Simplified Variable Node Unit Architecture for Nonbinary LDPC Decoder

Huyen Pham Thi, Hung Dao Tuan, Nghia Pham Xuan

Abstract— **Nonbinary low-density-parity-check (NB-LDPC) code outperforms their binary counterpart in terms of error correcting performance and error-floor property when the code length is moderate. However, the drawback of NB-LDPC decoders is high complexity and the complexity increases considerably when increasing the Galois-field order. In this paper, a simplified basic-set trellis min-max (sBS-TMM) algorithm that is especially efficient for high-order Galois Fields, is proposed for the variable node processing to reduce the complexity of the variable node unit (VNU) as well as the whole decoder. The decoder architecture corresponding to the proposed algorithm is designed for the (837, 726) NB-LDPC code over GF(32). The implementation results using 90-nm CMOS technology show that the proposed decoder architecture reduces the gate count by 21.35% and 9.4% with almost similar error-correcting performance, compared to the up-to-date works.**

Tóm tắt— **Các mã LDPC phi nhị phân (NB-LDPC) vượt trội so với các mã LDPC nhị phân về chất lượng sửa lỗi và thuộc tính lỗi san bằng khi chiều dài là trung bình. Tuy nhiên, nhược điểm của các bộ giải mã NB-LDPC là tính phức tạp cao và độ phức tạp tăng đáng kể khi bậc của trường Galois cao. Trong bài báo này, thuật toán Trellis Min-Max dựa trên tập cơ sở được đơn giản hóa được đề xuất cho xử lý nốt biến mà hiệu quả cho các trường Galois bậc cao để giảm độ phức tạp của khối nốt biến (VNU) cũng như cả bộ giải mã. Kiến trúc bộ giải mã tương ứng với thuật toán đề xuất được thiết kế cho mã NB-LDPC (837, 726) thông qua trường GF(32). Các kết quả thực hiện sử dụng công nghệ CMOS 90-nm chỉ ra rằng kiến trúc bộ giải mã được đề xuất giảm số lượng cổng logic 21,35% và 9,4% với chất lượng sửa lỗi gần như không thay đổi so với các nghiên cứu gần đây.**

Keywords— **NB-LDPC; Basic-set; Trellis min-max; VLSI design.**

Từ khóa— **NB-LDPC; Tập cơ bản; Trellis min-max; Thiết kế VLSI.**

This manuscript is received on July 1, 2019. It is commented on July 11, 2019 and is accepted on July 18, 2019 by the first reviewer. It is commented on July 16, 2019 and is accepted on July 22, 2018 by the second reviewer.

I. INTRODUCTION

Nonbinary low-density parity-check codes, which are defined over Galois Fields $GF(q)$ with ($q > 2$), outperform their binary counterpart in terms of error-correcting performance, burst error correction capability, and performance improvement in the error-floor region when the code length is moderate [1]. Nonetheless, the NB-LDPC decoding algorithms require complex computations, and their architectures have very high complexity and large memory requirements.

The works in [2, 3] show that NB-LDPC codes provide superior performance compared with the best optimized binary LDPC code over fading channels, and the combination of NB-LDPC code with high-order modulations improves not only the bandwidth efficiency but also the error-correction capability. Furthermore, the NB-LDPC codes demonstrate much promise for multilevel flash memory applications [4] because of the elimination of the error floor. However, the main disadvantage of NB-LDPC codes is their highly complex decoding algorithms and NB-LDPC decoder architecture.

For practical NB-LDPC decoder implementations, suboptimal algorithms such as extended min-sum (EMS) [5] and the min-max [6] algorithm have been proposed to reduce the complexity of the CNU as the main bottleneck of the NB-LDPC decoder. The min-max algorithm [6] is interesting because it uses comparisons instead of additions [5] in the check node processing, which not only reduces the hardware complexity but also prevents the numerical growth of the decoder.

Recently, some optimal min-max algorithms [7-9] have been proposed to improve both the throughput and the

complexity. The relaxed trellis min-max (R-TMM) algorithm [9] introduced the trellis representation and the minimum basis for check node processing to remove computing the forward-backward messages in [6]. However, the check node processing is sequentially processed which requires a large number of clock cycles. In [10], a simplified trellis min-max (STMM) algorithm was proposed to improve the throughput of the min-max decoders with less complexity by means of an extra column inserted to the original trellis. In [11], the one minimum-only TMM (OMO-TMM) algorithm was introduced on the basis of the STMM algorithm to reduce the CNU complexity by obtaining only one minimum and estimating the second one. In these works [10, 11], $q \times dc$ check node output messages are exchanged between the check node and the variable nodes. For high-order GFs or high-rate NB-LDPC codes, the amount of exchanged messages increases and the memory requirement is large, which limits the maximum throughput of the decoders and leads to a significant increase in the decoder area.

To overcome the drawbacks of [10, 11], the works in [12, 13] proposed to simplify the CNU architecture and reduce the exchanged messages with the almost similar error-correcting performance. In [14, 15], the approximated TMM algorithms are introduced to further decrease the number of intrinsic information at the cost of some error-correcting performance loss. In [16], a basic-set trellis min-max (sBS-TMM) algorithm, which is especially efficient for high-order Galois Fields, has been introduced to reduce the exchanged messages to a factor of $\log_2 q$ with a negligible performance loss.

In this paper, a simplified basic-set trellis min-max algorithm is proposed for the variable node processing to reduce the decoder complexity with the almost similar error correcting performance, compared to the existing decoding algorithm. The decoder architecture of a (837, 726) NB-LDPC code over GF(32) was performed using the sBS-TMM algorithm to demonstrate the efficiency of the proposal.

II. REVIEW OF NB-LDPC DECODING ALGORITHM

A. NB-LDPC codes

NB-LDPC codes, which are a kind of linear block code, are defined by a sparse parity-check matrix **H** having M rows and N columns. Let h_{mn} be a nonzero element of the matrix **H** that belongs to the GF(q=2^p). Furthermore, NB-LDPC codes are presented by a Tanner graph corresponding to the matrix **H**, where N columns correspond to N variable nodes, and M rows correspond to M check nodes. Let d_v and d_c be the variable node degree (column weight) and the check node degree (row weight) of matrix **H**, respectively. A regular NB-LDPC code is considered in this paper with the fixed values of d_v and d_c.

B. NB-LDPC Decoding Algorithm

Algorithm 1: Layered Min-Max Decoding Algorithm [16]

Input:

$L_n(a) = \ln(\Pr(c_n = z_n \mid channel) / \Pr(c_n = a \mid channel))$;

$Q_n^{1,0}(a) = L_n(a); R_{mn}^0(a) = 0; k = 1$

1: while $k \le I_{max}$ do

2: for l = 1 to M do

3: $R_{mn}^{k-1,l}(a) = DN\{z_n^*, E(a), B^*\}$;

4: $\widetilde{Q}_{mn}^{k,l}(a) = Q_n^{k,l-1}(h_{mn}\, a) - R_{mn}^{k-1,l}(a)$

5: $\widetilde{Q}_{mn}^{k,l} = \min_{a \in GF(q)} (\widetilde{Q}_{mn}^{k,l}(a))$

$z_n = \arg\min(\widetilde{Q}_{mn}^{k,l}(a))$

6: $Q_{mn}^{k,l}(a) = \widetilde{Q}_{mn}^{k,l}(a) - \widetilde{Q}_{mn}^{k,l}$

7: $\{z_n^*, E(a), B^*\} = BS\text{–}TMM\left\{Q_{mn}^{k,l}(a), z_n\right\}_{n \in N(m)}$

8: $R_{mn}^{k,l}(a) = DN\{z_n^*, E(a), B^*\}$

7: $Q_n^{k,l}(h_{mn}^{-1}\, a) = Q_{mn}^{k,l}(a) + R_{mn}^{k,l}(a)$

8: end for

9: end while

Output: $\widetilde{c}_n = \arg\min(Q_n^{k,l}(a))$

A horizontal layered decoding algorithm, which is well known for higher convergence speed compared to the flood decoding

algorithm, is applied in this paper. Algorithm 1 presents the layered basic-set trellis min-max (BS-TMM) decoding algorithm for the NB-LDPC codes [16].

Symbols c_n and z_n define the n-th reference symbol of a received codeword and the n-th hard-decision symbol with the highest reliability, respectively. Starting the decoding process is implemented by obtaining the log-likelihood ratio (LLR) vectors $L_n(a)$ with a size of q that are the channel information. At the first layer of the first iteration, the *a posteriori* information as $Q_n(a)$ corresponding the variable node n is equal to $L_n(a)$. The check node to variable node (C2V) messages $R_{mn}(a)$ are equal to zero. k and l define the loop index for k-th iteration and the layer index for l-th layer, respectively. In addition, the decompression network (DN) in step 3 and step 8 is implemented in the variable node processor to generate the C2V messages $R_{mn}(a)$ from outputs of the CNU architecture. The DN has three parts: 1) generating the LLR values of the extra column $\Delta Q(a)$ and the path information d(a) with maximum of p deviations from the basic set; 2) generating the C2V messages in the delta domain as $\Delta R_{mn}(a)$ on the basis of $\Delta Q(a)$, $E(a)$, and $d(a)$; 3) converting the C2V messages from delta to normal domain. It is noted that two DNs are required in the variable node processor.

However, the proposed decoder area is much lower than that of the conventional decoders [10], [11]. Then, the variable node to check node (V2C) messages $\Delta\tilde{Q}_{mn}(a)$ are calculated from the $Q_n(a)$ messages permuted using the nonzero element hmn of matrix **H**, as shown in step 4. The normalization of V2C messages are performed as shown in step 5 and 6 to stabilize the numerical growth that the LLR value of the symbol with highest possibility in each vector is equal to zero. Step 7 presents the computation of the basic-set messages and the information used to update the C2V messages using the BS-TMM function applied for the check node processing. Step 9 involves in the computation of the updated messages $Q_n(a)$, which is undergone the reverse permutation before processing a new layer. The decoding process is repeatedly implemented until the maximum number of iteration Imax is reached. Finally, the output codeword $\tilde{c}_n(a)$is the most reliable symbol corresponding to $Q_n(a)$ message.

C. Basic-set Trellis Min-max Algorithm

In this section, the BS-TMM algorithm [16] is illustrated, which greatly reduces the complexity and the memory requirement for check node processing as well as the exchanged messages between check nodes and variable nodes with a negligible error-correcting performance loss. The BS-TMM algorithm is highly efficient for designing the decoders with high-order GFs. Without loss of generality, the Galois-field GF(q) with $q = 2^p$ including q elements such as $\{0, \alpha^0, \alpha^1, ..., \alpha^{q-2}\}$ is considered in our work. For each Galois-field GF(2^p), any field element is uniquely represented by the linear addition of p independent field elements. To take advantage of this, a set of only $p = \log_2 q$ independent field elements with the smallest LLRs, called the basic set B^*, are generated in the check node processing instead of $(q-1)$ nonzero field elements in the extra column $\Delta Q(a)$ [14, 17]. Then, construction of the $\Delta Q(a)$, is implemented in the variable node processing based on the basic set B^*.

The BS-TMM algorithm is represented in *Algorithm* 2.

Algorithm 2: Basic-Set TMM Algorithm [16]

Input:

$z_n = \arg\min(Q_{mn}(a))_{a\in GF(q)}; \forall n \in N(m)$

1: $\Delta Q_{mj}(\eta = a \oplus z_j) = Q_{mj}(a); (0 \le j < d_c)$

2: $\beta = \sum_{j=0}^{d_c-1} z_j \in GF(q)$

3: $\{ml(a), I_{col}(a), m2(a)\} = \Psi\left\{\Delta Q_{m,k}(a)\middle|_{k=0}^{d_c-1}\right\}$

4: $B^* = \{ml_l^*, I_l^*, a_l^*\}_{1\le l\le p} = \Phi\{ml(a), I_{col}(a)\}_{1\le a<q}$

5: $E(a) = \begin{cases} ml(a) \ if \ a = a_l^* (1 \le l \le p) \\ m2(a) \ otherwise \end{cases}$

Output: $E(a) = \begin{cases} B^* \\ E(a) \\ z_n^* = z_n \oplus \beta \end{cases}$

The first step involves the transformation of the input messages from the normal domain

$Q_{mn}(a)$ to the delta domain $\Delta Q_{mn}(a)$. This transformation ensures that the most reliable symbols are always in the first index corresponding to the GF symbol 0, and the rest of the indexes are in order of $\{\alpha^0, \alpha^1, ..., \alpha^{q-2}\}$. Step 2 relates to the computation of the syndrome β using the most reliable symbols z_n from V2C messages. In step 3, the first minimum value $m1(a)$ and its column index $m1_{col}(a)$, as well as the second minimum value $m2(a)$ for each trellis row are calculated using the function ψ. Step 4 computes the basic set $B^* = \{m1_l^*, I_l^*, a_l^*\}_{1\le l\le p}$ including $3 \times p$ values (p LLR values, p column indexes, and p field elements), based on the minimum values $m1(a)$ and their column indexes $I_{col}(a)$ ($1 \le a < q$). Finding the basic set B^* is given by the ϕ function in *Algorithm* 2. Step 5 relates to calculating the complement values in set $E(a)$. The complement values for p field elements, which belong to the basic-set B^*, are assigned to the second minimum values $m2(a)$. For the remaining field elements, the complement values are assigned to the minimum values $m1(a)$. Finally, the output of the check node processing includes three sets B^*, $E(a)$, and z_n^* with a size of $3\times p + (q-1) + d_c$ values, which are used for generating the C2V messages in the variable node processing.

III. SIMPLIFIED BASIC-SET TRELLIS MIN-MAX DECODING ALGORITHM

A. Simplified basic-set Trellis min-max decoding algorithm

In the variable node processing, *Algorithm* 3 shows the simplified extra column construction $\Delta Q(a)$ based on the output sets of the check node processing, including B^*, $E(a)$, and z_n^*. The extra column $\Delta Q(a)$ and the path information $d(a)$ are calculated in steps 1 to 7. For p field elements, which belong to the basic set B^*, the $\Delta Q(a)$ value is the most reliable LLR $m1_l^*$, and the path information $d(a)$ has one deviation at the column index I_l^*, with $1 \le l \le p$.

Algorithm 3: Simplified Extra Column Construction $\Delta Q(a)$

Input:

$B^* = \{m1_l^*, I_l^*, a_l^*\}_{1\le l\le p}$

1: for a=1 to q-1 do

2: if $a = a_l^* (1 \le l \le p)$ then

3: $\Delta Q(a) = m1_l^*; d(a) = \{I_l^*\}$

4: else if $a = a_1^* \oplus a_2^* \oplus ... \oplus a_s^* (2 \le s \le p)$ then

5: $\Delta Q(a) = m1_p^*; d(a) = \{I_1^* \cup I_2^* \cup ... \cup I_s^*\}$

6: end if

7: end for

From our observation, a proximate approach is proposed for calculation of the remaining field elements. In [16], the remaining field elements are computed on the basis of all possible combinations of the field elements in the basic set B^*. Their $\Delta Q(a)$ values are the maximum LLR value from the LLR values corresponding to the combined field elements as $\Delta Q(a) = \max(m1_1^*, m1_2^*, ..., m1_s^*)$ with ($2 \le s \le p$), and their path information $d(a)$ has more than one deviation and a maximum of p deviations. It is remarked that there are $(q-1) - p$ remaining field elements, which are generated on the basic of combinations of the p field elements in the basic set B^*. For each remaining field element corresponding to each combination, $\Delta Q(a)$ value is the maximum LLR value from the LLR values of the combined field elements. Furthermore, the number of possible combinations including the last field element ap is the largest in comparison to the other field elements in the basic set. It is clear that the LLR value $m1_p^*$ corresponding to the last field element a_p* accounts for a larger factor than other field elements. Therefore, in this work, an approximate method is proposed to assign the LLR value of the last field element $m1_p$* to the LLR values of the remain field elements in the extra column $\Delta Q(a)$, as shown in Step 5 of *Algorithm* 3.

For example, in GF(8), the basic set B^* consists of three field elements such as $\{a_1^*, a_2^*, a_3^*\}$. Other nonzero field elements are constructed as $\Delta Q(a_1^*+a_2^*) = \max(m1_1^*, m1_2^*) = m1_2^*$; $\Delta Q(a_1^*+a_3^*) = \max(m1_1^*, m1_3^*) = m1_3^*$; $\Delta Q(a_2^*+a_3^*) = \max(m1_2^*, m1_3^*) = m1_3^*$; $\Delta Q(a_1^*+a_2^*+a_2^*) = \max(m1_1^*, m1_2^*, m1_3^*) = m1_3^*$.

It is obvious that most of LLR values of the remaining field element in the extra column are equal to $m1_3^*$ that is the LLR value of the last field element in the basic set. Thus, in this

work, $m1_3^*$ value is assigned to $\Delta Q(a)$ values of the all remaining field elements.

B. Performance Analysis

To demonstrate the error-correcting performance of the sBS-TMM decoding algorithm, Fig. 1 illustrates the frame error rate (FER) performance for (837, 726) NB-LDPC code over GF(32) with $d_v = 4$ and $d_c = 27$ under the additive white Gaussian noise (AWGN) channel and binary phase shift keying (BPSK) modulation. As shown in Fig. 1, the floating-point simulation result of the sBS-TMM algorithm with 15 iterations shows the almost similar error-correcting performance, compared to the BS-TMM algorithm [16], and a performance loss at almost 0.1 dB, compared to the two-extra-column TMM (TEC-TMM) algorithm [12] and the STMM algorithm [10]. Because the proposed sBS-TMM algorithm discards the computation of the $(q - 1) - p$ values of the remaining field elements in the extra column $\Delta Q(a)$, the decoder architecture corresponding to the sBS-TMM algorithm obtains a low computational complexity, a large area reduction, and a significant improvement in throughput.

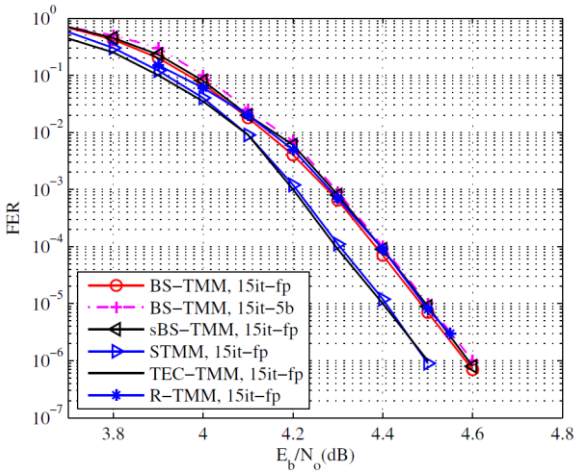

Fig 1. FER performance of the (837, 726) NB-LDPC code over GF(32) under the AWGN channel at 15 iterations.

IV. REDUCED-COMPLEXITY DECODER ARCHITECTURE

Fig. 2 shows the top-level decoder architecture for the proposed layered decoding algorithm, where one row of **H** corresponding to one layer is processed in one clock cycle.

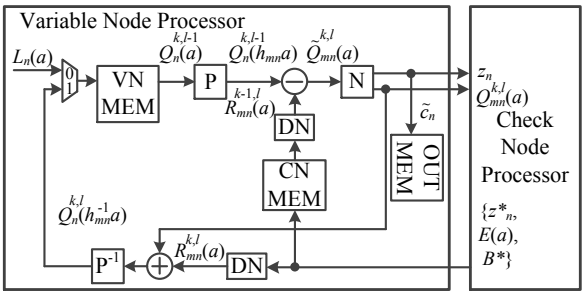

Fig. 2. Top-level NB-LDPC decoder architecture based on the sBS-TMM algorithm [16]

It can be seen that the decoder architecture is divided into a variable node processor and check node processor. To start the decoding process, the LLR messages from channel information $L_n(a)$ are loaded in variable node memory (VNMEM). From the next layer and next iteration, the output messages of the variable node processor $Q^{k,l}{}_n(a)$ are stored in the VNMEM.

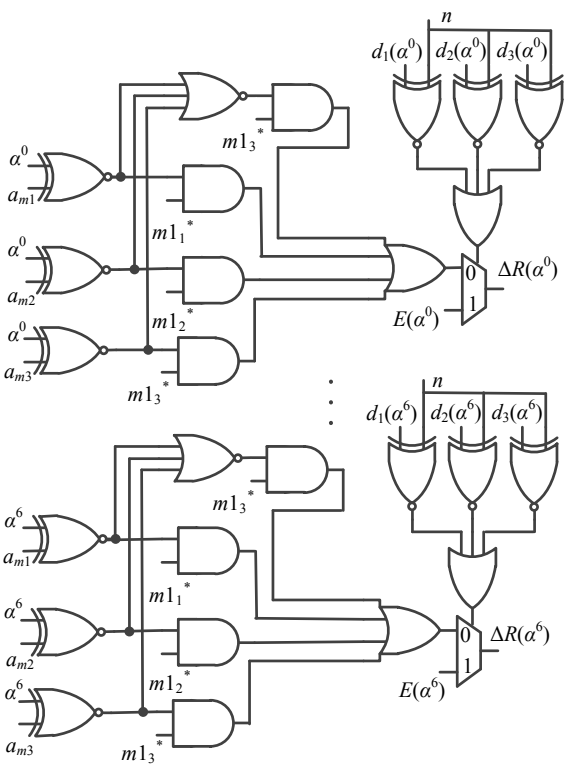

Fig. 3. Proposed C2V generator based on sBS-TMM algorithm for GF(8)

The VNMEM includes dc memories with a depth of $(q-1)$ as the size of the circulant permutation matrix [18] and a width of $q \times w$ bits. For each decoding time, one address is read and one address is written from each memory. Other modules such as permutation **P**, de-permutation **P**−1, and normalization **N** are

similar to the ones in [16]. It is remarked that the decompression network (DN) corresponding to *Algorithm* 3 is implemented in the variable node processor to generate the C2V messages $R_{mn}(a)$ from outputs of the CNU architecture. The DN has three parts: 1) generating the path information $d(a)$ with maximum of p deviations based on the basic set $B^* = \{m1_l^*, I_l^*, a_l^*\}_{1 \le l \le p}$; 2) generating the Fig. 3. Proposed C2V generator based on sBS-TMM algorithm for GF(8). C2V messages in the delta domain as _Rmn(a) based on the basic set B^*, complement set $E(a)$, and $d(a)$; 3) converting the C2V messages from delta to normal domain. It is noted that two DNs are required in the variable node processor. However, the proposed decoder area is much lower than that of the conventional decoders [10, 11]. Figure 3 shows the proposed C2V generator in the DN module, which is based on the sBS-TMM algorithm for each C2V message vector in GF(8). Since the extra-column constructor is eliminated, the complexity of the proposed C2V generator is significantly reduced. For three field elements in the basic set, the C2V messages are either the LLR values in the basic set or the complement values $E(a)$, which depend on the path information. It is clear that for remaining field elements, the C2V messages are either the LLR value of the last field element in the basic set as $m1_3^*$ or the complement values $E(a)$.

V. COMPLEXITY AND COMPARISON

In this work, the (837, 726) NB-LDPC code over GF(32) is constructed by the submatrix (d_v, d_c) = (4, 27) and a CPM of size $(q - 1) \times (q - 1)$ [18]. To illustrate the efficiency of our proposal for NB-LDPC codes, the complete decoder architectures were implemented for (837, 726) NB-LDPC code over GF(32).

TABLE 1. IMPLEMENTATION RESULTS OF THE PROPOSED DECODER FOR THE (837, 726) NB-LDPC CODE OVER GF(32) IN A 90-NM CMOS PROCESS

Algorithm	**RMM**	**T-MM**	**BS-TMM**	**sBS-TMM**
	[9]	**[15]**	**[16]**	**[proposed]**
Report	Syn.	Post-layout	Post-layout	Syn.
Quantization	5 bits	6 bits	5 bits	5 bits
Gate count (NAND)	871K	1.06M	756K	685K
f_{clk} (MHz) (Synthesis)	200	345	393	397
Iterations	8	8	8	8
Throughput (Mbps) (Layout)	154	1071	1261	1264
Efficiency (Mbps/M gates)	176.8	1010.4	1668	1845

The synthesis results of the proposed decoder for the (837, 726) NB-LDPC code and the comparison with previous works are presented in Table I. It can be seen that the proposed decoder reduces the gate count by 35.38% and achieves almost twice times higher efficiency, compared to the work from [15]. Compared to the works with using the basic sets of the reliable messages [9, 16], the proposed decoder improves not only the gate count but also the throughput because of a significant reduction of the complexity in the VNU as well as the whole decoder architecture. Therefore, the proposed decoder reduces the gate count by 21,35% and 9,4%, respectively. Moreover, the proposed decoder exhibits almost 10.61% higher efficiency, compared to the work in [16].

VI. CONCLUSION

In this paper, a simplified sBS-TMM algorithm is proposed for the NB-LDPC codes to reduce the decoder complexity. The decoder architecture corresponding to the proposed algorithm is designed to demonstrate the efficiency of the proposal. The implementation results show a gate count reduction of 21.35% and 9.4% with the almost similar error-correcting performance.

ACKNOWLEDGMENT

This work was supported by National Laboratory of Information Security, Ha Noi, Viet Nam.

REFERENCES

[1]. H. C. Davey and D. J. MacKay, "Low-density parity check codes over GF(q)," in Information Theory Workshop, pp. 165-167, Jun. 1998.

[2]. R. Peng and R.-R. Chen, "WLC45-2: Application of nonbinary LDPC codes for communication over fading channels using higher order modulations," in IEEE Global

Telecommunications Conference (GLOBECOM'06), pp. 1-5, Dec. 2006.

[3]. M. Arabaci, I. B. Djordjevic, L. Xu, and T. Wang, "Nonbinary LDPCcoded modulation for high-speed optical fiber communication without bandwidth expansion," IEEE Photonics Journal, vol. 4, no. 3, pp. 728-734, Jun. 2012.

[4]. C. A. Aslam, Y. L. Guan, and K. Cai, "Non-binary LDPC code with multiple memory reads for multi-level-cell (MLC) flash," in Asia-Pacific Signal and Information Processing Association, Annual Summit and Conference (APSIPA), pp. 1-9, 2014.

[5]. 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.

[6]. V. Savin, "Min-max decoding for non binary LDPC codes," in in Proc. IEEE Int. Symp Inf. Theory, Toronto, ON, Canada, pp. 960-964, Jul. 2008.

[7]. X. Zhang and F. Cai, "Reduced-complexity decoder architecture for nonbinary LDPC codes," IEEE Trans. Very Large Scale Integration (VLSI) Syst., vol. 19, no. 7, pp. 1229-1238, 2011.

[8]. K. He, J. Sha, and Z. Wang, "Nonbinary LDPC code decoder architecture with efficient check node processing," IEEE Trans. Circuits Syst. II, Express Briefs, vol. 59, no. 6, pp. 381-385, 2012.

[9]. F. Cai and X. Zhang, "Relaxed min-max decoder architectures for nonbinary low-density parity-check codes," , IEEE Trans. Very Large Scale Integration (VLSI) Syst., vol. 21, no. 11, pp. 2010-2023, Nov. 2013.

[10]. J. O. Lacruz, F. Garc´ıa-Herrero, D. Declercq, and J. Valls, "Simplified trellis min-max decoder architecture for nonbinary low-density paritycheck codes," IEEE Trans. Very Large Scale Integration (VLSI) Syst., vol. 23, no. 9, pp. 1783-1792, Sep. 2015.

[11]. J. O. Lacruz, F. Garc´ıa-Herrero, J. Valls, and D. Declercq, "One minimum only trellis decoder for non-binary low-density parity-check codes," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 62, no. 1, pp. 177-184, Jan. 2015.

[12]. H. P. Thi and H. Lee, "Two-extra-column trellis min–max decoder architecture for nonbinary LDPC codes," IEEE Trans. Very Large Scale Integration (VLSI) Syst., vol. 25, no. 5, pp. 1787-1791, May. 2017.

[13]. J. O. Lacruz, F. Garcia-Herrero, M. J. Canet, J. Valls, and A. P´erez-Pascual, "A 630 Mbps non-binary LDPC decoder for FPGA," in Circuits and Systems (ISCAS), 2015 IEEE International Symposium, pp.1989-1992, 2015.

[14]. J. O. Lacruz, F. Garc´ıa-Herrero, M. J. Canet, and J. Valls, "Highperformance NB-LDPC decoder with reduction of message exchange," IEEE Trans. Very Large Scale Integration (VLSI) Syst., vol. 24, no. 5, pp. 1950-1961, May. 2016.

[15]. J. O. Lacruz, F. Garca-Herrero, M. J. Canet, and J. Valls, "Reducedcomplexity nonbinary LDPC decoder for high-order galois fields based on trellis min–max algorithm," IEEE Trans. Very Large Scale Integration (VLSI) Syst., vol. 24, no. 8, pp. 2643-2653, Aug. 2016.

[16]. H. P. Thi and H. Lee, "Basic-set trellis min–max decoder architecture for nonbinary ldpc codes with high-order galois fields," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 26, no. 3, pp. 496-507, 2018.

[17]. J. O. Lacruz, F. Garc´ıa-Herrero, and J. Valls, "Reduction of complexity for nonbinary LDPC decoders with compressed messages," IEEE Trans. Very Large Scale Integration (VLSI) Syst., vol. 23, no. 11, pp. 2676-2679, Nov. 2015.

[18]. B. Zhou, J. Kang, S. Song, S. Lin, K. Abdel-Ghaffar, and M. Xu, "Construction of non-binary quasi-cyclic LDPC codes by arrays and array dispersions," IEEE Trans. Commun., vol. 57, no. 6, pp. 1652-1662, Jun. 2009.

ABOUT THE AUTHOR

PhD. Huyen Pham Thi

Workplace: University of Transport Technology.

Email: phamhuyenmta87@gmail.com

The education process: received the B.S. degree from the Department of Information and Communication Engineering, Military Technical Academy, Ha Noi, Vietnam, in 2011. She achieved Ph.D. degree with the Department of Information and Communication Engineering from Inha University, Incheon, South Korea, in 2018.

Research today: algorithms and VLSI architecture design for digital signal processing, forward error correction architectures, and communication systems. Currently, she is working on hardware security and Artificial Intelligence for Natural Language Processing.

PhD. Hung Dao Tuan

Workplace: National Laboratory for Securing Information.

Email: daotuanhung@gmail.com

The education process: He received the B.S. degree from the Department of Electronic-electric engineering, Military Technical Academy, Ha Noi, Viet Nam, in 1998 and his M.S degree from Deparment of Information Technology, Melbourne University, Australia, in 2002. He achieved Ph.D. degree with the Military Institute of technology and Science, Ha Noi, Viet Nam in 2018.

Research today: His research interests are encryptographic and decryptographic algorithms based on ECC and its' application for information security. Currently, he is in charge of head of National Laboratory for Securing Information.

PhD. Nghia Pham Xuan

Workplace: Military Technical Academy.

Email: nghiapx@mta.edu.vn

The education process: He received the B.S. and M.S degree from the Department of Information and Communication Engineering, Military Technical Academy, Ha Noi, Vietnam, in 1998 and 2002, respectively. He achieved Ph.D. degree with the Department of Information and Communication Engineering from Ryazan University, Russia, in 2007.

Research today: His research interests are channel coding algorithms and its' application for information security.