Thesis for Master’s Degree

Parity check codes and their application to wireless communications: Fundamental limits and simulation results

Su-Je Lee

School of Information and Mechatronics

Gwangju Institute of Science and Technology

2012

석사학위논문

무선 통신에서의 패리티 검사 부호와 적용: 한계 분석과 모의실험 결과

이수제

정보기전공학부

광주과학기술원

2012

Parity check codes and their application to wireless communications: Fundamental limits and simulation results

Advisor : Heung-No Lee

by

Su-Je Lee

School of Information and Communications

Gwangju Institute of Science and Technology

A thesis submitted to the faculty of the Gwangju Institute of Science and Technology in partial fulfillment of the requirements for the degree of Master of Science in the School of Information and Communications

Gwangju, Republic of Korea

2011. 12. 19

Approved by

Professor Heung-No Lee

Thesis Advisor

Parity check codes and their application to wireless communications: Fundamental limits and simulation results

Su-Je Lee

Accepted in partial fulfillment of the requirements for the degree of Master of Science

December 19th 2011

Thesis Advisor ____________

Prof. Heung-No Lee

Committee Member ____________

Prof. Kiseon Kim

Committee Member ____________

Prof. Dong soo Har

MS/SIM 20104007

Su-Je Lee (이수제). Parity check codes and their applications to wireless communications: Fundamental limits and simulation results (무선 통신에서의 패리티 검사 부호와 적용: 한계분석과 모의실험 결과).

School of Information and Mechatronics. 2012. 46p. Prof. Heung-No Lee

Abstract

LDPC (low density parity check) codes and compressive sensing are the main topics covered in this thesis. We analyze LDPC codes. In particular, we focus on the distance spectrum, which is useful for analyzing the ML (maximum likelihood) decoding performance of linear parity check codes. However, the distance spectrum is difficult to obtain when the code has a long length. In fact, this is known to be an NP-hard problem. In this thesis, we review some existing methods, present a new algorithm to obtain the distance spectrum, and extend the proposed algorithm to the cases of non-binary regular LDPC codes and check-irregular LDPC codes. The distance spectrum is also used to analyze the RIP (restrict isometric property), which is an important measure of quality for a sensing matrix used for compressive sensing. According to our numerical results, the spark becomes larger as the weight and the GF (Galois field) size increase. We also present a new compressive sensing method based on an OFDM (orthogonal frequency division multiplexing) communication system. The simulation results show that the proposed OFDM system based on compressive sensing has a PAPR (peak-to-average power ratio) advantage compared to a conventional OFDM system.

Contents

List of Figures

List of Tables

1 Introduction

1.1 WHAT ARE PARITY CHECK CODES?

In communication systems, a parity check coding system is used for the robust transmission of data. Easily understood examples are the two basic types of parity check coding systems—even and odd. The respective systems check whether the number of ones is even or odd. This parity check system is used for the detection of errors. However, when we use concatenated parity checking systems, we can not only detect errors but also correct them [1][2].

1.2 LDPC CODES

For good performance, people have developed many different types of channel coding methods such as turbo codes, convolution codes, and RS codes. An LDPC (low density parity check) code is one of the channel coding methods invented in the 1960s by Gallager. This code had been forgotten, but was rediscovered in the 2000s because of its performance and simple decoding algorithm. As the code length increases, it is shown that the performance approaches the Shannon limit very closely. The decoding algorithm for LDPC codes is the so-called MP (message passing) algorithm, which is much simpler than ML decoding. In this thesis, we aim to obtain the distance spectrum of LDPC codes. This is done for the ML performance analysis of LDPC codes [1][2][3].

1.3 COMPRESSIVE SENSING

Compressive sensing theory is explained well in several magazine articles on signal processing [4][5]. These articles claimed that the conventional sampling method based on the Shannon-Nyquist sampling theorem implies that we may need too many samples for perfect reconstruction, and thus, we may need a new sampling theory. They also claimed that we need only a few samples for reconstruction of compressible (sparse) signals. They insisted that most signals are compressible.

Let a vector $\underline{x}$ be represented by

$$\underline{x} = \sum_{i=1}^{N} s_i \psi_i \ or \ \underline{x} = \Psi \underline{s} \tag{1}$$

where $\underline{x}$ and $\underline{s}$ are $N \times 1$ column vectors, and Ψ is an $N \times N$ orthonormal matrix. We call $\underline{x}$ $K-sparse$ if it can be represented as a linear combination of K basis vectors, which means there are K non-zero elements in vector $\underline{s}$. We also call $\underline{x}$ compressible if there are few non-zero elements and many zero elements. We project this value using the measurement matrix Θ

$$\underline{y} = \Phi \underline{x} = \left(\Phi \Psi\right) \underline{s} = \Theta \underline{s} \tag{2}$$

where the sensing matrix Θ is $M \times N$, and $M << N$.

For a $K-sparse$ signal $\underline{x}$, $L1$ minimization gives the unique solution $\underline{x}$ under the sufficient condition that we call the RIP (restricted isometric property). Because $\underline{s}$ is a $K-sparse$ vector, when we know about the locations of non-zero elements, we can solve the problem when $M > K$. In other words, we can simplify the equation by reducing all the columns and elements corresponding to the zero elements. Thus, we can obtain the following equation:

$$\underline{y} = \Theta_K \underline{s}_K \tag{3}$$

where $\underline{s}_K$ is the set of non-zero elements, and Θ_K is the collected set corresponding to $\underline{s}_K$. However, this equation has a unique solution if all columns of Θ_K are linearly independent. If this is the case, the solution can be found by (4).

$$\underline{s}_K = \left(\Theta_K^T \Theta_K\right)^{-1} \Theta_K^T \underline{y} \tag{4}$$

Any K columns of the Θ matrix should be linearly independent and preserve the energy for reconstruction. This condition is as follows:

$$1-\varepsilon \le \frac{\|\Theta v\|_2}{\|v\|_2} \le 1+\varepsilon \qquad (5)$$

for the existence of a unique $L0$ -minimization solution, $2K \le v$. This result follows the RIP.

A linear combination of any $2K$ columns should not be a “zero” vector. In other words, $2K$ or smaller dependent column sets do not exist. In addition, we define the word “spark” used in compressive sensing. The spark is the smallest number S such that there exists a set of S columns of the matrix that are linearly dependent. In fact, this is the minimum distance of a parity check matrix [6][7].

1.4 THESIS OUTLINE

The rest of the paper is organized as follows. In Section II, we present the distance spectrum, existing methods, and proposed methods. In Section III, we introduce the relationship between the distance spectrum and the RIP and analyze it with different densities and field sizes. In Section IV, we provide a new communication method using OFDM and compressive sensing, and in Section V, we summarize the paper.

2 Distance spectrum

2.1 BASIC INFORMATION ABOUT THE DISTANCE SPECTRUM

The distance spectrum is useful for the union bound evaluation of channel codes. Gallager has provided a framework to obtain the exponent of the distance spectrum for regular LDPC codes in his thesis [3]. In this context, the term "regular" means that the number of ones in each column and in each row of a parity-check matrix is the same and fixed for an ensemble. In this chapter, we provide some existing and proposed methods to calculate the distance spectrum.

2.2 EXISTING METHODS

2.2.1 LITSYN'S DISTANCE SPECTRUM

In this section, Litsyn's method will be written for calculating the average distance spectrum of regular LDPC codes [8]. In brief, Litsyn calculates the distance spectrum by obtaining the proportion of matrices $\Lambda_{n,w}^{j,k}$ in ensemble $\Lambda_{n}^{j,k}$.

$\Lambda_{n}^{j,k}$: Total (n, j, k) regular LDPC matrices

$\Lambda_{n,w}^{j,k}$: (n, j, k) regular LDPC matrices that have first *w*-ones word as their code word

Figure 1 shows a first *w*-ones word.

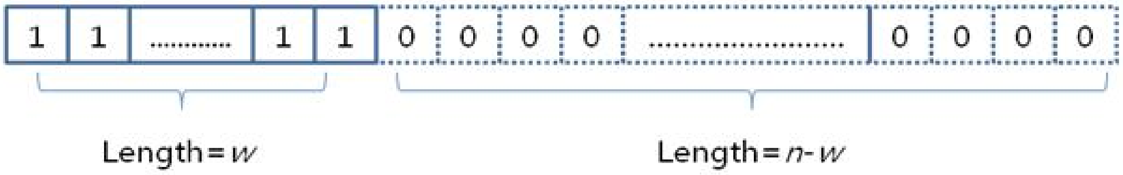


Figure 1. First w-ones word

Before calculating $\Lambda_{n}^{j,k}$ and $\Lambda_{n,w}^{j,k}$, we should know O'Neil's method, which gives us a tool to calculate the number of matrices based on column and row sum profiles. O'Neil's method starts with the following inequality:

$$\max\left\{\max_{1\le i\le n} k_i, \max_{1\le j\le m} l_j\right\} \le \left(\ln n\right)^{\frac{1}{4}-\varepsilon} \quad , \textit{certain } \varepsilon > 0 \tag{6}$$

where k_i is the number of ones in the *i*-th column, and l_j is the number of ones in the *j*-th row. This equation implies that there is a certain limit on the maximum column (row) density related to signal length.

Through this information, we can calculate the number of possible matrices with (7), when the number of ones of each column and row are given. Then, for $\delta > 0$

$$\begin{aligned}\left|\Lambda_{m,n,\mathcal{K},\mathcal{L}}\right| &= \frac{\left(\sum_{i=1}^{m} k_i\right)!}{\prod_{i=1}^{m} k_i! \prod_{i=1}^{n} l_i!} \times \\ &\exp\left(\frac{-1}{1\left(\sum_{i=1}^{m} k_i\right)^2}\left(\sum_{i=1}^{m} k_i\left(k_i-1\right)\sum_{i=1}^{n} l_i\left(l_i-1\right)\right)\right)\times\left(1+o\left(n^{-1+\delta}\right)\right)\end{aligned} \tag{7}$$

where $|H|$ denotes the number of possible matrices H, n is the width of the matrix, m is the height of the matrix, $\mathcal{K}$ denotes the set of k_i, and $\mathcal{L}$ denotes the set of l_i. In regular LDPC cases, every $k_i(l_i)$ is the same as *k*(*j*). Using (7), we can calculate $\left|\Lambda_n^{j,k}\right|$ and $\left|\Lambda_{n,w}^{j,k}\right|$. First, let us calculate $\left|\Lambda_n^{j,k}\right|$. This $\Lambda_n^{j,k}$ is the matrix that has *k* ones in every row and *j* ones in every column. Thus, when we adjust O'Neil's method, $\left|\Lambda_n^{j,k}\right|$ is as follows.

$$\left|\Lambda_n^{j,k}\right| = \frac{(nj)!}{(k!)^{n*\frac{j}{k}}(j!)^n} \times \exp\left(-\frac{(k-1)(j-1)}{2}\right) \times \left(1+o\left(n^{-1+\delta}\right)\right) \tag{8}$$

Next, let us calculate $\left|\Lambda_{n,w}^{j,k}\right|$. This $\Lambda_{n,w}^{j,k}$ is the subset of $\Lambda_n^{j,k}$ that has the first *w*-ones word as its code word. How can we obtain the number of possible matrices? It is our objective to find the number of matrices such that the sub-matrix consisting of the first *w* columns has an even row sum. In this method, we divide matrix $\Lambda_{n,w}^{j,k}$ into two parts, $L_{n,w}^{j,k}$ and $R_{n,w}^{j,k}$. $L_{n,w}^{j,k}$(size $m \times w$) is made from the left *w* columns of matrix $\Lambda_{n,w}^{j,k}$, and $R_{n,w}^{j,k}$ (size $m \times (n-w)$) is made from the right *n-w* columns of matrix $\Lambda_{n,w}^{j,k}$. Let m_i denote the number of rows in $L_{n,w}^{j,k}$ with a sum equal to *i*. When we consider *k* as even, $i \in \{0, 2, ..., k\}$. In this case, the following equations are valid:

$$\begin{gathered} m_0 + m_2 + \cdots + m_k = \frac{nj}{k} \\ 0m_0 + 2m_2 + \cdots km_k = jw \end{gathered} \tag{9}$$

Then, $\left|\Lambda_{n,w}^{j,k}\right|$ can be calculated by multiplying two sub-matrices in every case.

$$\left|\Lambda_{n,w}^{j,k}\right| = \sum \binom{jn/k}{m_0, m_2, ..., m_k} \left|L_{n,w}^{j,k}\right| \left|R_{n,w}^{j,k}\right| \tag{10}$$

Here, $\binom{jn/k}{m_0, m_2, ..., m_k}$ means the following multinomial coefficient:

$$\binom{jn/k}{m_0, m_2, ..., m_k} = \frac{(jn/k)!}{\prod_{i=0}^{k/2} m_{2i}!} \tag{11}$$

In case of a fixed m_i subset, we can calculate $\left|L_{n,w}^{j,k}\right|$ by using (7). Because $L_{n,w}^{j,k}$ is a subset of the regular LDPC codes, every column has *j*-ones. Moreover, there are m_i rows that have *i*-ones. Thus, $\left|L_{n,w}^{j,k}\right|$ is as follows:

$$\begin{aligned}\left|L_{n,w}^{j,k}\right| &= \frac{(jw)!}{(j!)^{w}\,2!^{m_2}\,4!^{m_4}\cdots k!^{m_k}}\times \\ &\exp\left(\frac{-1}{2(jw)^2}j(j-1)w\left(2m_2+12m_4+\cdots+k(k-1)m_k\right)\right)\times\left(1+o\left(n^{-1+\delta}\right)\right)\end{aligned} \tag{12}$$

When the m_i subset is fixed, the number of ones in $R_{n,w}^{j,k}$ is also determined. Thus, $\left|R_{n,w}^{j,k}\right|$ is given as follows:

$$\begin{aligned}\left|R_{n,w}^{j,k}\right| &= \frac{(j(n-w))!}{j!^{(n-w)}\,k!^{m_0}\,(k-2)!^{m_2}\cdots 2!^{m_{k-2}}}\times \\ &\exp\left(\begin{array}{l}\frac{-1}{2(j(n-w))^2}j(j-1)(n-w)\times \\ \left(k(k-1)m_0+(k-2)(k-3)m_2+\cdots+2m_{k-2}\right)\end{array}\right)\times\left(1+o\left(n^{-1+\delta}\right)\right)\end{aligned} \tag{13}$$

From (10), (11), (12), and (13), we can calculate $\left|\Lambda_{n,w}^{j,k}\right|$ as follows:

$$\left|\Lambda_{n,w}^{j,k}\right| = \sum\left(\begin{array}{l}\frac{(jn/k)!}{\prod_{i=0}^{k/2} m_{2i}!}\frac{(jw)!}{(j!)^{w}\,2!^{m_2}\,4!^{m_4}\cdots k!^{m_k}}\exp\left(\frac{-1}{2(jw)^2}j(j-1)w\left(2m_2+12m_4+\cdots+k(k-1)m_k\right)\right) \\ \frac{(j(n-w))!}{j!^{(n-w)}\,k!^{m_0}\,(k-2)!^{m_2}\cdots 2!^{m_{k-2}}}\exp\left(\begin{array}{l}\frac{-1}{2(j(n-w))^2}j(j-1)(n-w)\times \\ \left(k(k-1)m_0+(k-2)(k-3)m_2+\cdots+2m_{k-2}\right)\end{array}\right)\times\left(1+o\left(n^{-1+\delta}\right)\right)\end{array}\right) \tag{14}$$

O'Neil used the symbol "$\overset{\ln}{\sim}$." This symbol expresses logarithmic equivalence, such that $a_n \overset{\ln}{\sim} b_n$ if $\ln a_n \sim \ln b_n$. He ignored the exponential term of the result using this notation. Therefore, (14) changes as follows:

$$\begin{aligned}\left|\Lambda_{n,w}^{j,k}\right| &\overset{\ln}{\sim} \frac{(jw)!\left(k(n-w)\right)!}{(j)!^{n}} * \\ &\sum\frac{(w)!}{m_0!(0!k!)^{m_0}\,m_2!\left(2!(k-2)!\right)^{m_2}\cdots m_k!(k!0!)^{m_k}}\end{aligned} \tag{15}$$

Eq. (8) also changes as follows:

$$\left|\Lambda_n^{j,k}\right| \overset{\ln}{\sim} \frac{(nj)!}{(k!)^{n*\frac{j}{k}}(j!)^n} \tag{16}$$

Then, we can calculate the proportion $\left(P_{n,w}^{j,k}\right)$ of matrices $\Lambda_{n,w}^{j,k}$ in ensemble $\Lambda_n^{j,k}$.

$$P_{n,w}^{j,k} = \frac{\left|\Lambda_{n,w}^{j,k}\right|}{\left|\Lambda_n^{j,k}\right|} \tag{17}$$

Since we only considered the first *w*-ones words at first, we need to multiply by the number of all *w*-weight words. The average number of weight *w* words is as follows:

$$A_w = \binom{n}{w} \times P_{n,w}^{j,k} \tag{18}$$

The exponent of the distance spectrum is as follows:

$$B_w = \frac{1}{n}\ln A_w = \frac{1}{n}\ln\binom{n}{w} + \frac{1}{n}P_{n,w}^{j,k} \tag{19}$$

2.2.2 BURSHTEIN'S DISTANCE SPECTRUM

In this section, Burshtein's method [9] will be described. Burshstein derived the asymptotic distance spectrum of LDPC codes. He used a Tanner graph representation [10], a bipartite graph, [11] for convenience. In the bipartite graph, variable nodes are located on the left side, and check nodes are located on the right side. Edges connect both sides according to the parity check codes. Figure 2 shows a basic bipartite graph of LDPC codes.

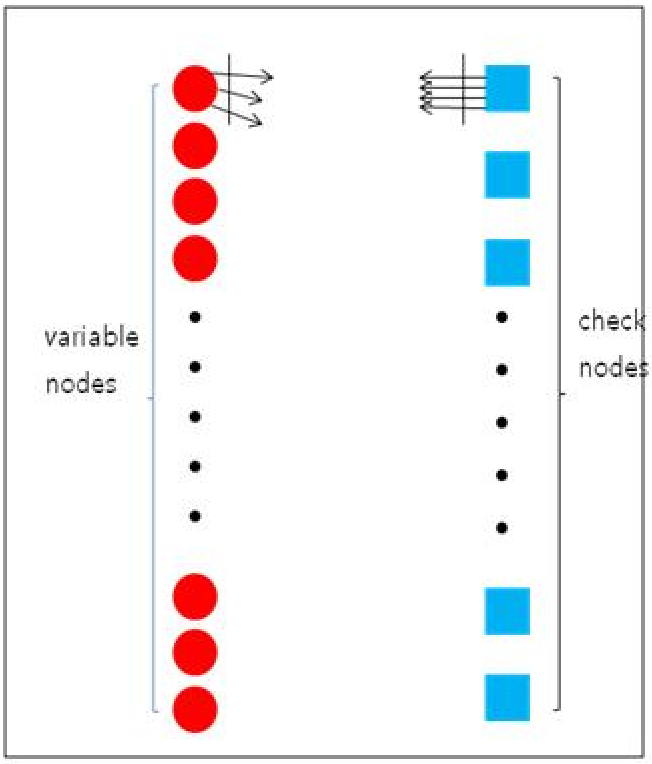


Figure 2. Bipartite graph representation of LDPC codes

(n,j,k) LDPC codes has n variable nodes and $m\left(=nj/k\right)$ check nodes. The total number of edges (E) is obtained as follows:

$$E = n \times j = m \times k \tag{20}$$

The code rate R is obtained as follows:

$$R = 1 - \frac{m}{n} \tag{21}$$

This method uses enumerating functions, which is very useful in solving combinatorial problems. Additionally, this problem contains powers of polynomials with non-negative coefficients.

Let us obtain the distance spectrum of regular LDPC codes. When an r.v (random variable) S_w is equal to the number of code words of weight "*w*" in the given code, the average distance spectrum is obtained as follows:

$$A_w = E\left(S_w\right) \tag{22}$$

For any j, such that $1 \le j \le \binom{n}{w}$, we take X_w^j as an r.v that is "1" if the *j*-th weight *w* word is a code word of the given LDPC codes, and "0" otherwise. Then,

$$S_w = \sum_{j=1}^{\binom{n}{w}} X_w^j \tag{23}$$

Therefore, the distance spectrum is

$$A_w = E\left(\sum_{j=1}^{\binom{n}{w}} X_w^j\right) \tag{24}$$

Because the expectation is only related to random variable X_w^j, (24) becomes

$$A_w = E\left(\sum_{j=1}^{\binom{n}{w}} X_w^j\right) = \sum_{j=1}^{\binom{n}{w}} E\left(X_w^j\right) \tag{25}$$

$E\left(X_w^j\right)$ can be written as $P\left(X_w^j = 1\right)$, so (25) becomes

$$A_w = E\left(\sum_{j=1}^{\binom{n}{w}} X_w^j\right) = \sum_{j=1}^{\binom{n}{w}} E\left(X_w^j\right) = \sum_{j=1}^{\binom{n}{w}} P\left(X_w^j = 1\right) \tag{26}$$

Without loss of generality, $P\left(X_w^j = 1\right)$ is equal to $P\left(X_w^1 = 1\right)$, so (26) becomes

$$A_w = E\left(\sum_{j=1}^{\binom{n}{w}} X_w^j\right) = \sum_{j=1}^{\binom{n}{w}} E\left(X_w^j\right) = \sum_{j=1}^{\binom{n}{w}} P\left(X_w^j = 1\right) = \sum_{j=1}^{\binom{n}{w}} P\left(X_w^1 = 1\right) = \binom{n}{w} P\left(X_w^1 = 1\right) \tag{27}$$

The exponent of the distance spectrum B_w is obtained as follows:

$$B_w = \frac{1}{n}\ln\left(A_w\right) = \frac{1}{n}\ln\left(\binom{n}{w}P\left(X_w^1 = 1\right)\right) \tag{28}$$

By using a property of binomial coefficients,

$$\ln\binom{n}{\alpha n} = n\left[h\left(\alpha\right) + o\left(1\right)\right] \tag{29}$$

We can write (28) as follows:

$$B_w = h\left(\frac{w}{n}\right) + \frac{\ln P\left(X_w^1 = 1\right)}{n} + o\left(1\right) \tag{30}$$

Thus, the problem of obtaining B_w becomes the problem of obtaining $P\left(X_w^1 = 1\right)$. X_w^1 is the random variable related to the first weight *w* word. We can consider this word as a first *w*-ones word, as shown in Figure 1. Let us denote this word as v and the *w*-ones variable nodes as *S*.

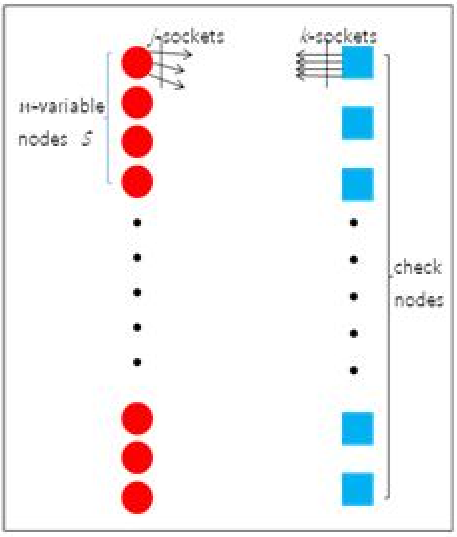


Figure 3. *v*-word (variable set S)

Then, the v-word is a code word if and only if every check node is connected to variable set *S* by an even number of edges.

We can calculate $P\left(X_w^1 = 1\right)$ by dividing the number of cases $e\left(w;n\right)$, where each check node is connected to a variable set *S* by an even number of edges, by the total number of cases $t\left(w;n\right)$, where each check node is connected to a variable set *S* (length *w*) arbitrarily. In regular (n, j, k) LDPC matrix cases, each variable node has *j* sockets and each check node has *k* sockets. Set *S* has $t\left(w;n\right)$ that is calculated as follows:

$$t\left(w;n\right) = \begin{pmatrix} k \cdot m \\ j \cdot w \end{pmatrix} = \begin{pmatrix} j \cdot n \\ j \cdot w \end{pmatrix} \tag{31}$$

When we consider the jw sockets of *S*, they are connected to any choice of jw out of the *km* (= jn) right sockets. Next, we need to calculate $e\left(w;n\right)$ in which each check node is connected an even number of times with *S*. This is easily done by using an enumerating function, and the result is as follows:

$$\begin{aligned} e\left(w;n\right) &= coef\left(\left(\sum_{i=0}^{\lfloor k/2 \rfloor} \begin{pmatrix} k \\ 2i \end{pmatrix} x^{2i}\right)^m, jw\right) \\ &= coef\left(\left(\frac{1}{2}\left[\left(1+x\right)^k + \left(1-x\right)^k\right]\right)^m, jw\right) \end{aligned} \tag{32}$$

From the proceeding discussion, we have

$$\begin{aligned} &P\left(X_w^1 = 1\right) = \frac{e\left(w;n\right)}{t\left(w;n\right)} \\ &\frac{1}{n}\ln P\left(X_w^1 = 1\right) = \frac{1}{n}\ln\frac{e\left(w;n\right)}{t\left(w;n\right)} = \frac{1}{n}\ln e\left(w;n\right) - \frac{1}{n}\ln t\left(w;n\right) \end{aligned} \tag{33}$$

This method uses some approximations. For a positive, rational number $\gamma > 0$; a polynomial with non-negative coefficients $p\left(x\right)^\gamma$; and a rational number $\alpha > 0$, the following inequality is valid:

$$coef\left(p\left(x\right)^n, \alpha n\right) \le \inf_{x>0} \frac{p\left(x\right)^n}{x^{\alpha n}} \tag{34}$$

In addition,

$$\lim_{x\to\infty}\frac{1}{n}\ln coef\left(p(x)^{n},\alpha n\right)=\inf_{x>0}\ln\frac{p(x)}{x^{\alpha}} \tag{35}$$

Using (33) and (35), we have

$$\begin{aligned}
\lim_{n\to\infty}\frac{\ln e(w;n)}{n} &= (1-R)\lim_{n\to\infty}\frac{1}{(1-R)n}\ln\left\{coef\left(\left(\frac{1}{2}\left[(1+x)^{k}+(1-x)^{k}\right]\right)^{m},jw\right)\right\} \\
&= (1-R)\lim_{n\to\infty}\frac{1}{n}\ln\ln\left\{coef\left(\left(\frac{1}{2}\left[(1+x)^{k}+(1-x)^{k}\right]\right)^{n},kw\right)\right\} \\
&= (1-R)\ln\left\{\inf_{x>0}\frac{(1+x)^{k}+(1-x)^{k}}{2x^{\frac{w}{n}k}}\right\}
\end{aligned} \tag{36}$$

From (31) and (29), we have

$$\frac{\ln t(w;n)}{n}=j\cdot h\left(\frac{w}{n}\right)+o(1) \tag{37}$$

Therefore, by combining (30), (33), (36), and (37), we obtain the following exponent of the distance spectrum. For any $0<w<n$

$$B_{w}=(1-R)\ln\left\{\inf_{x>0}\frac{(1+x)^{k}+(1-x)^{k}}{2x^{wk/n}}\right\}-h(\alpha)(j-1) \tag{38}$$

2.2.3 KASAI'S DISTANCE SPECTRUM

The previous two methods are used to obtain the distance spectrum in the binary case. Now, we aim to study how to obtain the distance spectrum in non-binary cases. This method [12] uses a bipartite graph, as in the case of Burshtein's method. An enumerator function and the polynomial method are also used. Let us investigate them in detail. There is only a single type of edge in the binary case; we need to add edge labels to a non-binary LDPC bipartite graph. The edge labels range from 1 to $(q-1)$. Thus, we can represent a non-binary LDPC bipartite graph as follows:

Variable nodes : $v_1, v_2, ..., v_n$

Check nodes : $c_1, c_2, ..., c_m$

Edge : $e=(v,c)\in E$, is labeled $h_e \in GF(q)\setminus\{0\}$

Let $\vartheta(n,j,k,q)$ be a code ensemble of regular non-binary LDPC codes, where q indicates the GF size. The total number of code ensembles ($\#\vartheta$) is

$$\#\vartheta = E!(q-1)^E \tag{39}$$

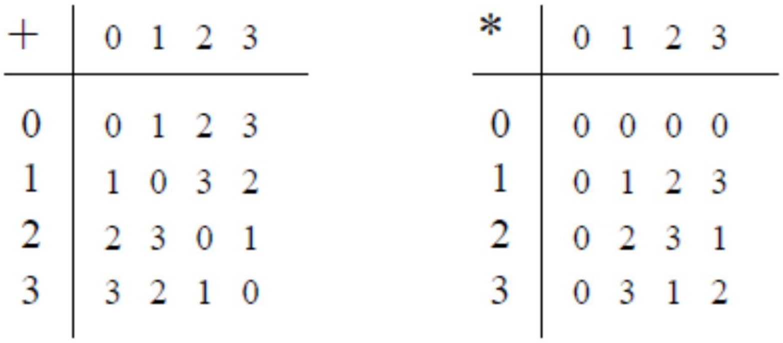
where E denotes the total number of edges. We can calculate E with (20).

Let G be a randomly selected regular non-binary LDPC code, and let $N(G,w)$ be the number of code words of symbol-weight "*w*" in a code G. Let $\vartheta_{\underline{x}_w} := \{G\in\vartheta \mid \underline{x}_w \in G\}$ be the graphs that have $\underline{x}_w$ (symbol-weight "w") as their code word.

An important point to remember is that the number of graphs in $\vartheta_{\underline{x}_w}$ is the same as the number of graphs in $\vartheta_{\underline{1}_w}$. In this case, $\underline{1}_w$ denotes a word whose locations of non-zero elements in $\underline{x}_w$ are filled by one. Now, let us prove this. $h_{(v,c)}$ denotes the label of the edge (v,c) in G. Then, the label of the edge of the image graph G' has the same connection with G, but the values are changed as follows:

$$h'(v_i, c_i) = x_i^{-1} h(v_i, c_i) \tag{40}$$

Figure 4 shows an addition and multiplication table of $GF(q)$. Let us consider the following example.



+	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

*	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	3	1
3	0	3	1	2

Figure 4. GF(4) addition and multiplication table

Let us suppose that matrix G is $\begin{bmatrix} 1\,1\,1\,0\,1 \\ 1\,2\,3\,0\,0 \\ 0\,0\,2\,0\,1 \end{bmatrix}$. In this case, one of the code words is $[1\,0\,2\,0\,3]^T$

$$\begin{bmatrix} 1\,1\,1\,0\,1 \\ 1\,2\,3\,0\,0 \\ 0\,0\,2\,0\,1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 2 \\ 0 \\ 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

Through (40), we can change G to G' as follows:

$$\begin{bmatrix} 1\,1\,1\,0\,1 \\ 1\,2\,3\,0\,0 \\ 0\,0\,2\,0\,1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 2 \\ 0 \\ 3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 1\,1\,2\,0\,3 \\ 1\,2\,1\,0\,0 \\ 0\,0\,3\,0\,3 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Therefore, all graphs that have $\underline{x}_w$ as their code words can be changed to graphs that have $\underline{1}_w$ as their code words, implying that the number of code ensembles in $\vartheta_{\underline{x}_w}$ is the same as that in $\vartheta_{\underline{1}_w}$. With this property, we can derive the average distance spectrum $E[N(G,w)]$ as follows:

$$
\begin{aligned}
&E\left[N(G,w)\right] \\
&= \frac{1}{\#\vartheta}\sum_{G\in\vartheta}\#\left\{\underline{x}\in G \mid W_H\left(x\right)=w\right\} \\
&= \frac{1}{\#\vartheta}\sum_{\underline{x}:W_H(\underline{x})=w}\#\left\{G\in\vartheta \mid \underline{x}\in G\right\} \\
&= \frac{1}{\#\vartheta}\underbrace{\binom{n}{w}(q-1)^w}_{\text{variable constellation}}\#\left\{G\in\vartheta \mid \underline{1}\in G\right\}.
\end{aligned}
\tag{41}
$$

where $W_H\left(x\right)$ denotes the symbol weight of x. We can calculate $\#\vartheta$ using (39).

If we calculate $\#\left\{G\in\vartheta \mid \underline{1}\in G\right\}$, we can calculate the average distance spectrum of a non-binary regular LDPC code. $\#\left\{G\in\vartheta \mid \underline{1}\in G\right\}$ denotes the total number of graphs that have $\underline{1}_w$ as their code word. Without loss of generality, we can assume the code word to be $\underline{1}_w=\left(\underbrace{\overbrace{1,1,....,1}^{w},0,0,....,0}_{n}\right)$. In this case, the graph is the same as that shown in Figure 3. Now, we need to find the check node allocation to satisfy the check equation. In the binary case, the sufficient condition is that all check nodes are connected with w variable nodes, with an even number of edges. However, for the non-binary case, this is not sufficient. Therefore, let us calculate the number of combinations by which one check node connects $\underline{1}_w$ and still satisfies the equation. We say an edge is active if it connects to the w variable node set (set of non-zero elements in $\underline{1}_w$). Let $C_0\left(h\right)$ be the number of combinations such that the sum of h non-zero numbers from $\mathrm{GF}\left(q\right)$ is equal to 0:

$$
C_0\left(h\right)=\frac{\left(-1\right)^h\left(q-1\right)+\left(q-1\right)^h}{q}
$$

We prove this in Appendix A.

Then, the number of combinations that one check node can connect $\underline{1}_w$ with h active nodes is

$$b(h)=\binom{k}{h}C_0(h)$$
$$b(h)=\binom{k}{h}\frac{(-1)^h(q-1)+(q-1)^h}{q} \tag{42}$$

Thus, the number of ways of allocating km sockets (m is the number of check nodes) with wj active labeling that satisfy all m check nodes can be calculated by the m-fold convolution of $b(h)$

$$b^{\otimes m}(wj)=\sum_{\sum_{i=1}^{m}k_i=wj}\prod_{i=1}^{m}b(k_i)=coef\left(f_j(u)^m,u^{wj}\right) \tag{43}$$

where we define $f_j(u)$ as

$$f_j(u)=\sum_{l=0}^{j}b(l)u^l=\frac{(1+(q-1)u)^j+(q-1)(1-u)^j}{q} \tag{44}$$

Now, we can calculate $\#\{G\in\vartheta\,|\,\underline{1}\in G\}$. Because there are $(E-jw)!$ permutations of non-active edges and $jw!$ permutations of active edges, $\#\{G\in\vartheta\,|\,\underline{1}\in G\}$ is as follows:

$$\#\{G\in\vartheta\,|\,\underline{1}\in G\}=(E-jw)!(jw)!(q-1)^{E-jw}coef\left(f_k(u)^m,u^{jw}\right) \tag{45}$$

Then, we can calculate the distance spectrum of non-binary regular LDPC codes as follows:

$$E\left[N(G,w)\right]=\binom{n}{w}\frac{coef\left(f_k(u)^m,u^{jw}\right)}{\binom{E}{jw}(q-1)^{(j-1)w}} \tag{46}$$

Thus, the exponent of the non-binary regular LDPC codes is as follows:

$$B_w=\frac{1}{n}\ln E\left[N(G,w)\right] \tag{46}$$

2.3 PROPOSED METHOD

The method proposed in this section was developed on the basis of the unpublished work of Lee [13]. In [13], Lee reported the convolution- and polynomial-based routines to calculate the distance spectra of linear LDPC and LDGM codes. In this thesis, we extend Lee's framework to obtain the distance spectra of non-binary and irregular LDPC codes. We refer to this algorithm as Lee's method in this thesis.

2.3.1 PROPOSED DISTANCE SPECTRUM OF BINARY LDPC CODES

Figure 5 and Figure 6 show a regular Gallager LDPC code. The first matrix is generated by allocating all allowed ones in a regular manner, as shown in Figure 5. This regularity is convenient for analysis. By independent permutations of the first sub-matrix, we obtain the second, third, and the rest sub-matrices. We calculate the ensemble-averaged distance spectrum with the following steps. First, find the total number of code words with weight *w* and length *n* that satisfy the first sub-matrix and denote this number as N_w. Second, divide N_w by $\binom{n}{w}$, which is the total number of weight *w* length *n* words for the entire matrix. This ratio (p_w) is the proportion of length *n* and weight *w* words that satisfy the first sub-matrix. In addition, this ratio remains the same for all other sub-matrices that are constructed as permutations of the first sub-matrix, whereas the sets of words satisfying each sub-matrix are different from one another. Third, calculate the probability of any weight *w* length *n* word satisfying all sub-matrices simultaneously, which becomes the product of the ratio and j.

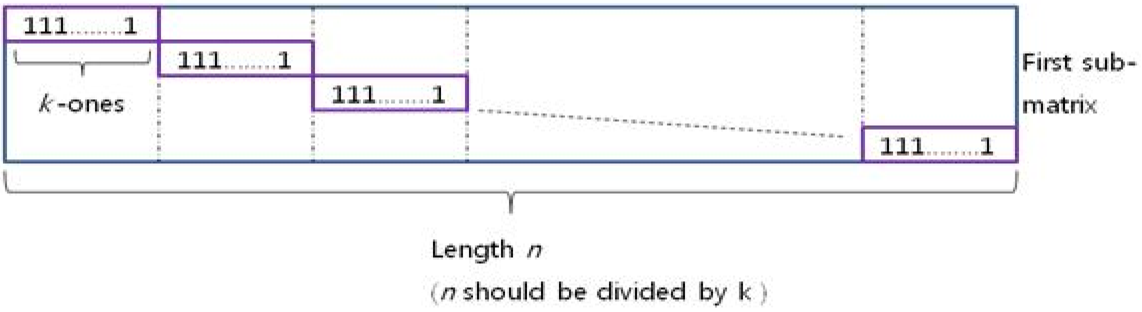


Figure 5. First sub-matrix of regular LDPC matrix

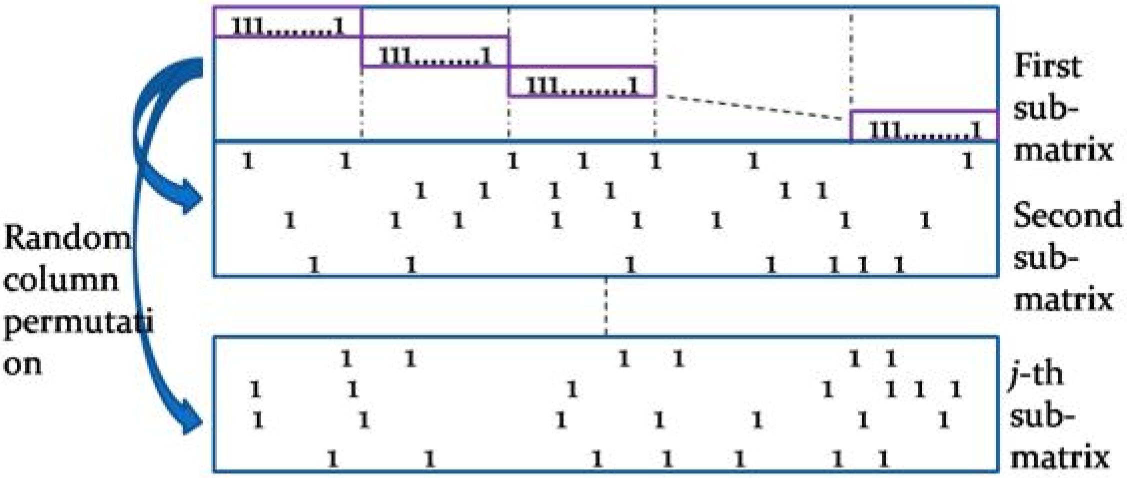


Figure 6. Rule for constructing LDPC matrix

For a regular (*n*, *j*, *k*) LDPC code, its distance spectrum can be obtained as follows.

We construct a basic combination sequence x_i, which satisfies $1_k * x_i = 0$.

$$x_i = \begin{cases} \begin{pmatrix} k \\ i \end{pmatrix} & 0 \le i \le k, i\, is\, even \\ 0 & 0 \le i \le k, i\, is\, odd \end{cases} \tag{47}$$

Then, we construct a stand sequence (N_w) that satisfies the first sub-matrix

$$N_w = \underbrace{x \circ x \circ x \circ \cdots\cdots \circ x}_{\frac{n}{k}\ convolution}, \qquad w \in (0, n)\ positive\ number \tag{48}$$

where "$\circ$" indicates convolution, and w is the weight.

The proportion of length *n* and weight *w* words is as follows:

$$p_w = \frac{N_w}{\binom{n}{w}} \tag{49}$$

We can find the distance spectrum (S_w) and the normalized distance spectrum (B_w) of the regular (n, j, k) LDPC code as follows:

$$S_w = p_w{}^j * \binom{n}{w}$$

(50)

$$B_w = \frac{1}{n} \ln S_w \tag{51}$$

The following is a complete algorithm to obtain the distance spectrum of regular (n, j, k) LDPC codes.

Algorithm for obtaining distance spectrum

1. Calculate the number of weight w code words (N_w) satisfying the first sub-matrix.
2. Obtain p_w by dividing N_w by the total number of weight w words (T_w).

   For example, in the binary case, $T_w = \binom{n}{w}$, $p_w = N_w / T_w$

3. This probability is also usable in other sub-matrices. Thus, the probability that a weight w word is a code word of an (n, j, k) LDPC matrix is $(p_w)^j$.
4. Obtain the distance spectrum from $A_w = (p_w)^j \times T_w$ and the normalized distance spectrum from $B_w = \frac{1}{n} \ln A_w$

This algorithm is the essence of Lee’s method.

2.3.2 PROPOSED DISTANCE SPECTRUM OF IRREGULAR LDPC CODES

In this thesis, we have extended Lee's method to handle check irregular LDPC codes. In check regular codes, check node degrees can have various values but the variable degree is fixed to one value. From Tanner graph analysis [11], it is found that $\vartheta\left(N,\lambda\left(x\right),\rho\left(x\right)\right)$ represents irregular LDPC codes. N is the signal length, $\lambda\left(x\right)$ represents the degree of the variable node, and $\rho\left(x\right)$ represents the degree of the check node. In our case, $\lambda\left(x\right)$ has only one polynomial equation, and $\rho\left(x\right)$ can have various polynomial equations.

$$\lambda\left(x\right)=\sum_{i=2}^{d_v}\lambda_i x^{i-1},\rho\left(x\right)=\sum_{j=2}^{d_c}\rho_j x^{j-1}$$

where λ_i denotes a fraction of all edges connected to degree *i* variable nodes, and ρ_i denotes a fraction of all edges connected to *i* check nodes.

Now, let us obtain the distance spectrum of the check irregular case. Likewise, we need to obtain the basic combination sequence. However, in this case, we should obtain number *T* of different basic combination sequences. *T* is defined as the number of non-zero values in $\rho_j\left(1\le j\le d_c\right)$.

Thus, we find *T* kinds of x_i according to each degree by using (47). We denote these as x_i^d $\left(1\le d\le T\right)$. We also need to find *T* kinds of convolution sequences. These are generated from convolution according to each fraction ρ_{T_i}

$$N_{T_j}=\underbrace{x_i^j\circ x_i^j\circ x_i^j\cdots\circ x_i^j}_{\left(N/T_j\right)*\rho_j\,convolution}$$

Next, we find *N* as follows:

$$\begin{aligned}&N=N_1\circ N_2\circ\cdots\circ N_T,\quad N_h=N(h)\\&h\in(0,n)\;positive\,number\end{aligned}\qquad(52)$$

In addition, we obtain the distance spectrum of check irregular LDPC codes by (49), (50) and, (51).

2.3.3 PROPOSED DISTANCE SPECTRUM OF NON-BINARY LDPC CODES

In this section, we derive the distance spectrum of non-binary LDPC codes. We use the same process in the case of binary LDPC codes. Thus, we first obtain the basic combination sequence x_i. In the binary case, we can calculate this x_i in a simple manner; however, in GF(q) cases, we should consider this step carefully.

Originally, the basic combination sequence means

$$1_k * x_i = 0 \tag{53}$$

where 1_k is a sequence $\left(\underbrace{111......1}_{k-ones}\right)$.

In the GF(q) case, there exist more possible sequences than those in the binary case. Thus, we need to find the number of cases that have *i* non-zero elements and that satisfy (53).

First, we find $b(k)$, the number of possible cases for which the sum of every *k*-non-zero element is zero.

$$b(k) = (q-1)^{k-1} - b(q-1) \tag{54}$$

We can realize this recurrence formula easily. When we obtain $b(k)$, we consider all possible k-1 elements and set the last element to satisfy (53). Then, we cannot find the last element only when the sum of *k-1* elements is equal to zero.

The generalized equation of (54) is as follows:

$$b(k) = \frac{(-1)^k (q-1) + (q-1)^k}{q} \tag{55}$$

Now, we can calculate x_i using $b(k)$.

$$x_i = \binom{k}{i} * b(i) \tag{56}$$

After obtaining x_i through (56), we can calculate the distance spectrum of non-binary LDPC codes with (48), (49), (50), and (51).

2.3.4 NUMERICAL RESULTS

This section illustrates our numerical results. Figure 7 shows the original distance spectrum of a regular LDPC code from Litsyn’s method and our proposed method. Because our method does not have any asymptotic bound, there exist some differences.

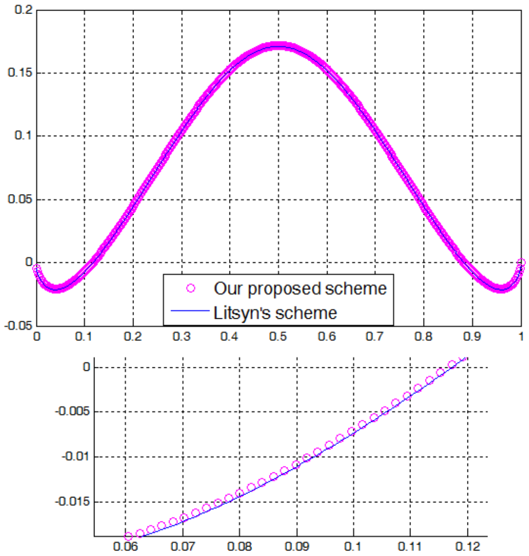


Figure 7. Distance spectrum of (n = 1024, j = 3, k = 4) regular LDPC code

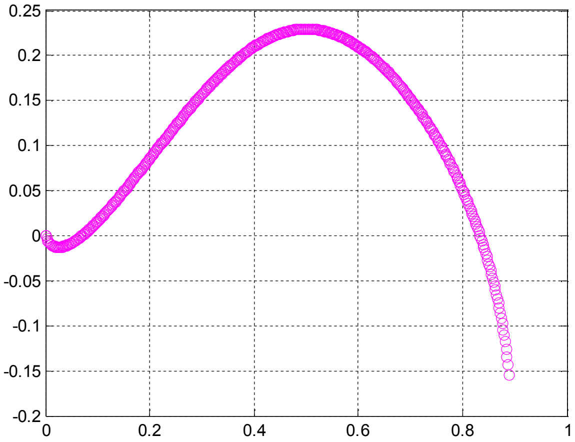


Figure 8. Distance spectrum of irregular LDPC codes with
$\left(n=1080,\lambda\left(x\right)=x^2,\rho\left(x\right)=1/2*x^3+1/2*x^4\right)$

Figure 8 shows the distance spectrum of irregular LDPC codes. In all distance spectrum graphs, the x-axis represents the normalized distance, which is calculated by (distance/signal_ length). This graph looks different from the graphs of the regular codes, and the minimum distance is located at approximately 0.08.

Figure 9 shows the distance spectrum of a non-binary LDPC matrix. We realize that the minimum distance does not increase monotonically with field size. Instead, there exists a certain field size that gives us the largest minimum distance. In our case, this occurs at $Q = 2^5$. This result is similar to Kasai's result [12].

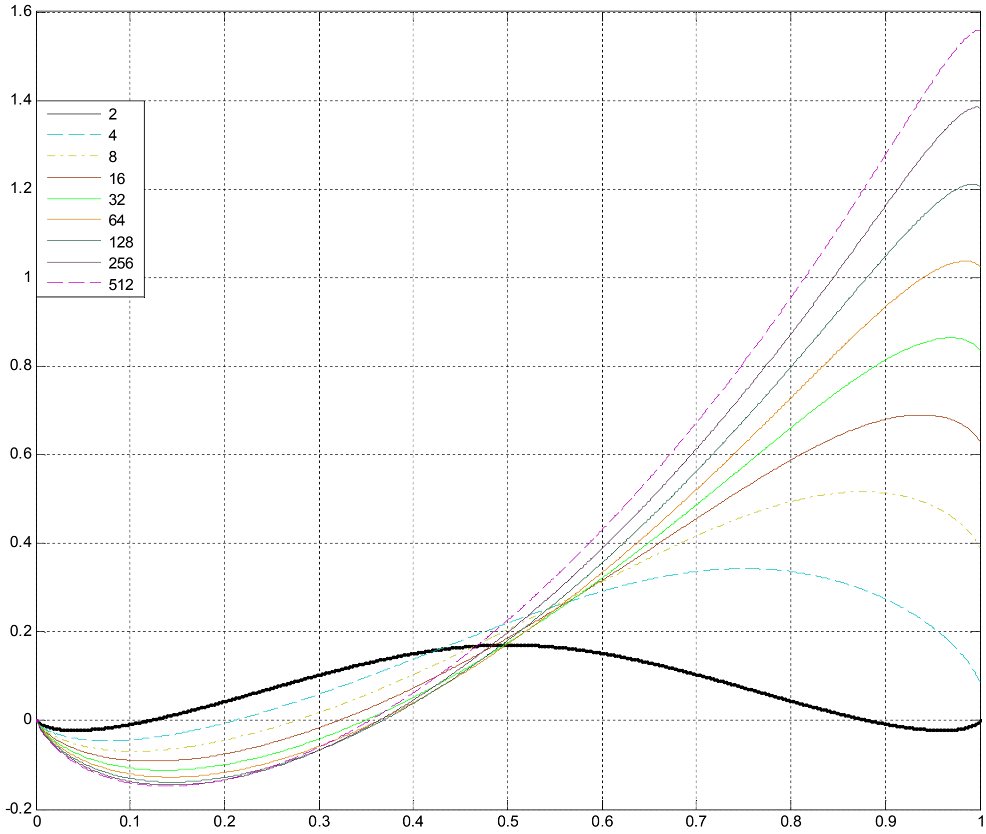


Figure 9. Distance spectrum of N = 1024, J = 3, K = 4 non-binary LDPC matrix

3 RIP OF AN LDPC MATRIX

3.1 RIP AND DISTANCE SPECTRUM

We aim to analyze the underdetermined problem of compressive sensing, using the approach of parity check systems research, particularly using the distance spectrum. Recently, the belief propagation algorithm, a decoding algorithm of LDPC codes, has attracted much attention as a compressive sensing recovery algorithm in finite fields [14][15]. With this algorithm, if we know the minimum distance of a sensing matrix, we can determine the spark value, because the spark is equal to the minimum distance [7].

3.2 RIP TREND DEPENDS ON DENSITY

Figure 10 and Figure 11 show the distance spectrum at different densities. As shown in these figures, as the density increases, the minimum distance also increases. However, the trend does not increase linearly, but has some limitations. When the length is 1200, the limitation is approximately 12% of the length, and when the length is 120, the limitation is approximately 10% of the length.

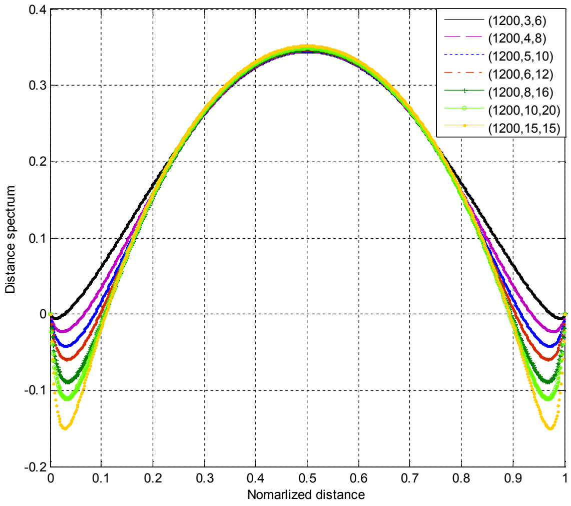


Figure 10. Distance spectrum of length 1200 binary code

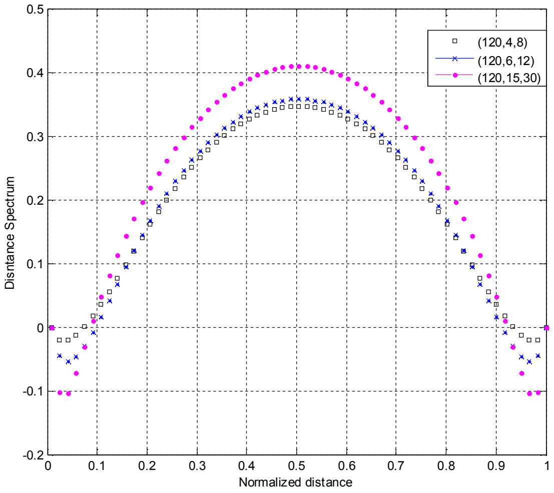


Figure 11. Distance spectrum of length 120 binary code

3.3 RIP TREND DEPENDS ON GF SIZE

Figure 12 shows the distance spectrum with different GF size.

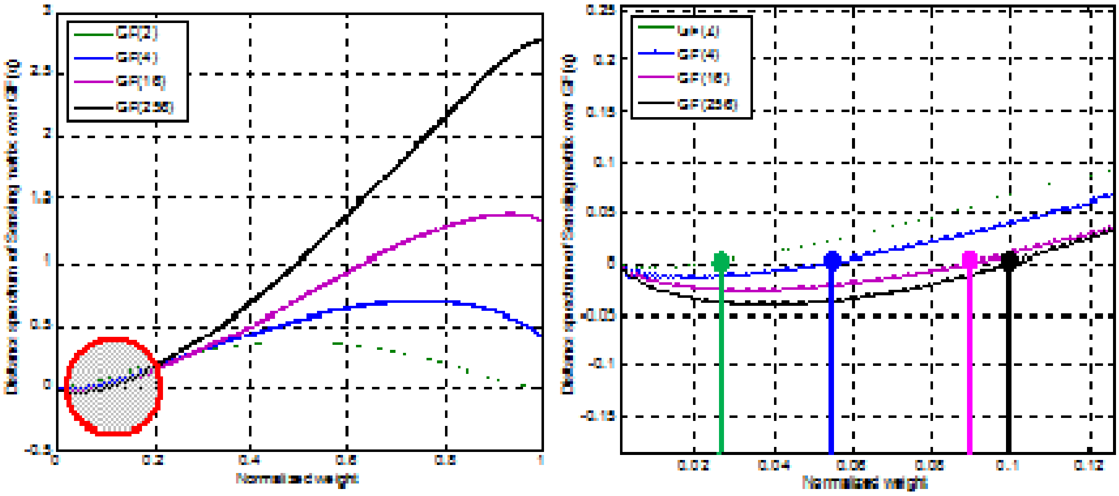


Figure 12. Distance spectrum of non-binary LDPC codes

We observe that as the GF size increases, the spark value also increases. However, Figure 9 shows that when the GF size is greater than 2^5, the spark value decreases by a small amount. The peak GF size is also related to density.

4 COMPRESSIVE SENSING VIA OFDM SYSTEM

A performance comparison of the OFDM (orthogonal frequency division multiplexing) system and the CS (compressive sensing) technique is presented in this section. In this section, we design an OFDM system based on CS and compare the proposed OFDM system performance with the original OFDM system over an AWGN channel.

CS is a sparse signal reconstruction technique. When the signal is sparse enough, we can compress it into a smaller size by using a rectangular matrix. In such a case, we can reconstruct the original signal by using an L-1 minimization algorithm. For the recovery algorithm, we used the interior point method and the log-barrier method [7][16].

4.1 ORIGINAL OFDM SYSTEM

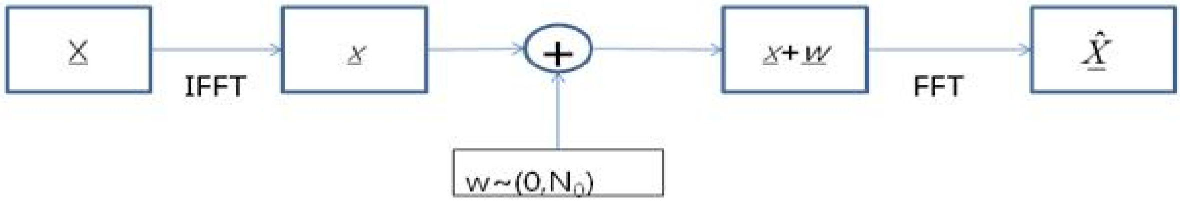


Figure 13. Original OFDM communication system diagram

Figure 13 shows the block diagram of the original OFDM system. The time domain transmitted signal $\underline{x}$ is generated by multiplying $\underline{X}$ with the IFFT matrix. For a signal length *N*, the IFFT equation and matrix can be represented as follows in equations (57) and (58), respectively.

$$\underline{x}[n]=\frac{1}{N}\sum_{k=0}^{N-1}\underline{X}_k e^{j(2\pi/N)kn} \tag{57}$$

$$A_{ifft\ [N\times N]}=\frac{1}{N}\times\begin{bmatrix}1 & 1 & 1 & \cdots & 1\\ 1 & e^{j\left(\frac{2\pi}{N}\right)} & e^{j\left(\frac{2\pi*2}{N}\right)} & \cdots & e^{j\left(\frac{2\pi*(N-1)}{N}\right)}\\ 1 & e^{j\left(\frac{2\pi*2}{N}\right)} & e^{j\left(\frac{2\pi*2*2}{N}\right)} & \cdots & e^{j\left(\frac{2\pi*2*(N-1)}{N}\right)}\\ \vdots & & \cdots & \cdots & \\ \vdots & & \cdots & \cdots & \\ 1 & e^{j\left(\frac{2\pi*(N-1)}{N}\right)} & & \cdots & e^{j\left(\frac{2\pi*(N-1)*(N-1)}{N}\right)}\end{bmatrix} \tag{58}$$

Gaussian noise $\underline{w}$ is added through the channel. The receiver reconstructs the original signal by multiplying the received signal with the FFT matrix. The original signal can be reconstructed via FFT, as in equation (60).

$$\underline{X}_k=\sum_{n=0}^{N-1}\underline{x}\left[n\right]e^{-j(2\pi/N)kn} \tag{59}$$

4.2 PROPOSED CS COMMUNICATION SYSTEM

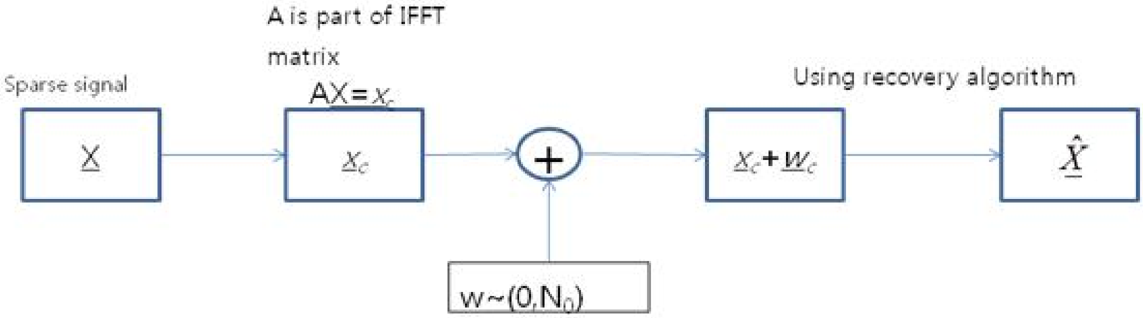


Figure 14. CS communication system diagram

Figure 14 shows the proposed CS communication system. Because the CS method works only with a sparse signal, we generate a *k*-sparse signal (*k* <<*N, sparsity*) and compress it using an IFFT matrix. The original IFFT matrix is square (size *N* × *N*); however, if we select some rows in the IFFT matrix, we can create a rectangular measurement matrix. The time domain signal ($\underline{x}_c$) is created by multiplying the sparse signal ($\underline{X}_C$) with the measurement matrix (A). Gaussian noise $\underline{w}$ is also added through the channel. In such a system, the difference in the reconstruction method is that we recover the signal using not an FFT matrix, but an L-1 recovery algorithm.

4.3 SIMULATION SETTINGS

- Original OFDM system settings

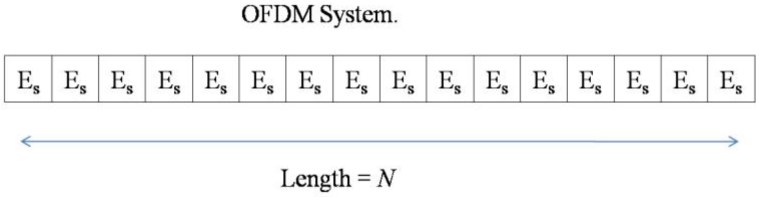


Figure 15. OFDM system energy distribution

To compare two systems (OFDM and CS), we should first set the energy per bit $\left(E_b\right)$. If we use the same symbol energy (E_s) and the modulation size is Q, the OFDM system's energy per bit is as shown in (60).

$$E_b = \frac{N * E_s}{N * \log_2 Q} = \frac{E_s}{\log_2 Q} \tag{60}$$

- Proposed CS system

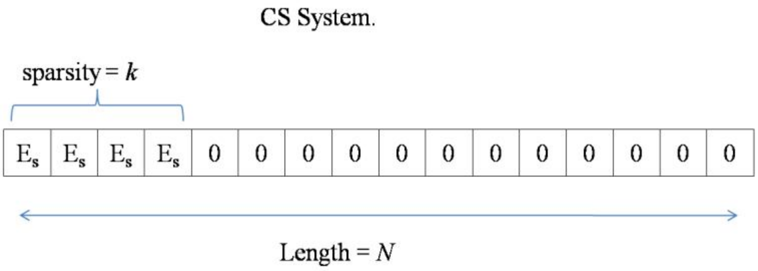


Figure 16. CS system energy distribution

The CS system has a different energy distribution. The total energy of a k-sparse signal is $k * E_s$, and the amount of information in a length *n, k*-sparse signal is (61).

$$\log_2 \left\{ \binom{N}{k} * (Q)^k \right\} = k * \log_2 Q + \log_2 \binom{N}{k} \tag{61}$$

The CS system's energy per bit can be calculated by dividing (60) by (61), as seen in (62):

$$E_b = \frac{k * E_s}{k * \log_2 Q + \log_2 \binom{N}{k}} \tag{62}$$

We set the sparse signal length to N = 128, the compressed signal size to $M = 2/N$ = 64, and the sparsity to k = 7. In addition, we use BPSK and QPSK modulation.

We use the RVD (real value decomposition) method for reconstructing complex numbers, because the IFFT matrix is composed of complex numbers. Eq. (63) shows the original linear problem with complex numbers:

$$AX_C = x_c \tag{63}$$

When we consider RVD, we reproduce (63) as (64):

$$\begin{bmatrix} \mathrm{Re}(A) & -\mathrm{Im}(A) \\ \mathrm{Im}(A) & \mathrm{Re}(A) \end{bmatrix} \begin{bmatrix} \mathrm{Re}(X_C) \\ \mathrm{Im}(X_C) \end{bmatrix} = \begin{bmatrix} \mathrm{Re}(x_c) \\ \mathrm{Im}(x_c) \end{bmatrix} \tag{64}$$

4.4 SIMULATION RESULTS

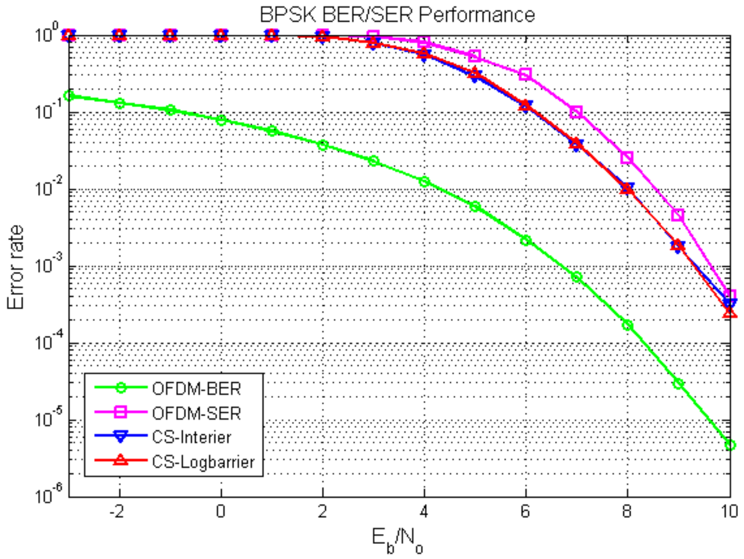


Figure 17. BPSK simulation results: N = 128, M = 64, K = 7

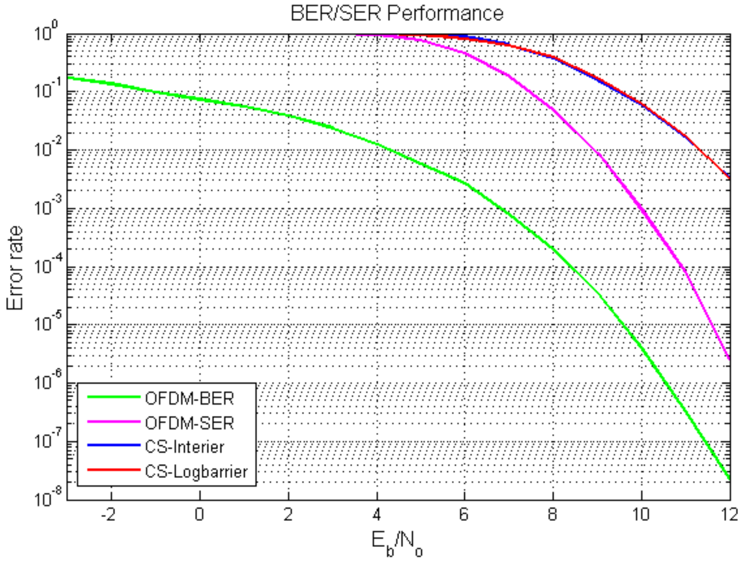


Figure 18. QPSK simulation results: N = 128, M = 64, K = 7

Figure 17 and Figure 18 show the simulation results of the BPSK and QPSK systems, respectively. In the graph, we observe that the OFDM system is better than the CS system, but within some region of the BPSK simulation, the CS system shows better performance than the OFDM SER curve. The interior point method and log-barrier method have almost the same result.

4.5 PEAK-TO-AVERAGE POWER RATIO

In communication systems, the PAPR (peak-to-average power ratio) is an important measurement [17].

The PAPR of a signal is given by

$$PAPR \simeq \frac{\max_n \left|x[n]\right|^2}{E_n\left[\left|x[n]\right|^2\right]} \tag{64}$$

Here is what Simon Litsyn wrote about the PAPR of multicarrier systems:

> "The main disadvantage of multicarrier modulation is that it exhibits a high PAPR. Namely, the peak values of some of the transmitted signals could be much larger than the typical values. This could lead to a necessity of using circuits with linear characteristics within a large dynamic range; otherwise the signal clipping at high levels would yield a distortion of the transmitted signal and out of band radiation. "

Many schemes have been proposed to reduce the PAPR. Clipping and filtering is one of these schemes [18]. TR (tone reservation) [19], active constellation extension [20], and SLM (selected mapping) techniques [21] are used to reduce the PAPR.

In this section, we compare the PAPR reduction performance of our proposed method with the well-known SLM scheme.

4.5.1 Computational Complexity

Computational complexity is evaluated in terms of multiplication and addition required for PAPR reduction. In an SLM scheme, U IFFT modules are used. Here, U is an independent phase sequence. In a TR scheme, additional multiplication of the signal length and additional addition of twice the signal length are required for PAPR reduction [22]. However, our proposed method does not need any additional computation, since there is no computation of the IFFT matrix.

Type of complex operation	TR scheme	SLM scheme	Proposed scheme
Multiplication	$(N/2)\log_2 N + N$	$(UN/2)\log_2 N$	$(N/2)\log_2 N$
Addition	$N\log_2 N + 2N$	$UN\log_2 N$	$N\log_2 N$

Table 1. Computational complexity of TR, SLM, and proposed schemes

4.5.2 Complementary cumulative distribution function of PAPR

The CCDF (complementary cumulative distribution function) of the PAPR is the probability that the PAPR exceeds threshold $PAPR_0$. In this section, we compare the CCDF of a conventional OFDM scheme, an SLD scheme, and our proposed scheme.

Generally, we consider the working range of the CCDF to be $10^{-2}-10^{-4}$. As shown in Figure 19, the CCDF of the PAPR of the OFDM system falls into this range for $PAPR_0$ above 12 dB for a raw OFDM system, because it does not use any techniques to reduce the PAPR. The SLM scheme falls into the working range at approximately 8–9 dB. Our proposed scheme falls into the working range at lower levels of $PAPR_0$ than the SLM scheme.

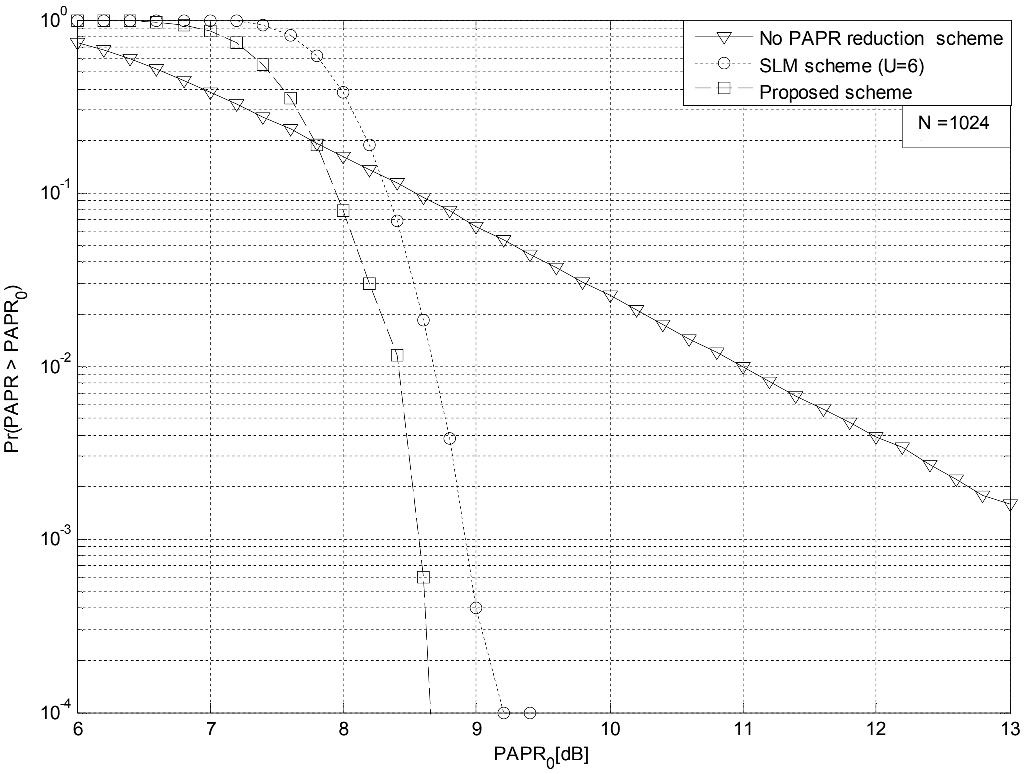


Figure 19. CCDF of the PAPR for raw OFDM scheme, SLM scheme, and proposed scheme when N = 1024

4.6 SYSTEM WITH LIMITED INPUT POWER AND RANDOM MAPPING

We have a new simulation based on the results that the proposed CS-OFDM scheme shows a better PAPR value than the original OFDM system. As the value of PAPR increases, the deviation of the input value increases. The maximum value of the input signal also increases. In this case, we need an outstanding amplifier. Thus, we assume that the model has such an amplifier.

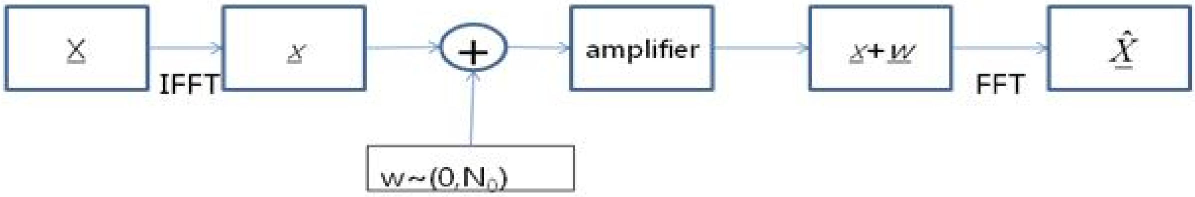


Figure 20. Original OFDM system block diagram with amplifier

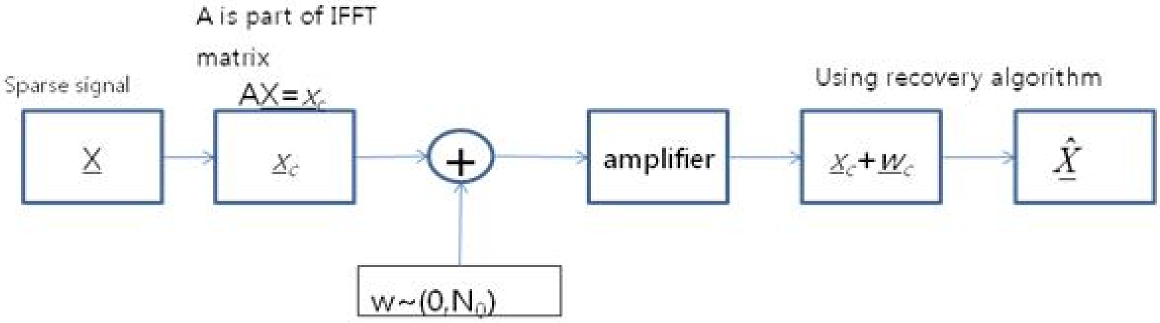


Figure 21. CS communication system diagram with amplifier

Figure 20 and Figure 21 show the block diagrams with amplifiers. When noise has a large effect on the signal, we need to increase the signal energy to obtain a high SNR. However, in that case, there can be some levels of energy beyond the limits of the input level of the amplifier. We consider the proposed CS communication scheme to work better than the original OFDM because the PAPR of the CS communication scheme is smaller than that of the original scheme.

We only calculate SER (not BER) in the proposed CS communication methods, because we do not have any algorithm that maps onto a sparse signal. However, we can use a random mapping algorithm to obtain the BER.

4.7 RANDOM MAPPING TECHNIQUE

The original mapping method creates the transmission message $\underline{x}$ (code word) by multiplying generator matrix (G) with the original message ($\underline{m}$).

$$\underline{x} = \underline{m}G \tag{65}$$

This method is difficult to adjust to our proposed system, because we do not have a sparse mapping algorithm. Thus, we use a reverse mapping system, like a parity check matrix. A possible BPSK K-sparse signal is as follows:

$$\# of\ possible combination = \binom{N}{K} * (2)^{K} \tag{66}$$

The number of binary bits to represent this many combinations is

$$M_{bits} = \left\lceil \log_2 \binom{N}{K} * (2)^{K} \right\rceil = \left\lceil K + \log_2 \binom{N}{K} \right\rceil \tag{67}$$

We create a random $N \times M_{bits}$ binary matrix (H) and multiply it with the K-sparse signal to create the message signal.

$$m = H \times \underline{X}_k \tag{68}$$

The H matrix should have a spark value greater than K to distinguish each of the message bits.

4.8 SIMULATION SETTINGS WITH LIMITED INPUT POWER AND RANDOM MAPPING

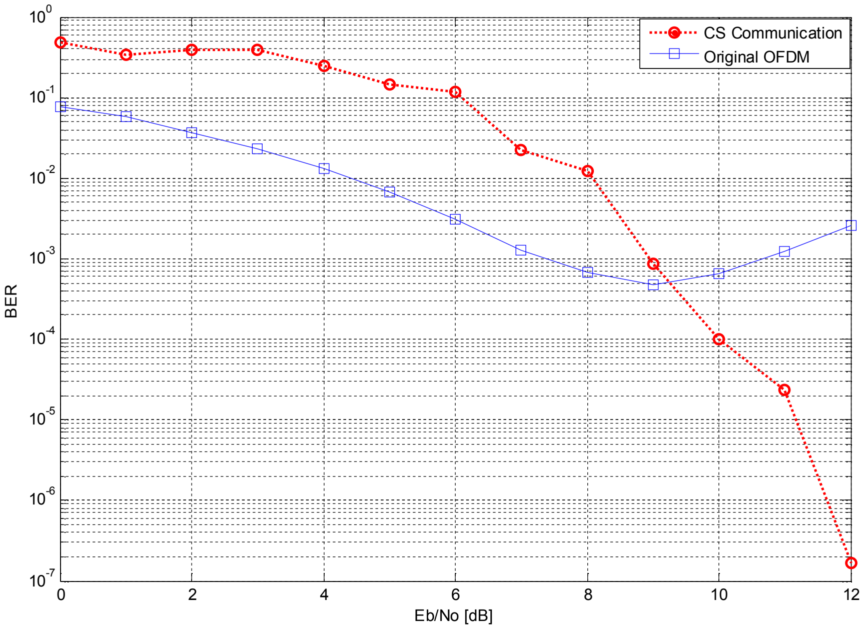


Figure 22. Simulation results (BPSK, N = 128, M = 64, K = 7, noise variance = 1, threshold of amplifier = 3)

Figure 22 shows the simulation results with random mapping and a limited amplifier. As shown in the graph, we observe that in the region above 10 dB, the proposed method shows better performance than the original method. The reason why the proposed method has a lower PAPR value is that there is little signal energy being lost in the amplifier.

5 Conclusion and Future Work

We discussed the fundamental limits of parity check codes, particularly LDPC codes, and techniques that use a parity check system and compressive sensing. Regarding distance spectra, the distance spectrum of the proposed method shows more accuracy than the existing method and can be adjusted to short codes. The remaining research problem is that we need to obtain the distance spectra of totally irregular codes (in this thesis, we only calculated the distance spectrum for check irregular LDPC codes).

Next, we analyzed the RIP through the distance spectrum. We found that the spark value increases with density. However, there are certain limitations. Likewise, the spark value increases with the GF size; however, in this case, there exists a peak point at 2^5.

Next, we proposed compressive sensing via an OFDM system. Overall, the OFDM system shows better performance than the CS system. For future work, we will find an optimum threshold for the CS reconstructed signal. Because the transmitted signal is sparse, we cannot use conventional methods (i.e., value > 0 = 1, value < 0 = -1). Instead, we use a threshold value (BPSK case: 0.5·Es, QPSK case: 0.35·Es); however, we cannot be certain that this is the optimum threshold value. Additionally, when we use the RVD method, a BPSK system has more sparsity (because the signal is real, K is not changed, but signal length doubles), and in the MPSK case, they have a symmetric location.

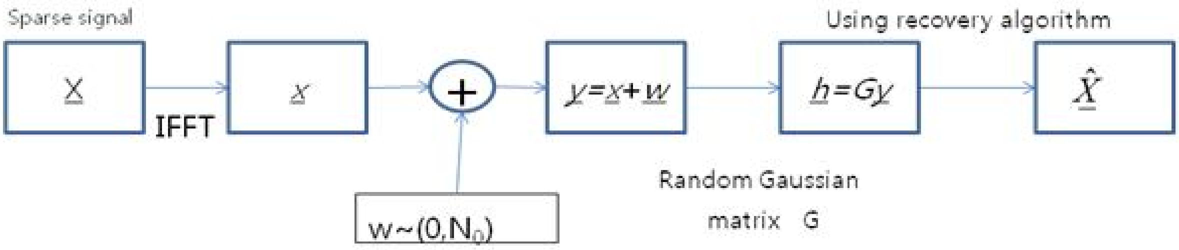


Figure 23. Another CS communication method

This topic is a worthwhile area for further research. In this thesis, we send M-length data, using part of the IFFT matrix. There is another scheme that uses a random Gaussian method. Figure 23 shows that type of system. This system will be a new research problem.

APPENDIX A

Let $C_{\omega}(k)$ be the number of combinations such that the sum of k non-zero numbers from GF(q) is equal to $\omega \in \mathrm{GF}(q)$. Then,

$$c_{\omega}(k)=\begin{cases}\dfrac{(-1)^{k}(q-1)+(q-1)^{k}}{q}, & \omega=0\\ (-1)^{k-1}\left(\dfrac{1-(1-q)^{k}}{q}\right), & \omega\neq 0\end{cases}$$

Let $C_{\omega}(k)$ be the number of combinations such that k non-zero numbers from GF(q) are equal to 1.

We conjecture that $C_{\omega}(k)=(q-1)^{k-1}-C_{w}(k-1)$. The rationale is that there are $(q-1)^{k-1}$ combinations of k - 1 non-zero values in GF(q). From these sequences of length k - 1, we remove all sequences whose sum is already 1; there are $C_{\omega}(k-1)$ of them. We need to remove them because of the following case:

$$(x_1+\cdots+x_{k-1})+x_k=1 \tag{69}$$

If $(x_1+\cdots+x_{k-1})=1$, then we have

$$1+x_k=1 \tag{70}$$

leading to $x_k=0$, which contradicts the given condition. Thus, it is necessary to remove all of these sequences.

However, we need to check if this is also a sufficient condition. If $(x_1+\cdots+x_{k-1})\neq 1$, then it is 0, 2, 3, …, q-1.

Let us check that the sum is zero. Then, (70) takes the following form:

$$0+x_k=1 \tag{71}$$

Then, we notice that we can satisfy the equality by choosing $x_k=1$. Thus, all those combinations for which $\left(x_1+\cdots+x_{k-1}\right)=0$ should remain in the pool.

Let us now check when the sum is 2. Then, (70) takes the following form:

$$2+x_k=1 \tag{72}$$

which leads to

$$x_k=1+2\neq 0\,. \tag{73}$$

This holds for the rest of other numbers, i.e., 3, 4, …, q-1. Thus, this is a sufficient condition as well.

We solve this recurrence equation by induction

$$\begin{aligned} C_w(k+1)&=(q-1)^k-C_w(k)\\ C_w(k+1)+C_w(k)&=(q-1)^k \end{aligned}$$

$$\begin{aligned} C_w(k)&=(q-1)^{k-1}-C_w(k)\\ C_w(k)+C_w(k-1)&=(q-1)^{k-1} \end{aligned}$$

$$\begin{aligned} C_w(1)+C_w(0)&=(q-1)^1\\ C_w(2)+C_w(1)&=(q-1)^2\\ C_w(3)+C_w(2)&=(q-1)^3\\ &\vdots\\ &\vdots\\ C_w(k)+C_w(k-1)&=(q-1)^k \end{aligned}$$

Let us set k to zero. Then

$$\begin{aligned} C_w(1)+C_w(0)&=(q-1)^1\\ -\left(C_w(2)+C_w(1)\right)&=(-1)^{2-1}(q-1)^2\\ C_w(3)+C_w(2)&=(q-1)^3\\ &\vdots\\ &\vdots\\ (-1)^{k-1}\left(C_w(k)+C_w(k-1)\right)&=(-1)^{k-1}(q-1)^k \end{aligned} \tag{74}$$

The summation of all the equations in (74) is

$$
\begin{aligned}
C_w(0)+(-1)^{k-1}C_w(k) &= \sum_{i=0}^{k-1}(-1)^i(q-1)^i \\
&= \sum_{i=0}^{k-1}(1-q)^i \\
&= \left(\frac{1-(1-q)^k}{1-(1-q)}\right)=\left(\frac{1-(1-q)^k}{q}\right)
\end{aligned}
\tag{75}
$$

A rearrangement of (75) is

$$
C_w(k)=(-1)^{k-1}\left(\frac{1-(1-q)^k}{q}\right)-(-1)^{k-1}C_w(0) \tag{76}
$$

The initial value is different according to whether w is zero or not. When $w=0$, then $C_0(0)=1$. Otherwise, $C_{w(\neq 0)}(0)=0$. Thus, when $w=0$,

$$
C_0(k)=\frac{(-1)(q-1)+(q-1)^k}{q}.
$$

When $w\neq 0$,

$$
C_{w(\neq 0)}=(-1)^{k-1}\left(\frac{1-(1-q)^k}{q}\right)
$$

REFERENCES

1. Todd K. Moon, “Error Correction Coding”, John Wiley & Sons, 2005.

2. D. J. C. MacKay, “Good Error-Correcting Codes Based on Very Sparse Matrices”, IEEE Trans. on Information Theory, vol. 45, no. 2, pp. 399-431, March 1999.

3. Robert G. Gallager, Low-Denisty Parity-Check Codes, MIT Press, 1963.

4. Richard Baraniuk, “Lecture Notes: Compressive Sensing”, IEEE Signal Processing Magazine, p. 118-121, July 2007.

5. Justin Romberg, “Imaging via compressive sampling”, IEEE Signal Processing Magazine, 25(2), pp. 14-20, March 2008.

6. David L. Donoho, “Compressed Sensing”, IEEE Trans. Information Theory, vol. 52, no. 4, pp. 1289-1306, Apr. 2006.

7. Heung-No Lee, “ Lecture Note for Compressive Sensing”, Spring Semester, 2011.

8. S. Litsyn and V. Shevelev, “On ensembles of low-density parity-check codes: Asymptotic distance distributions”, IEEE Trans. Inform. Theory, vol. 48, pp. 887–908, Apr. 2002.

9. D. Burshtein and G. Miller, “Asymptotic enumeration method for analyzing LDPC codes”, IEEE Trans. Inform. Theory, submitted for publication.

10. R. M. Tanner, “A recursive approach to low complexity codes”, IEEE Transactions on Information Theory, vol. IT-27, no. 5, pp. 533–547, Sept. 1981.

11. K. Kasai, C. Poulliat, D. Declercq, T. Shibuya, and K. Sakaniwa, “Weight distribution of non-binary LDPC codes”, Proc. 2008 Int. Symp. Inf. Theory Appl. (ISITA’08), Auckland, New Zealand, Dec. 2008, pp. 748–753.

12. Heung-No Lee, “Distance Spectrum of LDPC and LDGM Codes”, Technical Report (Communications Research Lab at the University of Pittsburgh), July 2006.

13. D. Baron, S. Sarvotham, and R. G. Baraniuk, “Bayesian Sensing via Belief Propagation”, IEEE Sig. Proc., vol. 58, no. 1, pp. 269-280, Jan. 2010.

14. Stark C. Draper and Sheida Malekpour, “Compressed Sensing over Finite Fields”, ISIT, Seoul,

Korea, 2009.

15. David L. Donoho, "Compressed Sensing", IEEE Trans. Information Theory, vol. 52, no. 4, pp. 1289-1306, Apr. 2006.

16. Y.Wu and W. Y. Zou, "Orthogonal frequency division multiplexing: A multi-carrier modulation scheme", IEEE Trans. Consumer Electronics,vol. 41, no. 3, pp. 392–399, Aug. 1995.

17. R. O'Neil and L. B. Lopes, "Envelope Variations and Spectral Splatter in Clipped Multicarrier Signals", Proc IEEE PIMRC 95, Toronto, Canada, pp. 71-75, Sept.1995.

18. S. H. Han and J. H. Lee, "An Overview of Peak – to – Average Power Ratio Reduction Techniques for Multicarrier Transmission", IEEE Transaction on Wireless Communication, April 2005.

19. S. H. Muller, J. B. Huber, "A Comparison of Peak Power Reduction Schemes for OFDM", Proc. IEEE GLOBECOM '97, Phoenix, AZ, November 1997.

20. R. W. Baumi, R. F. H. Fisher, and J. B. Huber, "Reducing the Peak – to – Average Power Ratio of Multicarrier Modulation by Selected Mapping", Elect. Lett., Vol. 32, No. 22, October 1996.

21. Eonpyo Hong, Youngin Park, Sangchae Lim and Dongsoo Har, "Adaptive Phase Rotation of OFDM Signals for PAPR Reduction," IEEE Transactions on Consumer Electronics, Vol 57, pp. 78-82, Dec. 2011.

Acknowledgements

보잘것 없는 저를 인도해주시고 사랑해주신 하나님께 감사합니다. 짧지도 길지도 않은 2 년이라는 시간동안 감사하고 즐거운일들이 참 많았던것 같습니다. 많은 관심과 사랑으로 저를 이끌어주신 이흥노 교수님께 감사드립니다. 그리고 함께 생활한 INFONET 연구실 가족들 박순철교수님, 진택이형, 환철이형, 웅비형, 영학이형 상준이선배, 형원이, 승찬이형, 준호형, 재건이형, 정민이, 성진누나, 수정누나, Oliver, Asad, Raze, Le Wang, LinLin, Subhro, Aparna, Lan, Nitin, Ashish 같은연구실에서 함께 생활하고 호흡했던 시간들이 너무나 소중한 추억이 될것 같습니다. 또 친동생처럼 돌봐주신 재욱이형, 형민이형, 윤동이형 너무나 감사드립니다. 승원이형, 보람이, 은실누나 만나서 너무 반가웠습니다 더 일찍 친해졌으면 하는 아쉬움이 남네요. 광주라는 낯선땅에서 알게된 서림교회 지체들 너무너무 감사합니다. 또한 언제나 밝게 맞아주고 관심가져준 신호등교회 지체분들 너무나 사랑합니다. 미국땅에서 응원해준 최스릭과 시화땅에서 응원해준 영일이 그리고 무엇보다 항상 큰 힘이 되어준 가족들, 초베 성민이에게 감사와 사랑을 전합니다. 끝으로 사랑하는 정화와 이기쁨을 함께하고 싶습니다.