BER Performance Study of Column Weight Two Non-Binary LDPC Codes with Predetermined Girth

Nut Tantibut[1], Ambar Bajpai[2], Kritsada Mamat[3], Tharathorn Phromsa-ard[4],
Watid Phakphisut[5], Piya Kovintavewat[6] and Lunchakorn Wuttisittikulkij[7]
[1,2,4,7] Department of Electrical Engineering, Faculty of Engineering, Chulalongkorn University
254 Phayathai Road, Pathumwan, Bangkok, 10330, Thailand
[3] Department of Electrical Engineering, Faculty of Engineering, Kasetsart University
50 Ngam Wong Wan Road, Lat Yao, Chatuchak, Bangkok 10900, Thailand
[5] Bangkok Suvarnabhumi University
489 Prachapattana Road, Tapyao, Ladkrabang, Bangkok 10520, Thailand
[6] Data Storage Technology Research Center, Nakhon Pathom Rajabhat University
85 Malaiman Road, Muang, Nakhon Pathom 73000 Thailand
E-mail: [1]Nut.Tant@student.chula.ac.th, [2]ambarbajpai@gmail.com, [3]mkritsada1@gmail.com,
[4]Tharathorn.P@student.chula.ac.th, [5]phwatid@gmail.com, [6]piya@npru.ac.th, [7]Lunchakorn.W@chula.ac.th

Abstract: In this paper, we constructed column weight two parity-check matrix by imposing pre-defined target girth and code length. Then the constructed binary parity-check matrix is transformed into non-binary matrix by using a randomly generated elements in Galios field GF(q), where q=2^p. We investigate The bit-error rate performance (BER) for higher orders of Galios field GF(q) as compare to its binary counterpart. We found that the column weight two LDPC codes with higher orders of GF(q) substantially improve BER performance and convergence speed than low values of GF(q).

1. Introduction

Since Claude Shannon proposed the theory of mathematical constraints for channel capacity, enormous potential research has been carried out on channel coding in digital communication system. After more than one decade, LDPC codes were first introduced by Gallagar [1] in 1962. Since the analytical tools weren't available at that time, these codes were ignored almost three decades.. Later these codes were rediscovered by Mackay and Neal in 1996 [2]. They show that the performance of LDPC codes approaches the Shannon's limit. Afterward researchers focused on this potential forward error correction (FEC) codes in multiple domains of LDPC codes. Various standards such as IEEE 802.11e, IEEE 802.11n, WiMAX, and DVB-S2/T2 have adopted LDPC codes [3]. Today, LDPC codes are considered as the most eligible channel codes for future generation high data rate communication and various practical applications. Development of LDPC codes have been also studied widely in current decade.

This paper focuses on non-binary LDPC codes, which are derivative of LDPC over Galois field GF(q), where q =2^p, p is integer number. Non-binary LDPC codes were first investigated by Davey and MacKay in 1998 [4], whose performance is much better than its binary counterpart. A most widely used sum-product algorithm (SPA), which is used in binary LDPC codes, causes the high computation of LDPC decoding, almost infeasible for higher order of q. It is shown that complexity can be deduced to $O(q\log_2 q)$ if we transferred a SPA algorithm for GF(q) into a frequency domain computation [5]. Although many researchers are working on the topic of non-binary LDPC codes, there is still a lot of works needed to be done for non-binary LDPC codes.

A girth is one of the important constraints for designing a good LDPC code because a large girth facilitates an iterative decoding and imposes a respectable minimum distance which can improve the decoding performance at high signal-to-noise ratio (SNR) scenario [7]. Therefore, this work aims the constructing good column weight two non- binary LDPC codes by imposing upon a target girth as a major constraint.

This paper is organized as follows. Section II summarizes non-binary LDPC code including encoding and decoding procedure. Section III explains our proposed method and Section IV gives simulation details and results. Finally, Section V concludes this paper.

2. Non-Binary LDPC Codes

LDPC codes are a class of linear block codes, which can be defined by a sparse parity-check matrix **H** of size $M \times N$, where M is the number of rows and N is the number of columns. In general, the **H** matrix can also be represented by a Tanner graph [8]

2.1 Construction of Non-Binary LDPC Codes

A non-binary LDPC consists of a sparse parity-check matrix over finite field GF (q). David and Mackay presented an idea of LDPC over finite fields (where $q > 2$) [4]. Meaningfully, it shows better performance than its binary counterpart.

If the array **H** viewed as a matrix has a constant column weight γ and a constant row weight ρ, the code given by the null space of **H** is said to be (γ, ρ)-regular, otherwise it is said to be irregular [9]. Row-Column (RC) constraint ensures that the Tanner graph of the LDPC code given by the null space of **H** has a girth of at least 6 and that the minimum distance of the code, if (γ, ρ)-regular, is at least γ+1 [9--11]. The distance bound is tight for regular LDPC codes whose parity-check matrices have large column weights and row redundancies, such as the algebraic LDPC codes constructed using finite fields, finite geometries, and

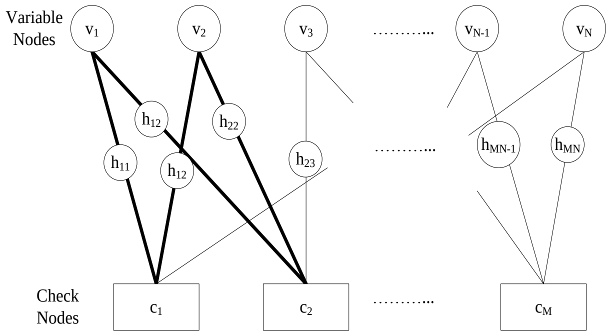

Figure 1. A Tanner graph of non-binary LDPC codes.

combinatorial designs. Figure 1. shows the Tanner graph for non-binary LDPC codes.

2.2 Decoding of Non-Binary LDPC

There are number of decoding algorithms discussed in literature [4] for binary and non-binary LDPC codes as we can summarize following steps.

a) Initialization: By using the received vector r, variable nodes are initially assigned with the likelihoods of channel reliability.

b) Check node update: This step also called as horizontal step. Each check nodes updated using the likelihoods message from adjacent variable nodes except considering updated check nodes. **Q** matrix construction.

c) Variable node update: This step is known as vertical update, in this step, variable nodes receive message from adjacent check nodes. **R** matrix construction.

d) Iterative decoding: Most likelihood value of code word $\hat{c}_n$ is computed with the step a) and variable nodes messages. Decoded code word is valid only if it satisfies $c.\mathbf{H}^T = 0$. In case of no valid code word produced, decoding process stopped after certain number of iterations.

2.3 FFT Based SPA decoding for GF (*q*)

For decode non-binary LDPC codes, SPA algorithm for binary LDPC can be extended with the cost of increased in decoding complexity as value of *q* increases. **Q** matrix in case of GF (*q*) becomes more complex to evaluate. In horizontal step, as more possible non-binary sequences needs to satisfy parity- check constraints, similarly **R** vertical matrix from **Q** matrix becomes much more complicated. Permutation and depermutation required in case of non-binary LDPC. Cyclic shift of the likelihoods in downwards called permutation and upwards likelihoods cyclic shift is called depermutations. FFT used in [10] to perform the computation of the check nodes update in the frequency domain for simple product form transforms from convolution in mathematical implications. By this method the complexity in horizontal step for check node update meaningfully reduced [10--11]. In general, parity check equations are of the form satisfies (1).

$$\sum_{j=1}^{N} h_{ij} c_j = 0 \qquad (1)$$

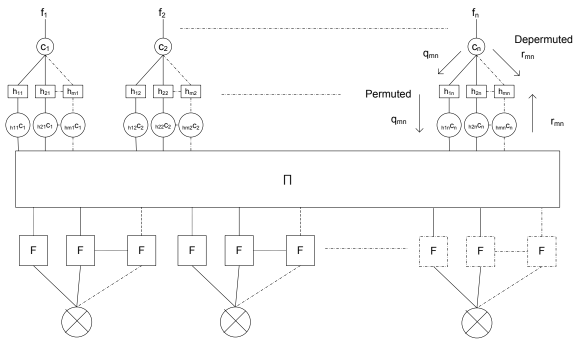

Figure 2. Generalized factor graph of a non-binary LDPC code using FFTs operations

Where parity check matrix h_{ij} and $c_j \in$ GF (*q*), $i = 1,2,...,M$ and $j = 1,2,...,N$, Algorithm can be summarized in Fig. 2 known as factor graph of non-binary LDPC. Coded symbol likelihoods c_j are the column weight of codes. Likelihoods of each coded symbols f_j are column vectors containing *q* likelihoods of coded symbol. Block labeled ∏ connects non-binary element in each row to parity check matrix.

3. Proposed method for the construction of H matrix with predetermined girth

To construct parity check matrix with target girth g_t and desired number of bits nodes N_t , we start with an initial column weight two **H** matrix of size $g_t/2xg_t/2$. To expand this matrix into N_t columns, we recursively find an appropriate condition that how to add bit nodes and check nodes into this matrix. Note that, adding more bit nodes may lead the current girth smaller than g_t . Hence, new check nodes should be also added to balance the target girth. We continue the process until the desire N_t is obtained. The steps are summarized in algorithm 1 and algorithm 2 as follows.

Algorithm 1:

```
Define a target girth $g_t$
Define a target number of columns $N_t$
Construct an initial column weight two $\mathbf{H}$ matrix of size $g_t/2 \times g_t/2$
$N$ = number of columns of $\mathbf{H}$
while $N < N_t$ do
    Choose the check node with the lowest row-weight and let it be $c_{\text{init}}$ in level 1.
    $(L_{\max}, c_{L_{\max}}) = \text{findLmax}(\mathbf{H}, c_{\text{init}})$ \\Algorithm 2
    if $L_{\max}$ ³ $g_t/2$ then
```

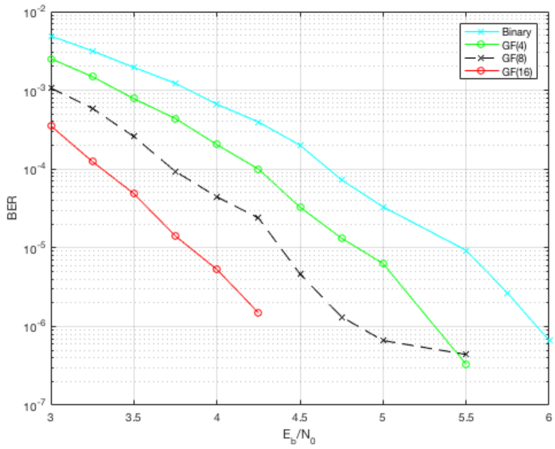

Figure 3. BER perfomance of different **H** matrics

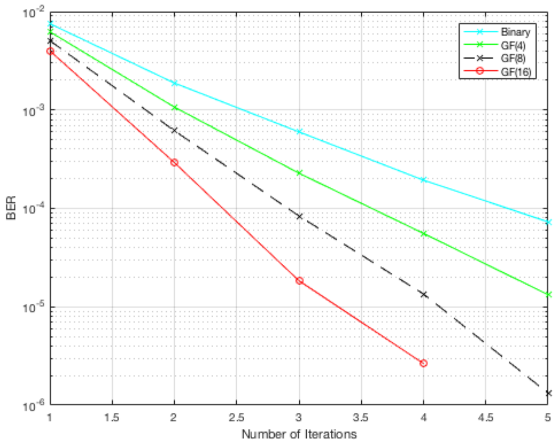

Figure 4. Average number of iterations required for decoding different LDPC codes over GF(*q*)

```
        Add one bit node into $\mathbf{H}$ to connect $c_{L_{\max}}$ and $c_{\text{init}}$.
    else
        Add $g_t/2$ - $L_{\max}$ $+1$ bit nodes into $\mathbf{H}$ and also
        add $g_t/2 - L_{\max}$ check nodes to keep target girth $g_t$.
    end if
    Update $N$ = number of columns of $\mathbf{H}$
end while
return $\mathbf{H}$
```

Algorithm 2:

```
function : $(L_{\max}, c_{L_{\max}}) = \text{findLmax}(\mathbf{H}, c_{\text{init}})$
$L = 1;$ \\ Set initial current level = 1
$\mathbf{c}_L = c_{\text{init}}$ \\ Set initial check node in current level
while forever do
    $\mathbf{c}_u = [\ ]$ \\ Initial all check nodes for current level
    $\mathbf{b}_u = [\ ]$ \\ Initial all bit nodes for current level
    for all $c_n \in \mathbf{c}_L$ do
        $\mathbf{b}_n = \text{find}(\mathbf{H}(c_n,:))$ \\ Find all bit nodes which
                                are connected with
                                the current check node
        $\mathbf{b}_u = [\mathbf{b}_u\ \mathbf{b}_n]$ \\ Update without redundancy
        $\mathbf{H}(c_n, \mathbf{b}_n) = 0$ \\ Eliminate current check
                                node and its bit nodes
    end for all
    for all $b_n \in \mathbf{b}_u$ do
        $\mathbf{c}_n = \text{find}(\mathbf{H}(:, b_n))$
        $\mathbf{c}_u = [\mathbf{c}_u\ \mathbf{c}_n]$
        $\mathbf{H}(\mathbf{c}_n, b_n) = 0$
    end for all
    if $\mathbf{c}_u$ is an empty set then \\ Check for lowest level
        return $L_{\max} = L$
        return $c_{L_{\max}} = \mathbf{c}_L(1)$
        break
    else
        $L = L + 1$ \\ Go to next level
        $\mathbf{c}_L = \mathbf{c}_u$ \\ Update $\mathbf{c}_L$
    end if
end while
```

Algorithm 2 is used to find $L_{\max}$ which is the highest level when the check node with lowest row-weight is assigned to be lowest level. With the binary parity check matrix $\mathbf{H}$ with desired N_t and target girth g_t, we apply a non-binary LDPC algorithm following Mackay's algorithm. Nonzero elements of H matrix is replaced by randomly generated GF(q) elements. In the next section, we will show by simulation that a non-binary technique outperforms its corresponding binary LDPC, especially in high-order of Galois filed.

4. Simulation results and discussions

Consider an $M \times N$, **H** matrix, where N is the length of a code word, and M is the number of parity bits. To evaluate the performance of the proposed algorithm, we simulate the system based on an additive white Gaussian noise (AWGN) channel model, where a binary input sequence $a_k \in \{0, 1\}$ of length $N - M$ bits is encoded by an LDPC encoder and is mapped to an N-bit coded sequence $b_k \in \{\pm 1\}$. Then, the received sequence is given by $y_k = b_k + n_k$, where n_k is AWGN with zero mean and variance σ^2. At the receiver, the received sequence y_k is decoded by an LDPC decoder implemented based on a message passing algorithm [1]. Examined **H** matrix of size (324,648). Each BER point is computed based on a minimum number of 5000 data packets and all LDPC codes use the **H** matrix of size 324×648. We use the LDPC decoder with 5 iterations and plot the BER performance as shown in Figure 2. In simulation, the signal-to-noise ratio is defined as

$$\mathrm{SNR} = 10\log_{10}\left(\frac{1}{\sigma^2}\right), \qquad (2)$$

in decibel (dB).

We also compare the performance of different schemes by plotting the average number of iterations needed to decode all codeword of finite GF (q) LDPC codes as a function of E_b/N_0 as shown in Fig. 4 based on our example. It is obvious that the LDPC codes iteration can help to increase the performance of the system.

As expected, a non-binary LDPC code with large q performs better than that with small q.

5. Conclusions

In this paper, we explained importance of non-binary LDPC codes and its discerption from associated open literature. In addition, we describes decoding procedure of non-binary LDPC codes in Section 2. We have presented construction steps for generating **H** matrix with column weight two by using fixed target girth g_t and code length N. Section 3 showed performance comparisons of our tested simulation for constructed **H** matrix. We take account of higher orders of GF(q), as well. As depicted higher orders of GF(q) outperforms binary LDPC codes and having substantially improved performance. In addition, we also compares BER as a function of number of iterations for a fixed SNR=4.5 dB.

References

[1] R. Gallager, “Low-density parity-check codes,” *IRE Trans. on Inform. Theory,* vol. 8, no. 1, pp. 21–28, January 1962.

[2] D. J. C. MacKay and R. M. Neal, “Near Shannon limit performance of low density parity check codes,” *Electron. Lett.*, vol. 32, pp. 1645–1646, August. 1996.

[3] D. Declercq and M. Fossorier, “Decoding algorithm for non-binary LDPC codes over GF (2^m) “. *IEEE Transactions on Communication vol.* 55, no. 4 pp. 633-643, April 2007.

[4] Davey, Matthew C., and David JC MacKay. "Low density parity check codes over GF (q)."*Information Theory Workshop, 1998*. IEEE, 1998.

[5] D. J. C. MacKay and M. Davey, “Evaluation of Gallager codes for short block length and high rate applications”, *Proc. IMA Workshop Codes, Syst., Graphical Models,* 1999.

[6] L. Barnault and D. Declercq, “Fast decoding algorithm for LDPC over GF”, *Proc. Inf. Theory Workshop Paris, France,* Mar. 2003, p. 70.

[7] M. Fossorier, “Quasi-cyclic low-density parity-check codes from circulant permutation matrices,” *IEEE Trans. Inf. Theory,* vol. 50, no. 8, pp. 1788–1793, 2004.

[8] Tanner, Robert Michael. "A recursive approach to low complexity codes. "*Information Theory, IEEE Transactions on* 27.5 (1981): 533-547.

[9] Y. Kou, S. Lin, and M. P. C. Fossorier,”Low density parity-check codes based on finite geometries: A rediscovery and new results,” *IEEE Trans. Inform. Theory,* vol. 47, no. 7, pp. 2711–2736, Nov. 2001.

[10] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2004.

[11] S. Lin, S. Song, L. Lan, L. Zeng, and Y.-Y. Tai, “Constructions of non binary quasi-cyclic LDPC codes: a finite field approach,” in *Proc. Inform. Theory Applications Workshop,* 2006.