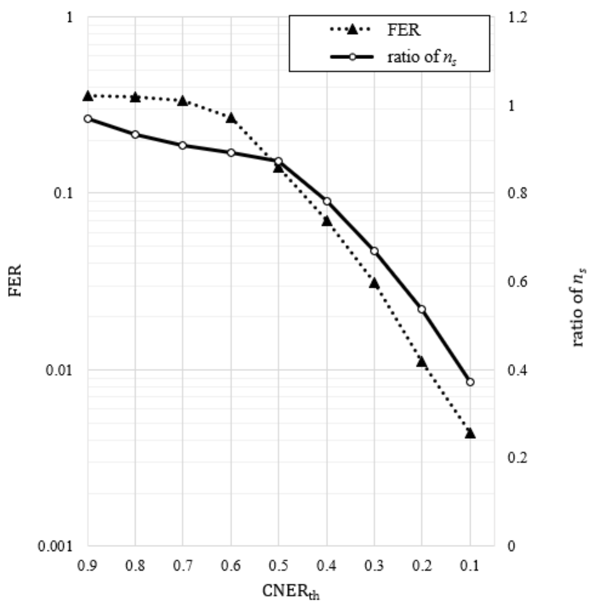

Figure 2. The *FER* and the *ratio of* n_s over various *CNERs* for (2, 12) regular NB-LDPC codes over GF(64) with *SNR*=2.5.

Fig. 2 shows the *FER* performance and the *ratio of* n_s according to various *CNER* values. The *ratio of* n_s represents the ratio of the n_s value as the vector length between the two truncation sizes (n_m, n_s) where $n_m > n_s$. Since the truncation size affects the decoding complexity, the *ratio of* n_s indicates the reduction amount of the decoding complexity. With a low $CNER_{th}$, the *FER* performance increases but the *ratio of* n_s decreases in our experiments. If the $CNER_{th}$ value goes below 0.1, the *FER* performance remains almost the same but the *ratio of* n_s decreases. Therefore, we set $CNER_{th}$ to 0.1 to maintain the *FER* performance while reducing the decoding complexity.

IV. EXPERIMENTAL RESULTS

We have implemented EMS, A-EMS, and the proposed TB-EMS algorithms to compare the *FER* performance and the average message size for transmissions over the binary phase shift keying (BPSK) modulation and the additive white gaussian noise (AWGN) channel. The maximum number of decoding iterations is set to 15, and 100,000 frames were examined to obtain the *FER* at each *SNR* for testing the decoding performance.

A. *Frame Error Rate*

The *FER* performances with (2, 4) regular NB-LDPC codes over GF(64) are shown in Fig. 3. The *FER* performance degradation of TB-EMS is less than 0.1dB compared with EMS using the same n_m. However, the *FER* performance of TB-EMS is better than that of A-EMS when n_s is equal to *A* in (1).

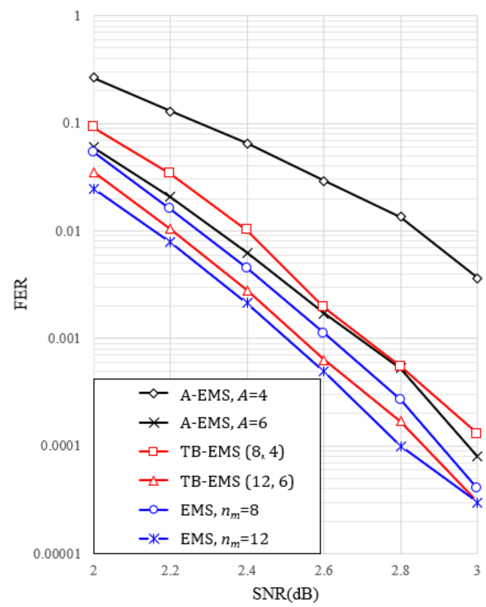

Figure 3. The *FER* performances of EMS, A-EMS, and TB-EMS for (2, 4) regular NB-LDPC codes.

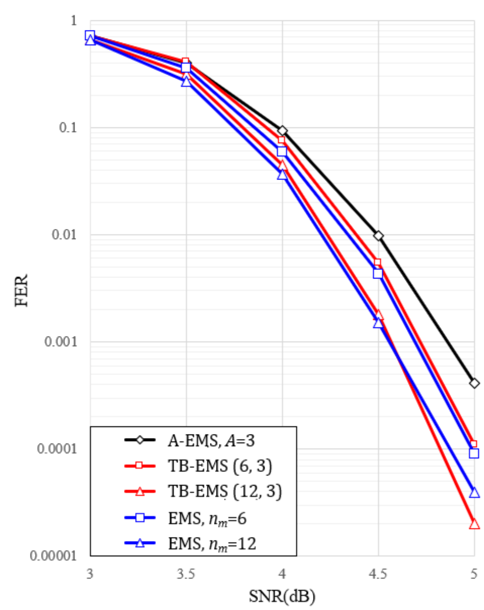

Figure 4. The *FER* performances of EMS, A-EMS, and TB-EMS for (2, 12) regular NB-LDPC codes.

The proposed TB-EMS algorithm shows better *FER* performances with higher code rates. The *FER* performances of the three algorithms with (2, 12) regular NB-LDPC codes over GF(64) are described in Fig. 4. The *FER* performance of TB-EMS with message vector length (n_m, n_s) is almost the same to that of EMS. And the *FER* performance degradation of TB-EMS is less than 0.15dB when compared with that of A-EMS.

B. *Average Message Size*

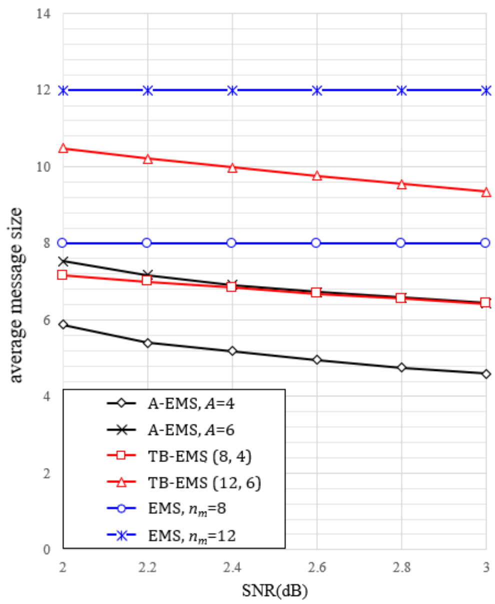

Figure 5. Average message size of EMS, A-EMS, and TB-EMS for (2, 4) regular NB-LDPC codes.

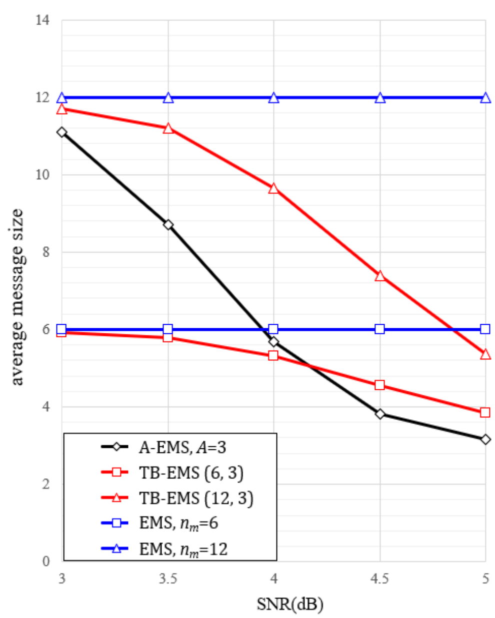

Figure 6. Average message size of EMS, A-EMS, and TB-EMS for (2, 12) regular NB-LDPC codes.

Next, the comparison results of the average message size of the three algorithms with (2, 4) regular NB-LDPC codes over GF(64) are shown in Fig. 5. The average message size of TB-EMS is bigger than that of A-EMS, but the *FER* performance is at least 0.3dB better as shown in Fig.3 when n_s is equal to *A*. The difference of the average message size between TB-EMS and A-EMS is less than 3.

The average message size of the three algorithms with (2, 12) regular NB-LDPC codes over GF(64) are shown in Fig. 6. When the code rate goes high, the smaller message vector length n_s of TB-EMS may be decreased leading to a smaller average size as shown in Fig. 6. We compare TB-EMS with the two message vector lengths (6, 3) and A-EMS with *A*=3. The average message size of TB-EMS is less than that of A-EMS by 4 when the *SNR* is less than 4.0. If the *SNR* goes higher, the average message size of TB-EMS slightly higher than A-EMS.

V. CONCLUSION

In this paper, a low-complexity adaptive EMS algorithm for NB-LDPC codes called threshold-based EMS (TB-EMS) is proposed. The proposed algorithm selects one of the two candidate message vector lengths by comparing the check node error rate (*CNER*) with a predetermined threshold value called $CNER_{th}$. The message truncation rule of the proposed TB-EMS is simpler than that of a *CNER*-based adaptive EMS algorithm called A-EMS. Experimental results show the frame error rate (*FER*) performance of TB-EMS is almost the same as that of the EMS algorithm while the decoding complexity is significantly low. The advantage is more pronounced when the *SNR* or the code rate goes high. The average message size is used as a metric to measure the effective decoding complexity. The average message size of TB-EMS algorithm is slightly bigger than that of A-EMS. However, the *FER* performance of TB-EMS is better than that of A-EMS by more than 0.5dB when code rate is 0.5 or higher.

Since the proposed method is tested on BPSK modulation, further experiments on other modulations are required to verify whether the proposed method is suitable or not.

ACKNOWLEDGMENT

This paper was supported by the Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (N0001883, The Competency Development Program for Industry Specialist).

REFERENCES

[1] M. Davey and D. J. C. MacKay, “Low-density parity check codes over GF(q)," IEEE Commun. Lett., vol. 2, no. 6, pp. 165-167, Jun. 1998.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[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.K. Elissa, “Title of paper if known,” unpublished.

[3] H.Wymeersch, H. Steendam, and M. Moeneclaey, “Log-domain decoding of LDPC codes over GF(q),” in Proc. 2004 IEEE Intern. Conf. on Commun., vol. 2, pp. 772-776, June 2004.R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

[4] Y. Hwang, K. Yang, and K. Cheun, “Low-latency low-complexity heap-based extended min-sum algorithms for non-binary low-density

parity check codes," IET Commun., vol. 9, ISS. 9, pp. 1191-1198, June 2015.

[5] E. Li, D. Declercq, and K. Gunnam, "Trellis-based extended min-sum algorithm for non-binary LDPC codes and its hardware structure," IEEE Trans, Commun., vol. 61, no. 7, pp. 2600-2611, Jul. 2013.

[6] V. Savin, "Min-max decoding for non-binary LDPC codes," presented at the IEEE Int. Symp. Inf. Theory, Toronto, ON, Canada, Jul. 2008.

[7] D. Declercq and M. Fossorier, "Decoding algorithms for nonbinary LDPC codes over GF(q)," IEEE Trans. Commun., vol. 55, no. 4, pp. 633-643, Apr. 2007.M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[8] X. Guan and Y. Fei, "Adaptive extended min-sum algorithm for nonbinary LDPC decoding," in Global Telecommunications Conference (GLOBECOM) 2011.D. Kornack and P. Rakic, "Cell Proliferation without Neurogenesis in Adult Primate Neocortex," Science, vol. 294, Dec. 2001, pp. 2127-2130, doi:10.1126/ science.1065467.

[9] Y. Hwang and S. Cho, "An Adaptive EMS Algorithm for Nonbinary LDPC Codes," in International Symposium on Information Theory (ISIT) 2017.

[10] W. Huang and L. Wang, "Low-Power Non-Binary LDPC Decoder Design via Adaptive Message Length Control Expliting Domian-Specific Information," J Sign Process Syst, May. 2020, doi: 10.1007/s11265-020-01533-2

[11] R. M. Tanner, "A recursive approach to low complexity codes," IEEE Trans. Inf. Theory, vol. 27, pp. 533–547, 1981.

[12] R. G. Gallager, Low-Density Parity-Check Codes. Cambridge, MA: MIT Press, 1963.

[13] X.-Y. Hu and E. Eleftheriou, "Binary representation of cycle Tanner graph GF(2q) codes," in Proc. IEEE Int. Conf. Commun., Paris, France, Jun. 2004, pp. 528–532.