Advances in Electrical and Computer Engineering **Volume 11, Number 4, 2011**

Low Complexity Encoder of High Rate Irregular QC-LDPC Codes for Partial Response Channels

Mongkol KUPIMAI, Anupap MEESOMBOON, Virasit IMTAWIL
Department of Electrical Engineering, Faculty of Engineering
Khon Kaen University,40002, Thailand
virasit@kku.ac.th

***Abstract*—High rate irregular QC-LDPC codes based on circulant permutation matrices, for efficient encoder implementation, are proposed in this article. The structure of the code is an approximate lower triangular matrix. In addition, we present two novel efficient encoding techniques for generating redundant bits. The complexity of the encoder implementation depends on the number of parity bits of the code for the one-stage encoding and the length of the code for the two-stage encoding. The advantage of both encoding techniques is that few XOR-gates are used in the encoder implementation. Simulation results on partial response channels also show that the BER performance of the proposed code has gain over other QC-LDPC codes.**

***Index Terms*—circulant permutation matrices, high rate irregular QC-LDPC codes, low encoding complexity, partial response channels, redundant parity bits.**

I. INTRODUCTION

Low-density parity-check (LDPC) codes are a class of systematic linear block codes with a sparse matrix and also a subclass of prominent iteratively decodable codes. LDPC codes are forward error correction (FEC) codes as first proposed by Gallager [1] in the early 1960s and rediscovered by Mackay and Neal [2]. The name of this code comes from the characteristic of their parity-check matrix which contains only a few non-zero elements in comparison to the number of zero elements. A lot of researchers have been interested in LDPC codes because they provide good performance, which is very close to the theoretical limit. LDPC codes also offer lower complexity decoding than the well know capacity approaching code called the Turbo code [3]. References [4]-[7] show that long length irregular LDPC codes obtain a performance within a fraction of a decibel (dB) of the theoretical limit. Due to the excellent performance, these codes have been adopted in a wide range of applications such as IEEE 802.11n [8] and IEEE 802.16e [9]. Recently, LDPC codes have attracted a lot of interest in magnetic recording channels [10]-[12]. However, a main drawback of general LDPC codes is a high encoding complexity. The random structure LDPC code has high complexity encoding quadratic with the code length. The structure of the LDPC codes can be used to significantly alleviate the encoding problem and simplify hardware implementation. The Quasi-Cyclic LDPC (QC-LDPC) code is a subclass of LDPC codes that can perform as well as randomly constructed LDPC codes with iterative decoding on belief propagation in terms of bit error rate (BER). The advantage of the generator matrix of the QC-LDPC code is that it can be encoded by using a simple shift register. References [15]-[16] show that the complexity of encoding of QC-LDPC codes is linearly proportional to the number of parity bits of the code for serial encoding and the length of the code for high-speed parallel encoding.

The QC-LDPC codes based on circulant matrices [13]-[14], which are kinds of structured QC-LDPC codes, are easy to implement because of their block and cyclic interconnection. In this paper, we design a new structure of $(3, L)$ regular QC-LDPC codes based on circulant permutation matrices that are free from girth 4, girth 6, and sometimes girth 8. Moreover, we modify this code by replacing some circulant permutation matrices with zero matrices. The new structure code is an approximate lower-triangular matrix so that the generator of this code is a linear independent form. In addition, we present two kinds of efficient encoding techniques that can generate parity bits of the codeword. It shows that the encoding complexity of the proposed code depends on the number of parity bits of the code for a one-stage encoding scheme, and the length of the code for a two-stage encoding scheme. The advantage of both encoding techniques is that it can generate redundant bits of the codeword with lower complexity, as compared with previously found ones. Finally, we show the BER of the proposed code at high code rate for magnetic recording channels.

The rest of this paper is organized as follows. The basic concept of (J, L) regular QC-LDPC codes is reviewed in section II. The proposed construction of $(3, L)$ regular QC-LDPC codes based on circulant permutation matrices is introduced in section III. The modification of the proposed code for efficient encoding is presented in section IV. The magnetic recording channels model used in simulations is described in section V. Finally, section VI and section VII contain summaries of the main results and conclusions.

II. (J, L) REGULAR QC-LDPC CODE BACKGROUND

The (J, L) regular QC-LDPC codes are defined by a parity-check matrix $\mathbf{H}$, in which each column has Hamming weight J and each row weight L [1] and [13]-[14]. The (J, L) regular QC-LDPC code introduced in [13] is

This work was supported in part by the Industry/University Cooperative Research Center (I/U CRC) in HDD component, the Faculty of Engineering, Khon Kaen University, and National Electronics and computer Technology Center (NECTEC), National Sciences and Technology Development Agency (NSTDA), Grant No. CPN-HR 03-02-50 D and the Faculty of Engineering, Khon Kaen University, Thailand, under the new researcher grant project.

Digital Object Identifier 10.4316/AECE.2011.04008

based on circulant matrices of size $p \times p$ and can be represented as

$$\mathbf{H} = \begin{bmatrix} I(0) & I(0) & \cdots & I(0) \\ I(0) & I(p_{1,1}) & \cdots & I(p_{1,L-1}) \\ \vdots & \vdots & \ddots & \vdots \\ I(0) & I(p_{J-1,1}) & \cdots & I(p_{J-1,L-1}) \end{bmatrix}_{(J\times p)\times(L\times p)} \quad (1)$$

where $I(p_{j,l})$ represents the circulant permutation matrix which shifts the identity matrix of size $p \times p$ to the right by $p_{j,l}$ times. The value of $p_{j,l}$ depends on some conditions that were described in [13], where $1 \le j \le J-1$, $1 \le l \le L-1$. $I(0)$ is the identity matrix. The code rate of (J, L) regular QCLDPC codes is $R \ge 1 - K \ / \ N$, where K is the message length and $N = L \times p$ is the code length. The parity-check matrix of (J, L) regular QC-LDPC codes with column weight $J \ge 2$ has rank of at most $pJ - J + 1$. The upper bound of the minimum distance is $d_{\min} \le (J + 1)!$ if $J \ge 3$ [13]-[14] and [17].

III. THE PROPOSED REGULAR QC-LDPC CODE

In this section, we show the method to construct the parity-check matrix of $(3, L)$ regular QC-LDPC codes based on circulant matrices. Our parity-check matrix can be represented in transpose matrix $\mathbf{H^T}$ form as follows.

$$\mathbf{H^T} = \begin{bmatrix} I(X_1) & I(Y_1) & I(0_1) \\ I(X_2) & I(Y_2) & I(0_2) \\ I(X_3) & I(Y_3) & I(0_3) \\ \vdots & \vdots & \vdots \\ I(X_L) & I(Y_L) & I(0_L) \end{bmatrix}_{(L\times p)\times(3\times p)} \quad (2)$$

where X_l and Y_l are integers that are selected from set $S = \{2, 3, 4, \ldots p\}$, where $X_1 < X_2 < \cdots < X_{l-1} < X_l$, $Y_1 > Y_2 > \cdots > Y_{l-1} > Y_l$, and $1 \le l \le L$. All $I(0_l)$ are $p \times p$ identity matrices. $I(X_l)$ and $I(Y_l)$ are $p \times p$ circulant permutation matrices which shift the columns of the identity matrix to the right by $X_l - 1$ and $Y_l - 1$ positions, respectively. The relation between $I(X_l)$ and $I(Y_l)$ can be defined as

$$X_l + Y_l = p + 2 \quad (3)$$

The position of the $1'$s of any circulant matrices can be located according to (4), (5), and (6) for the $I(X_l)$, (7), (8), and (9) for the $I(Y_l)$, and (10) for the $I(0_l)$, respectively. It is important to note that we set the first row and the first column of any matrices with the index 1.

$$I(X_l)_{s,t} = \begin{cases} 1 & \text{if } t = (X_l + s - 1) \bmod p \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

$$s = (Y_l + t - 1) \bmod p \quad (5)$$

$$t = (X_l + s - 1) \bmod p \quad (6)$$

$$I(Y_l)_{m,n} = \begin{cases} 1 & \text{if } n = (Y_l + m - 1) \bmod p \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

$$m = (X_l + n - 1) \bmod p \quad (8)$$

$$n = (Y_l + m - 1) \bmod p \quad (9)$$

$$I(0_l)_{m,n} = \begin{cases} 1 & \text{if } j = k \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

where $1 \le j, k, m, n, s,$ and $t \le p$. The order pairs (s, t), (m, n), and (j, k) are indices of $I(X_l)$, $I(Y_l)$, and $I(0_l)$, respectively. Note that if s, t, m, and n are zero they will be replaced with p.

We group $I(X_l)$, $I(Y_l)$ and $I(0_l)$ to circulant permutation submatrix rows of $\mathbf{H^T}$ shown as follows

$$\mathbf{H^T} = \begin{bmatrix} \mathrm{H}_1 \\ \mathrm{H}_2 \\ \mathrm{H}_3 \\ \vdots \\ \mathrm{H}_L \end{bmatrix}_{(L\times 1)} \quad (11)$$

$$\mathbf{H}_l = \begin{bmatrix} I(X_l) & I(Y_l) & \mathrm{I}(0_l) \end{bmatrix}_{(1\times 3)} \quad (12)$$

where $1 \le l \le L$.

A. The girth of the proposed QC-LDPC

Now we consider a cycle or the closed path in the Tanner graph [18]. The Tanner graph can be created with an edge between bit and check node if there are corresponding non-zero elements in the matrix $\mathbf{H}$. A cycle in the Tanner graph refers to a closed path that starts and ends at the same node. A girth is the smallest cycle in the graph. In this paper, the index of $1'$ s of any cycles of length $2i$ in the Tanner graph of the code can be represented as the ordered series $(x_1, y_1), (x_2, y_1), (x_2, y_2), \cdots, (x_i, y_i), (x_1, y_1)$, where $1 \le k \le i$, $x_k \ne x_{k-1}$, $y_k \ne y_{k-1}$, $x_i \ne x_1$, $y_i \ne y_1$. The i is a positive integer.

Lemma 1: If the $\mathbf{H^T}$ is defined as (11), the parity-check matrix is without cycles of four.

Proof 1: For any positions of $1'$ *s* between two circulant permutation submatrix rows of $\mathbf{H^T}$ e.g. H_a and H_b, where $1 \le a < b \le L$, $X_a < X_b$ and $Y_a < Y_b$, if the matrix $\mathbf{H^T}$ has a path of length 4, it will be created by three patterns as follows.

Firstly, the path that combines between two $I(X)$ and two $I(Y)$ is shown as

$$I(X_a)_{(s_1,t_1)} \rightarrow I(X_b)_{(s_2,t_2)} \rightarrow I(Y_b)_{(m_1,n_1)} \rightarrow I(Y_a)_{(m_2,n_2)} \rightarrow I(X_a)_{(s_1,t_1)}$$

The closed path starts and ends at $I(X_a)_{(s_1,t_1)}$. The path of length 4 exists if $s_1 = m_2$, $n_1 = n_2$, $t_1 = t_2$, $m_1 = s_2$ and $s_1 \ne s_2$. However, s_1 is not equal to m_2 whereas $n_1 = n_2$. Let us prove this by writing

$$s_1 = (Y_a + t_1 - 1) \bmod p \quad (13)$$

$$s_2 = (Y_b + t_1 - 1) \bmod p \quad (14)$$

$$n_1 = (Y_b + s_2 - 1) \bmod p \quad (15)$$

$$m_2 = (Y_a + n_2 - 1) \bmod p \quad (16)$$

By substituting (14) into (15) and then (15) into (16)

$$m_2 = (Y_a + 2Y_b + t_1 - 3) \, mod \; p \tag{17}$$

It can be seen that (13) is not equal to (17). Clearly, a path of length 4 does not exist.

Secondly, consider the path that combines two $I(X)$ and two $I(0)$.

$$I(X_a)_{(s_1,t_1)} \rightarrow I(X_b)_{(s_2,t_2)} \rightarrow I(0_b)_{(j_1,k_1)} \rightarrow I(0_a)_{(j_2,k_2)} \rightarrow I(X_a)_{(s_1,t_1)}$$

The closed path starts and ends at $I(X_a)_{(s_1,t_1)}$. The path of length 4 exists if $j_1 = j_2 = k_1 = k_2 = t_1 = t_2 = s_1 = s_2$. However, s_1 is not equal to s_2 whereas $t_1 = t_2$. Clearly, a path of length 4 does not exist.

Finally, consider the path that combines two $I(Y)$ and two $I(0)$

$$I(Y_a)_{(m_1,n_1)} \rightarrow I(Y_b)_{(m_2,n_2)} \rightarrow I(0_b)_{(j_1,k_1)} \rightarrow I(0_a)_{(j_2,k_2)} \rightarrow I(Y_a)_{(m_1,n_1)}$$

The closed path starts and ends at $I(Y_a)_{(m_1,n_1)}$. The path of length 4 exists if $j_1 = j_2 = k_1 = k_2 = n_1 = n_2 = m_1 = m_2$. However, m_1 is not equal to m_2 whereas $n_1 = n_2$. Clearly, a path of length 4 does not exist.

Lemma 2: If the $\mathbf{H}^{\mathbf{T}}$ is defined as (11), the parity-check matrix exists with length of 6 if and only if the condition given below is satisfied.

$$X_c = 2X_b - X_a \tag{18}$$

In this paper, there are two different closed paths of length 6. The closed paths are created by combination of three circulant permutation submatrices rows of $\mathbf{H}^{\mathbf{T}}$ e.g. H_a, H_b and H_c, where $1 \le a < b < c \le L$, $X_a < X_b < X_c$ and $Y_a > Y_b > Y_c$.

The first path can be shown as

$$I(X_a)_{(s_1,t_1)} \rightarrow I(X_b)_{(s_2,t_2)} \rightarrow I(Y_b)_{(m_1,n_1)} \rightarrow I(Y_c)_{(m_2,n_2)} \rightarrow I(0_c)_{(j_1,k_1)} \rightarrow I(0_a)_{(j_2,k_2)} \rightarrow I(X_a)_{(s_1,t_1)}$$

where $m_1 = s_2, t_1 = t_2, n_1 = n_2$, and $m_2 = s_1 = j_1 = k_1 = j_2 = k_2$

The second path can be shown as

$$I(Y_a)_{(m_1,n_1)} \rightarrow I(Y_b)_{(m_2,n_2)} \rightarrow I(X_b)_{(s_1,t_1)} \rightarrow I(X_c)_{(s_2,t_2)} \rightarrow I(0_c)_{(j_1,k_1)} \rightarrow I(0_a)_{(j_2,k_2)} \rightarrow I(Y_a)_{(m_1,n_1)}$$

where $m_2 = s_1,\ t_1 = t_2,\ n_1 = n_2$, and $m_1 = s_2 = j_1 = k_1 = j_2 = k_2$

Proof 2.1: If the first closed path starts and ends at $I(X_a)_{(s_1,t_1)}$, let us prove that (18) can create a girth 6 as follows

$$s_1 = (p + 2 - X_a + t_1 - 1) \, mod \; p \tag{19}$$

$$s_2 = (p + 2 - X_b + t_1 - 1) \, mod \; p \tag{20}$$

$$n_1 = (p + 2 - X_b + s_2 - 1) \, mod \; p \tag{21}$$

$$m_2 = (X_c + n_1 - 1) \, mod \; p \tag{22}$$

By substituting (20) into (21) and then (21) into (22), now we have

$$m_2 = (X_c + 2p - 2X_b + t_1 + 1) \, mod \; p \tag{23}$$

If there is a closed path, (19) is equal to (23).

$$(X_c + 2p - 2X_b + t_1 + 1) \, mod \; p = (p + 2 - X_a + t_1 - 1) \, mod \; p$$

or equivalently,

$$X_c - 2X_b + t_1 + 1 = -X_a + t_1 + 1 \tag{24}$$

Finally, we have

$$X_c = 2X_b - X_a \tag{25}$$

Example 1: If $p = 11$, $X_a = 3$, $X_b = 5$, and $X_c = 7$, so that $Y_a = 10$, $Y_b = 8$, and $Y_c = 6$. The first closed path of cycle 6 can be written as

$$I(3)_{(1,3)} \rightarrow I(5)_{(10,3)} \rightarrow I(8)_{(10,6)} \rightarrow I(6)_{(1,6)} \rightarrow I(0_c)_{(1,1)} \rightarrow I(0_a)_{(1,1)} \rightarrow I(3)_{(1,3)}$$

Proof 2.2: If the second closed path starts and ends at $I(Y_a)_{(m_1,n_1)}$. Let us prove that (18) can create a girth 6 as follows

$$m_1 = (X_a + n_1 - 1) \, mod \; p \tag{26}$$

$$m_2 = (X_b + n_1 - 1) \, mod \; p \tag{27}$$

$$t_1 = (X_b + m_2 - 1) \, mod \; p \tag{28}$$

$$s_2 = (p + 2 - X_b + t_1 - 1) \, mod \; p \tag{29}$$

By substituting (27) into (28) and then (28) into (29), now we have

$$s_2 = (p - X_c + 2X_b + n_1 - 1) \, mod \; p \tag{30}$$

If there is a closed path, (26) is equal to (30).

$$(p - X_c + 2X_b + n_1 - 1) \, mod \; p = (X_a + n_1 - 1) \, mod \; p$$

or equivalently,

$$X_c - 2X_b + t_1 + 1 = -X_a + t_1 + 1 \tag{31}$$

Finally, we have

$$X_c = 2X_b - X_a \tag{32}$$

Example 2: If $p = 11$, $X_a = 3$, $X_b = 5$, and $X_c = 7$, so that $Y_a = 10$, $Y_b = 8$, and $Y_c = 6$. The first closed path of cycle 6 can be written as

$$I(10)_{(1,10)} \rightarrow I(8)_{(3,10)} \rightarrow I(5)_{(3,7)} \rightarrow I(7)_{(1,7)} \rightarrow I(0_c)_{(1,1)} \rightarrow I(0_a)_{(1,1)} \rightarrow I(10)_{(1,10)}$$

The closed paths of length 8 are created by combination of three circulant permutation submatrix rows of $\mathbf{H}^{\mathbf{T}}$ e.g. H_a, H_b and H_c, where $1 \le a < b < c \le L$, $X_a < X_b < X_c$ and $Y_a > Y_b > Y_c$.

The first path can be shown as

$$I(X_a)_{(s_1,t_1)} \rightarrow I(X_b)_{(s_2,t_2)} \rightarrow I(Y_b)_{(m_1,n_1)} \rightarrow I(Y_c)_{(m_2,n_2)} \rightarrow I(0_c)_{(j_1,k_1)} \rightarrow I(0_b)_{(j_2,k_2)} \rightarrow I(Y_b)_{(m_3,n_3)} \rightarrow I(Y_a)_{(m_4,n_4)} \rightarrow I(X_a)_{(s_1,t_1)}$$

where $m_1 = s_2, t_1 = t_2, n_1 = n_2$, $n_3 = n_4, m_2 = m_3 = j_1 = k_1 = j_2 = k_2$, and $m_4 = s_1$.

The second path can be shown as

$$I(Y_a)_{(m_1,n_1)} \rightarrow I(Y_b)_{(m_2,n_2)} \rightarrow I(0_b)_{(j_1,k_1)} \rightarrow I(0_c)_{(j_2,k_2)}$$
$$\rightarrow I(Y_c)_{(m_3,n_3)} \rightarrow I(Y_b)_{(m_4,n_4)} \rightarrow I(0_b)_{(j_3,k_3)} \rightarrow$$
$$I(0_a)_{(j_4,k_4)} \rightarrow I(Y_a)_{(m_1,n_1)}$$

where $n_1 = n_2, n_3 = n_4, n_3 = n_4, m_2 = m_3 = j_1 = k_1 = j_2 = k_2$, and $m_1 = m_4 = j_3 = j_3 = k_3 = k_4$.

Lemma 3: The proposed code with girth 8 can be constructed by two conditions. The first condition is the same as the condition of girth 6 as in (18). The second condition is

$$X_c = 3X_b - 2X_a \tag{33}$$

Proof 3.1: If the first closed path starts and ends at $I(X_a)_{(s_1,t_1)}$, let us prove that (33) can create a girth 8 as follows

$$s_1 = (p + 2 - X_a + t_1 - 1) mod\ p \tag{34}$$
$$s_2 = (p + 2 - X_b + t_1 - 1) mod\ p \tag{35}$$
$$n_1 = (p + 2 - X_b + s_2 - 1) mod\ p \tag{36}$$
$$m_2 = (X_c + n_1 - 1) mod\ p \tag{37}$$
$$n_4 = (p + 2 - X_b + m_2 - 1) mod\ p \tag{38}$$
$$m_4 = (X_a + n_4 - 1) mod\ p \tag{39}$$

By substituting (35) into (36), then substituting (36) into (37), then substituting (37) into (38) and then substituting (38) into (39), now we have

$$m_4 = (X_c + 3p - 3X_b + t_1 + 1) mod\ p \tag{40}$$

If there is a closed path, (34) is equal to (40).

$$(X_c + 3p - 3X_b + t_1 + 1) mod\ p = (p - X_a + t_1 + 1) mod\ p$$

or equivalently,

$$X_c - 3X_b + X_a + t + 1 = -X_a + t_1 + 1 \tag{41}$$

Finally, we have

$$X_c = 3X_b - 2X_a \tag{42}$$

Example 3: If $p = 11$, $X_a = 2$, $X_b = 3$, and $X_c = 5$, so that $Y_a = 11$, $Y_b = 10$, and $Y_c = 8$. The first closed path of cycle 8 can be written as

$$I(2)_{(1,2)} \rightarrow I(3)_{(11,2)} \rightarrow I(10)_{(11,9)} \rightarrow I(8)_{(2,9)} \rightarrow I(0_c)_{(2,2)}$$
$$\rightarrow I(0_b)_{(2,2)} \rightarrow I(10)_{(2,11)} \rightarrow I(11)_{(1,11)} \rightarrow I(2)_{(1,2)}$$

Proof 3.2: If the second closed path starts and ends at $I(X_a)_{(s_1,t_1)}$, let us prove that (18) can create girth 8 as follows

$$m_1 = (X_a + n_1 - 1) mod\ p \tag{43}$$
$$m_2 = (X_b + n_1 - 1) mod\ p \tag{44}$$
$$n_3 = (p + 2 - X_c + m_2 - 1) mod\ p \tag{45}$$
$$m_4 = (X_b + n_3 - 1) mod\ p \tag{46}$$

By substituting (44) into (45) and (45) into (46), now we have

$$m_4 = (p + 2X_b - X_c + n_1 - 1) mod\ p \tag{47}$$

If there is a closed path, (43) is equal to (47).

$$(p + 2X_b - X_c + n_1 - 1) mod\ p = (X_a + n_1 - 1) mod\ p$$

or equivalently,

$$2X_b - X_c + n_1 - 1 = X_a + n_1 - 1 \tag{48}$$

Finally, we have

$$X_c = 2X_b - X_a \tag{49}$$

Example 4: If $p = 11$, $X_a = 2$, $X_b = 3$, and $X_c = 5$, so that $Y_a = 11$, $Y_b = 10$, and $Y_c = 8$. The first closed path of cycle 8 can be written as

$$I(10)_{(1,10)} \rightarrow I(8)_{(3,10)} \rightarrow I(0_b)_{(3,3)} \rightarrow I(0_c)_{(3,3)} \rightarrow I(6)_{(3,8)}$$
$$\rightarrow I(8)_{(1,8)} \rightarrow I(0_b)_{(1,1)} \rightarrow I(0_a)_{(1,1)} \rightarrow I(10)_{(1,10)}$$

B. Algorithms for constructing the proposed code

In this section we introduce two algorithms for constructing the proposed code. *Algorithm I* is used to construct the matrix **H** that is free from girth 4 and girth 6 whereas the *algorithm II* is free from girth 4, girth 6, and girth 8.

Define $S = \{2, 3, 4, \cdots, p\}$, where p is the size of the circulant matrix that depends on the rate of the codes and the length of the codeword. Define set X and set Y as the set of integers. Initially,

$X = \{2, 3\}$, $Y = \{p + 2 - 2, p + 2 - 3\}$.

Define R as a set of integers that is computed from step 2 of the algorithm, with initially $R = \varnothing$.

Algorithm I

Step 1. Set $X = \{2, 3\}$

Step 2. Compute the process to find the integer which satisfies the condition of (18) by the following script.

```
w = 1
v = 2
for $j$ = 1 to $n_v - 1$ do ; Note $n_v$ is the cardinality of set $X$
    for $i$ = v to $n_v$ do
        $j_f = ((p + 2 - Y(i)) + Y(j) - 1) mod\ p$ ; Note if $j_f = 0$ replace with $j_f = p$
        $k_f = ((p + 2 - Y(i)) + j_{(f-1)}) mod\ p$ ; Note if $k_f = 0$ replace with $k_f = p$
        $R$(w) ← $k_f$ ; Note keep $k_f$ in set $R$.
        w ← w + 1 ; Note increase w by 1
    end
    v ← v + 1 ; Note increase v by 1
end
```

Step 3. Decision, if $X \cap R = \varnothing$, store set X and then select the next member from set S to be the next member of set X and go to step 2. If $X \cap R \neq \varnothing$, select the next member from set $S - (X \cap R)$ instead of the last number in set X then go to step 2.

Step 4. The process ends when the members of set X equal to L. The members of X are used to construct the proposed matrix **H** that is free from girth 6.

Next, we show the algorithm for the proposed code that is free from girth 8.

Initially, $X = \{2, 3, 5\}$, $Y = \{p + 2 - 2, p + 2 - 3, p + 2 - 5\}$.

Define $R1$ and $R2$ as the set of integers that are computed from step 2 of the algorithm, with initially $R1 = \varnothing$ and $R2 = \varnothing$.

Algorithm II

Step 1. Set $X = \{2, 3, 5\}$.

Step 2. Compute the process to find the integer which satisfies condition of (33) by the following script.

w = 1.

```
v = 2.
for $j$ = 1 to $n_v - 1$ do ; Note $n_v$ is the cardinality of set $X$
    for $i$ = v to $n_v$ do
        $j_f = ((p + 2 - X(i)) + X(j) - 1) mod\ p$ ; Note if $j_f = 0$ replace with $j_f = p$
        $k_f = ((p + 2 - X(i)) + j_{(f-1)}) mod\ p$ ; Note if $k_f = 0$ replace with $k_f = p$
        $j_{f1} = (X(i+1) + j_{(f-1)}) mod\ p$ ; Note if $j_{f1} = 0$ replace with $j_{f1} = p$
        $R1(w) \leftarrow j_{f1}$ ; Note keep $j_{f1}$ into set $R1$.
        $j_{f2} = ((p + 2 - Y(i)) + Y(j) - 1) mod\ p$ ; Note if $j_{f2} = 0$ replace with $j_{f2} = p$
        $R2(w) \leftarrow j_{f2}$ ; Note keep $j_{f2}$ into set $R2$.
        w ← w + 1 ; Note increase w by 1
    end
    v ← v + 1 ; Note increase v by 1
end
```

Step 3. Decision, if $R1 \cap R2 = \varnothing$, store the set X and select the next one member from set $S - X$ into set X and then go to step 2. If $R1 \cap R2 \neq \varnothing$, select the next member from set $S - X$ instead of the last number in set X and then go to step 2.

Step 4. The process ends when the members of set X are equal to L. The members of X are used to construct the proposed matrix **H** that is free from girth 8.

IV. THE PROPOSED IRREGULAR QC-LDPC CODE

The proposed parity-check matrix as in (2) can be modified to be an irregular QC-LDPC by replacing $I(X_1)$ with identity matrices and replacing $I(Y_1)$, $I(0_1)$, and $I(0_2)$ with **0**, where **0** is a $p \times p$ zero matrix. This code is an approximate lower-triangular matrix so that the generator of this code is a linear independent form. The transpose of the parity-check matrix $\hat{\mathbf{H}}^{\mathbf{T}}$ is shown in (50).

Next we show a novel method to generate redundant bits by using a simple shift register as follows:

Denote $w = (Z^1, Z^2\ Z^3, D^1, D^2, D^3, \cdots, D^{(L-3)})$ as a codeword vector. w is generated from systematic LDPC codes so that $w\hat{\mathbf{H}}^{\mathbf{T}} = 0$.

Denote $Z^u = (Z_1^u, Z_2^u, Z_3^u, \ldots, Z_p^u)$ as a redundant parity bits vector, where $1 \le u \le 3$.

Denote $D^v = (D_1^v, D_2^v, D_3^v, \ldots, D_p^v)$ as an information vector, where $1 \le v \le L\text{-}3$.

Now divide $\hat{\mathbf{H}}^{\mathbf{T}}$ into $\mathbf{B}^{\mathbf{T}}$ and $\mathbf{G}^{\mathbf{T}}$ as follows

$$\hat{\mathbf{H}}^{\mathbf{T}} = \begin{bmatrix} I(0) & 0 & 0 \\ I(X_2) & I(Y_2) & 0 \\ I(X_3) & I(Y_3) & I(0_3) \\ \vdots & \vdots & \vdots \\ I(X_L) & I(Y_L) & I(0_L) \end{bmatrix}_{(L\times P)\times(3\times P)} \tag{50}$$

$$\mathbf{B}^{\mathbf{T}} = \begin{bmatrix} I(0) & 0 & 0 \\ I(X_2) & I(Y_2) & 0 \\ I(X_3) & I(Y_3) & I(0_3) \end{bmatrix}_{(3\times P)\times(3\times P)} \tag{51}$$

$$\mathbf{G}^{\mathbf{T}} = \begin{bmatrix} I(X_2) & I(Y_2) & I(0_2) \\ I(X_3) & I(Y_3) & I(0_3) \\ I(X_4) & I(Y_4) & I(0_4) \\ \vdots & \vdots & \vdots \\ I(X_L) & I(Y_L) & I(0_L) \end{bmatrix}_{((L-3)\times P)\times(3\times P)} \tag{52}$$

If $w\hat{\mathbf{H}}^{\mathbf{T}} = 0$, so that $Z\mathbf{B}^{\mathbf{T}} + D\mathbf{G}^{\mathbf{T}} = 0$. Denote $D\mathbf{G}^{\mathbf{T}} = Q^w$, where $Q^w = (Q_1^w, Q_2^w, Q_3^w, \cdots, Q_p^w)$, $1 \le w \le 3$. We can generate all redundant parity bits by using the sequentially relation of equations as follows:

$$Z^3 I(0) = Q^3 \tag{53}$$

$$Z^2 I(Y_2) = Z^3 I(Y_3) + Q^2 \tag{54}$$

$$Z^1 I(0) = Z^2 I(X_2) + Z^3 I(X_3) + Q^1 \tag{55}$$

The results of $Z^u I(X_l)$ and $Z^u I(Y_l)$ are equivalent to the vector Z^u that has data shifted to the right by $(X_l) - 1$ positions and $(Y_l) - 1$ positions, respectively.

A. Two-stage encoding schemes

The two-stage encoding scheme is as shown in figure 1. In the first stage of the encoding process, all information is read into p buffers of the $L - 3$ feedback shift registers or equivalently, vector D. In this process, each subvector of vector Q contains the results that are generated by multiplying information vectors with circulant permutation submatrix columns of $\mathbf{G}^{\mathbf{T}}$ as shown in (56), (57), and (58).

$$Q^1 = (D^1, D^2, \ldots, D^{L-3}) \begin{bmatrix} I(X_4) \\ I(X_5) \\ \vdots \\ I(X_L) \end{bmatrix} \tag{56}$$

$$Q^2 = (D^1, D^2, \ldots, D^{L-3}) \begin{bmatrix} I(Y_4) \\ I(Y_5) \\ \vdots \\ I(Y_L) \end{bmatrix} \tag{57}$$

$$Q^3 = (D^1, D^2, \ldots, D^{L-3}) \begin{bmatrix} I(0_4) \\ I(0_5) \\ \vdots \\ I(0_L) \end{bmatrix} \tag{58}$$

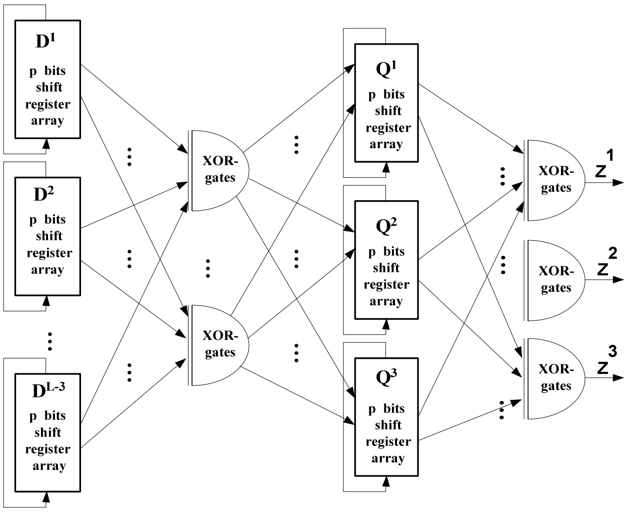
D1
p bits shift register array
D2
p bits shift register array
DL-3
p bits shift register array
XOR-gates
XOR-gates
Q1
p bits shift register array
Q2
p bits shift register array
Q3
p bits shift register array
XOR-gates
XOR-gates
XOR-gates
Z1
Z2
Z3

Figure 1. Block diagram of the two-stage encoding circuit

In the encoder implementation, we use several buffers of information registers that correspond to bits 1 of the first column of each circulant permutation submatrix columns of matrix $\mathbf{G}^\mathbf{T}$ as inputs of XOR-gates. The summation results from all XOR-gates are set to the first buffer of all Q registers. We can form new equations by using (5) and (8) as follows.

$$Q_1^1 = D_{Y_4}^1 + D_{Y_5}^2 + D_{Y_6}^3 + \ldots, D_{Y_l}^{L-3} \tag{59}$$

$$Q_1^2 = D_{X_4}^1 + D_{X_5}^2 + D_{X_6}^3 + \ldots, D_{X_l}^{L-3} \tag{60}$$

$$Q_1^3 = D_1^1 + D_1^2 + D_1^3 + \ldots, D_1^{L-3} \tag{61}$$

At the beginning of the first stage encoding, all $L-3$ information sections are read into $L-3$ feedback shift registers $(D^1, D^2, \cdots, D^{(L-3)})$ in the first clock cycle. The results are formed based on (59), (60), and (61) and then are shifted into Q^1, Q^2, and Q^3 registers concurrently. By continuing to shift registers to the left $p-1$ times until the end at the $p-th$ clock cycles, all of the results are stored in Q^1, Q^2, and Q^3 registers. This stage requires a total of $(L-3)p$ flip-flops and $((L\ \text{-}\ 3)\ \text{-}\ 1)3$ XOR-gates.

Now we replace Z^3 with Q^3 into (54) so that $Z^2(Y_2) = Q^3(Y_3) + Q^2$. After that we set indices of Q^3 and Q^2 that correspond to the first index of Z^2 by using (8), and we have

$$Z_1^2 = Q^3_{((X_3+Y_2\text{-}1)\,mod\ p)} + Q^2_{Y_2} \tag{62}$$

Next, we replace Z^3 with Q^3 and Z^2 with $Q^3_{((X_2+Y_2\text{-}1)\,mod\ p)} + Q^2_{Y_2}$ into (55), then using (5) we get the indices of Q^3 and Q^2 that correspond to the first index of Z^1 as follows:

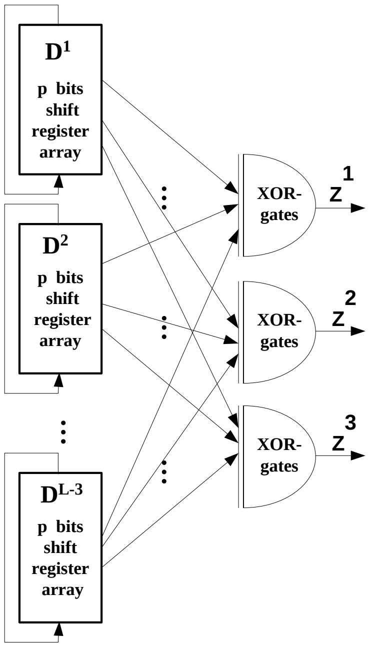
D1
p bits shift register array
D2
p bits shift register array
DL-3
p bits shift register array
XOR-gates
XOR-gates
XOR-gates
Z1
Z2
Z3

Figure 2. Block diagram of the one-stage encoding circuit.

$$Z_1^1 = Z_{Y_2}^2 + Z_{Y_3}^2 + Q_1^1 \tag{63}$$

$$Z_1^1 = Q^3_{((X_3+Y_2+Y_2\text{-}2)\,mod\ p)} + Q^3_{((X_2+Y_2\text{-}1)\,mod\ p)} + Q^3_{Y_3} + Q_1^1 \tag{64}$$

In the second stage, equation (53) shows that the vector Z^3 equals Q^3 so that the output can be directly connected to the first buffer of the Q^3 register. The $((X_3 + Y_2\text{-}\ 1) mod\ p) - th$ buffer of Q^3 and $Y_2 - th$ buffer of Q^2 are set to be the input of a XOR-gate, and the results of this gate are redundant parity bits of vector Z^2. In the last connection, the $((X_3 + Y_2 + Y_2\ \text{-}\ 2) mod\ p) - th$ buffer of Q^3, the $((Y_2 + Y_2\ \text{-}\ 1) mod\ p) - th$ buffer of Q^2, the $Y_3 - th$ buffer of Q^3, and the first buffer of Q^1 are set to be inputs of two XOR-gates, the results of each gates are set to be inputs of the last XOR-gates and the result of the last gates are redundant parity bits of vector Z^1. At the first clock cycle, the results are formed to be Z^1, Z^2, and Z^3.

By continuing to cyclically shift buffers of all Q to the left $p-1$ times until the end at the $p-th$ clock cycles, all redundant parity bits of Z can be generated. This stage requires a total of $3p$ flip-flops and 4 XOR-gates. Finally, we conclude that the two-stage encoding uses a total $2p$ clock cycles, Lp flip-flops and $(((L-3)-1)3\ +\ 4)$ XOR-gates.

TABLE I. COMPARISON OF ENCODING SCHEMES.

Encoding scheme	Clock Cycles	Flip-Flops	XOR-gates (Two-input)	AND-gates (Two-input)
QC-LDPC Serial SRAA [16]	(L-c)p	2cp	cp	cp
QC-LDPC Parallel SRAA [16]	cp	(L-c)p	(L-c)p-1	(L-c)p
QC-LDPC Two-stage [16]	p	Lp	O(c^2p)	0
QC-LDPC Proposed Two-Stage	2p	Lp	((L-3)-1)(3) + 4	0
QC-LDPC Proposed One-Stage	p	(L-3)p	((L-3)-1)(7) + 4	0

Note, c and L are the number of circulant permutation matrices of rows and columns of the parity-check matrix. p is the size of a circulant matrix.

B. One-stage encoding schemes

Now we reduced the stages of the two-stage encoding scheme to be a one-stage encoding scheme as shown in figure 2. In the redundant parity bit generating process, we set buffers of information registers to be inputs of XOR-gates at the first clock cycle by modifying (62) and (64) as follows.

$$Z_1^2 = \sum_{v=1}^{L-3} D^v_{((X_3+Y_2\text{-}1)\ mod\ p)} + \sum_{v=1}^{L-3} D^v_{((X_{v+3}+Y_2\text{-}1)\ mod\ p)} \tag{65}$$

$$Z_1^1 = \sum_{v=1}^{L-3} D^v_{((X_3+Y_2+Y_2\text{-}2)\ mod\ p)} + \sum_{v=1}^{L-3} D^v_{Y_3}$$
$$+\sum_{v=1}^{L-3} D^v_{((X_{v+3}+Y_2+Y_2\text{-}1)\ mod\ p)} + \sum_{v=1}^{L-3} D^v_{Y_{v+3}} \tag{66}$$

As for the first-stage of the two-stage scheme, all $L-3$ information sections are read into $L-3$ feedback shift registers in the fist clock cycle. All redundant parity bits of Z^3 can be generated using (53). However, the redundant parity bits of Z^2 and of Z^1 are formed by using (65) and (66), respectively. By continuing to shift information of the shift registers to the left $p-1$ times, all redundant parity bits of Z can be generated. This process requires a total of p clock cycles, $(L-3)p$ flip-flops and $(((L-3)-1)7\ +\ 4)$ XOR-gates.

Table I shows the encoding speed and complexity of both encoding schemes compared to other encoding schemes. One can see that the two-stage encoding scheme uses twice as many clock cycles as the one-stage encoding and the two-stage encoding of [16]. However, the advantage of both encoding schemes is that a few XOR-gates are used in encoder implementation.

V. CHANNEL MODEL

A block diagram of a magnetic recording system is shown in Fig.3. The $(3, 26)$ regular QC-LDPC code and irregular QC-LDPC code are error correcting codes. The magnetic recording channel is modeled as a perfectly equalized partial response (PR) channel that can be defined by target response as

$$h(D) = \sum_{i=0}^{v} h_i D_i \tag{67}$$

where v is the length of the channel memory and h_i is the $i-th$ channel coefficient. The information bits are encoded by the proposed code, the codeword $w=(w_1,\ w_2,\ \cdots,\ w_N)$ is mapped into antipodal encoded data symbols $r=(r_1,\ r_2,\ \cdots,\ r_N)$, where $r_i = 2w_i - 1$ and $i = 1, 2, 3, \cdots, N$, and then passed to the partial response channel with Additive White Gaussian Noise (AWGN). In this paper we assume that the soft output Viterbi algorithm (SOVA) [19] is used as the PR channel detector. The sum-product algorithm SPA is employed as the LDPC decoder. The signal to noise ratio (in dB) of the partial response channel is defined as

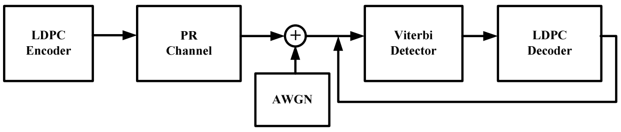

Figure 3. Diagram of concatenation LDPC code and PR channel.

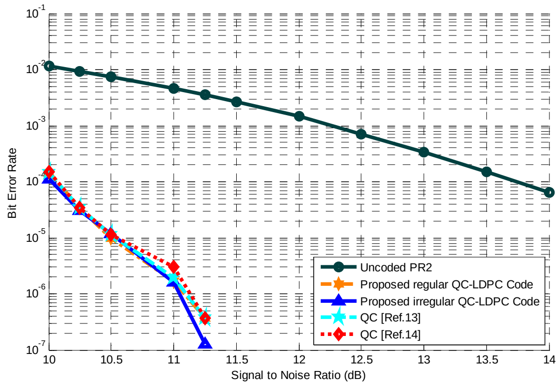

Figure 4. Comparison BER of the modified code (new encoding method) with other QC-LDPC codes over PR2 channel.

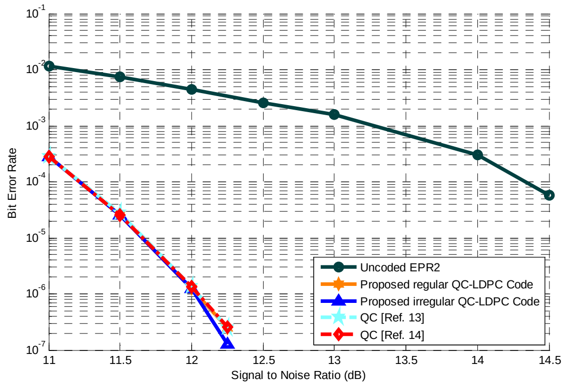

Figure 5. Comparison BER of the modified code (new encoding method) with other QC-LDPC codes over EPR2 channel.

$$SNR = 10\log_{10}\frac{\sum_{i=0}^{v} h_i^2}{\sigma^2} \tag{68}$$

where, σ^2 is the variance of the AWGN noise.

VI. PERFORMANCE RESULT

In the simulations, a $(3, 26)$ regular QC-LDPC code (7982, 7063) and an irregular QC-LDPC code (7982, 7061) were constructed from the proposed procedure. To compare the performance results, the other $(3, 26)$ regular QC-LDPC codes were constructed with the same parameters. In this paper, QC random code [13] and QC Tanner code [14] were used for comparison with the proposed code. All codes were constructed with circulant matrices of size 307×307 $p=307$. The partial response over the magnetic recording channels used Binary Phase Shift Keying (BPSK) modulation. For encoding of the regular QC-LDPC codes, Gaussian elimination was employed to yield the generator matrix whereas an irregular QC-LDPC code was used for the new encoding scheme. The sum-product algorithm SPA was used for all decoders.

Fig. 4 and fig. 5 show the performance of the proposed $(3, 26)$ regular QC-LDPC code and the proposed irregular QC-LDPC code at code rate 0.8846 over PR2 channel $(h(D) = 1 + 2D + D^2)$ and EPR2 channel $(h(D) = 1 + 3D + 3D^2 + D^3)$, respectively. The iteration between Viterbi detector and SPA was set as 3 while the iteration of SPA was set as 5. One can see from the figures that the proposed *regular* QC-LDPC code has performance very close to the other codes whereas the proposed *irregular* QC-LDPC code achieves the best BER performance in the high signal to noise region.

VII. CONCLUSION

We have designed and investigated some conditions for constructing the proposed parity-check matrix without short girth. The new structure of the $(3, L)$ regular QC-LDPC codes is based on circulant permutation matrices. The modified code is constructed by replacing some circulant permutation matrices with zero circulant matrices. In addition, we have presented two new efficient encoding techniques that can generate redundant bits of the codeword with lower complexity compared with the previous methods. The encoding complexity of the proposed code depends on the number of parity bits of the code for the one-stage encoding scheme, and the length of the code for the two-stage encoding scheme. Finally, we demonstrate the BER performance of the proposed code at high code rate on partial response channels.

REFERENCES

[1] R.G. Gallager, “Low density parity check codes,” IRE Trans. Inf. Theory, vol.IT-8, pp.21-28, Jan. 1962

[2] D.J.C. MacKay and R.M. Neal, “Good error correcting codes based on very sparse matrices,” IEEE Trans. Inf. Theory, vol.45, pp. 399-431, Mar. 1999.

[3] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1,” Proc. IEEE Int. Conf. on Commun., vol.2, pp.1064-1070, May 1993.

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

[5] T.J. Richardson, M.A. Shokrolahi, and R.L. Urbanke, “Design of capacity-approaching irregular low-density parity-check codes,” IEEE Trans. Inf. Theory, vol.47, no.2, pp.619-637, Feb. 2001.

[6] S.Y. Chung, G.D. Forney, T.J. Richardson, and R.L. Urbanke, “On the design of low density parity check codes within 0.0045 dB of the Shannon limit,” IEEE Commun .Lett., vol.5, no.2, pp.58-60, Feb. 2001.

[7] L. Chen, X. Jun, I. Djurdjevic, and S. Lin, “Near-Shannon-limit quasi-cyclic low-density parity-check codes,” IEEE Trans. Commun., vol.52, no.7, pp. 1038- 1042, July 2004.

[8] IEEE P802.11n TM/D1.02, “Draft Amendment to Standard Information Technology Part 11: Wireless Lan Medium Access Control (MAC) and Physical Layer (PHY) specification: Enhancements for higher Throughput,” IEEE 802.11 document, July 2006.

[9] IEEE P 802.16eTM, “Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access System,” IEEE 802.16 document, Feb 2005.

[10] A. Dholakia, E.Eleftheriou, T.Mittelholzer and M.P.C. Fossorier, “Capacity-aapproaching code; Can they be applied to the magnetic recording channel?,” IEEE Commun. Mag., vol. 42, no. 2, pp. 122-130, Feb. 2004.

[11] H. Zhong, T. Zhang and E. F. Hartsch, “Quasi-Cyclic LDPC code for the magnetic Recording Channel Code Design and VLSI Implementation,” IEEE Trans. Mag., vol.43, no.3, pp. 1118-1123, March. 2007.

[12] X. Liu, W. Zhang and Z. Fan, “Construct of Quasi-Cyclic LDPC Codes and the performance on the PR4 Equalizer MRC Channel,” IEEE Trans. Mag., vol.45, no.10, pp. 3699-3702, Oct. 2009.

[13] M.P.C. Fossorier, “Quasi-Cyclic low density parity check codes from circulant permutation matrices,” IEEE Trans. Inform. Theory, vol. 50, no. 8, pp. 1788-1794, Aug. 2004.

[14] A. Sridharan, D. J. Costello,Jr., D. Sridhara, T. E. Fuja, and R.M. Tanner, “LDPC Block and Convolutional Codes Based on Circulant Matrices,” IEEE Trans. Inform. Theory, vol. 50, no.12, pp. 2966-2984, Dec. 2004.

[15] S. Myung, K. Yang, and J. Kim, “Quasi-Cyclic LDPC codes for fast encoding,” IEEE Trans. Inform. Theory, vol. 51, pp. 2894-2901, Aug. 2005.

[16] Z. Li., L. Chen, L. Zeng, S. Lin, and W.H. Fong, “Efficient encoding of quasi-cyclic low density parity-check codes,” IEEE Trans. Commun., vol. 54, no.1, pp. 71-81, Jan. 2006.

[17] D.J.C. Mackay and M. Davey, “Evaluation of Gallager codes for short block length and high rate applications,” in Proc. IMA Workshop Codes, System-Margulis and Graphical Models, 1999.

[18] R.M. Tanner, ”A Recursive Approach to Low Complexity Codes,” IEEE Trans. Inform. Theory, vol. 27, pp. 533-547, Sep. 1981.

[19] J. Hagenuer and P. Hoecher, ”A Viterbi algorithm with soft decision output and its application,” in Proc. IEEE GLOBECOM, pp. 47.11-47, Dallas, TX, Nov. 1989.