Ph.D. DISSERTATION

New Automorphism Group Decoder for Erasure Channel and Constructions of LRCs and Generalized RP LDPC Codes

새로운 소실 채널을 위한 자기동형 군 복호기 및 부분 접속 복구 부호 및 일반화된 근 프로토그래프 LDPC 부호의 설계

BY

KIM CHANKI

FEBRUARY 2019

DEPARTMENT OF ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

Ph.D. DISSERTATION

New Automorphism Group Decoder for Erasure Channel and Constructions of LRCs and Generalized RP LDPC Codes

새로운 소실 채널을 위한 자기동형 군 복호기 및 부분 접속 복구 부호 및 일반화된 근 프로토그래프 LDPC 부호의 설계

BY

KIM CHANKI

FEBRUARY 2019

DEPARTMENT OF ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

New Automorphism Group Decoder for Erasure Channel and Constructions of LRCs and Generalized RP LDPC Codes

새로운 소실 채널을 위한 자기동형 군 복호기 및 부분 접속 복구 부호 및 일반화된 근 프로토그래프 LDPC 부호의 설계

지도교수 노 종 선

이 논문을 공학박사 학위논문으로 제출함

2019년 2월

서울대학교 대학원

전기 컴퓨터 공학부

김 찬 기

김찬기의 공학박사 학위 논문을 인준함

2019년 2월

위 원 장: _____
부위원장: _____
위 원: _____
위 원: _____
위 원: _____

Abstract

In this dissertation, three main contributions are given as; i) new two-stage automorphism group decoders (AGD) for cyclic codes in the erasure channel, ii) new constructions of binary and ternary locally repairable codes (LRCs) using cyclic codes and existing LRCs, and iii) new constructions of high-rate generalized root protograph (GRP) low-density parity-check (LDPC) codes for a nonergodic block interference and partially regular (PR) LDPC codes for follower noise jamming (FNJ), are considered.

First, I propose a new two-stage AGD (TS-AGD) for cyclic codes in the erasure channel. Recently, error correcting codes in the erasure channel have drawn great attention for various applications such as distributed storage systems and wireless sensor networks, but many of their decoding algorithms are not practical because they have higher decoding complexity and longer delay. Thus, the AGD for cyclic codes in the erasure channel was introduced, which has good erasure decoding performance with low decoding complexity. In this research, I propose new TS-AGDs for cyclic codes in the erasure channel by modifying the parity check matrix and introducing the preprocessing stage to the AGD scheme. The proposed TS-AGD is analyzed for the perfect codes, BCH codes, and maximum distance separable (MDS) codes. Through numerical analysis, it is shown that the proposed decoding algorithm has good erasure decoding performance with lower decoding complexity than the conventional AGD. For some cyclic codes, it is shown that the proposed TS-AGD achieves the perfect decoding in the erasure channel, that is, the same decoding performance as the maximum likelihood (ML) decoder. For MDS codes, TS-AGDs with the expanded parity check matrix and the submatrix inversion are also proposed and analyzed.

Second, I propose new constructions of binary and ternary LRCs using cyclic codes and existing two LRCs for distributed storage system. For a primitive work, new constructions of binary and ternary LRCs using cyclic codes and their concatena-

tion are proposed. Some of proposed binary LRCs with Hamming weights 4, 5, and 6 are optimal in terms of the upper bounds. In addition, the similar method of the binary case is applied to construct the ternary LRCs with good parameters. Also, new constructions of binary LRCs with large Hamming distance and disjoint repair groups are proposed. The proposed binary linear LRCs constructed by using existing binary LRCs are optimal or near-optimal in terms of the bound with disjoint repair group.

Last, I propose new constructions of high-rate GRP LDPC codes for a nonergodic block interference and anti-jamming PR LDPC codes for follower jamming. The proposed high-rate GRP LDPC codes are based on nonergodic two-state binary symmetric channel with block interference and Nakagami-m block fading. In these channel environments, GRP LDPC codes have good performance approaching to the theoretical limit in the channel with one block interference, where their performance is shown by the channel threshold or the channel outage probability. In the proposed design, I find base matrices using the protograph extrinsic information transfer (PEXIT) algorithm. Also, the proposed new constructions of anti-jamming partially regular LDPC codes is based on follower jamming on the frequency-hopped spread spectrum (FHSS). For a channel environment, I suppose follower jamming with random dwell time and Rayleigh block fading environment with M-ary frequnecy shift keying (MFSK) modulation. For a coding perspective, an anti-jamming LDPC codes against follower jamming are introduced. In order to optimize the jamming environment, the partially regular structure and corresponding density evolution schemes are used. A series of simulations show that the proposed codes outperforms the 802.16e standard in the presence of follower noise jamming.

keywords: Automorphism group decoder (AGD), Bose-Chaudhuri-Hocquenghem (BCH) codes, block fading (BF), block interference (BI), cyclic codes, distributed storage systems (DSSs), error correcting codes, erasure channel, frequency hopping spread spectrum (FHSS), follower noise jamming (FNJ), iterative erasure decoder (IED), low-density parity-check (LDPC) codes, locally repairable codes (LRCs), maximum distance separable (MDS) codes, military communication, perfect codes, protograph LDPC codes, protograph extrinsic information transfer (PEXIT), root protograph (RP) LDPC codes, stopping redundancy.

student number: 2013-20778

Contents

List of Tables

List of Figures

Chapter 1

INTRODUCTION

1.1 Background

Research on error correcting codes in the erasure channel is one of the major subjects in information and communication theory. Erasure channel is a typical channel model for wireless sensor networks and distributed storage systems (DSSs), where the locations of symbol errors are known. In addition, some error channels such as those with block fading (BF), block interference (BI), and jamming are often considered as erasure channel because these approaches are advantageous for analysis and optimization of coding schemes.

Subjects on the research of coding for these channels can be divided into two parts; decoder implementation and code construction. For a decoder implementation, it is known that codes for erasure channel have lower decoding complexity than error channel. For low-density parity-check (LDPC) codes, low-complex belief propagation (BP) decoder achieves the performance of maximum likelihood (ML) decoders for sufficiently large codelength in the optimized constructions using density evolution (DE) and avoiding stopping sets. However, existing decoders for classical algebraic codes are generally highly complex and their performances are inferior to the ML decoders.

Algebraic codes have a long history from Hamming codes to algebraic geometry codes. The decoders of algebraic codes are designed using the mathematical properties of the codes and thus it is difficult to implement practical decoders for algebraic codes. However, lots of research works for their decoding algorithms have been done to reduce the decoding complexity and delay. In cyclic codes, one-step majority decoding [1] and permutation decoding [2] schemes are exemplary methods which can be practically implemented using their cyclic property in the error channel. A low complexity iterative decoder can be one of the solution as an implementable decoder and thus, the iterative decoding algorithms and error correcting codes with iterative decoder such as turbo and LDPC codes have been widely studied. In addition, there have been lots of researches to apply iterative decoding to algebraic codes in error channels. In [3] and [4], the iterative decoding of Reed-Solomon (RS) codes with sparse parity check matrix and belief-propagation decoding algorithm is proposed. Iterative erasure decoder (IED) of algebraic codes [5] has also been studied. However, IED has inherently inferior decoding performance compared to the ML decoder and the gap between the decoding performances becomes larger for the algebraic codes, because the sparseness of their parity check matrices is not guaranteed contrary to the LDPC codes. Thus a possible solution for decoding of algebraic codes is to modify the structure of the decoder in the erasure channel.

Recently, one approach to overcome the inferior decoding performance of IED for the algebraic codes in the erasure channel was proposed, called the automorphism group decoder (AGD) for cyclic codes [6]. AGD uses the permutations of the automorphism group in the middle of the IED procedure. For cyclic codes, the permutation operation can be substituted by the cyclic shift operation for codewords, which are also codewords. In fact, many similar concepts have been proposed for cyclic LDPC codes in the error channel such as multiple-bases belief-propagation (MBBP) [7] and revolving iterative decoding (RID) [8], [9]. It was shown that for some cyclic codes, AGD improves the decoding performance but it requires higher decoding complexity

and delay due to repeated decoding process.

In order to operate AGD efficiently, it is important to design the appropriate parity check matrix. However, the conventional design method in [6] includes the problem to find codewords with minimum Hamming weight known as NP-hard problem in general. In contrast, the proposed TS-AGD includes a construction algorithm of good parity check matrix with polynomial-time complexity and also has excellent decoding performance with low decoding complexity for cyclic codes.

The proposed decoding process is done in two decoding stages, referred to as a two-stage AGD (TS-AGD), that is, the first decoding stage finds the cyclic shift values of the received vector for the successful erasure decoding while in the second decoding stage, the erasure decoding process is done for the received vectors cyclically shifted by the cyclic shift values found in the first decoding stage. Further, the proposed TS-AGD algorithm can be implemented by the modified parity check matrix for the (n, k) cyclic code such that some of the $(n - k)$-tuple column vectors in the parity check matrix are standard vectors in the appropriate column indices and Hamming weight of the row vectors in the parity check matrix becomes as low as possible, which requires polynomial-time complexity. The numerical analysis shows that the proposed algorithms are advantageous for two classes of cyclic codes. For perfect codes, it is shown that the proposed modification of parity check matrix for TS-AGD achieves the perfect decoding, showing the decoding performance identical to that of the ML decoding. For the BCH codes with long codelength, TS-AGD and AGD with modified parity check matrix have the near-ML decoding performance or better than the regular LDPC codes with similar parameters. Further, the TS-AGD reduces decoding complexity compared to AGD.

Generally, each check equation in the IED has its own erasure decoding capability. For some algebraic codes, it is known that $n - k$ check equations are not sufficient to achieve good decoding performance. Thus, stopping redundancy was proposed [10], which increases the number of check equations and it guarantees the successful de-

coding for all the erasure symbols up to $d_{min} - 1$. However, stopping redundancy increases the decoding complexity. Stopping redundancy has been studied for several algebraic codes such as maximum distance separable (MDS) codes [11], Reed-Muller codes [12], and algebraic geometry codes [13].

MDS codes are the algebraic codes which satisfy the Singleton bound, that is, for (n, k, d) MDS codes as

$$d = n - k + 1. \tag{1.1}$$

It is known that MDS codes have the optimal decoding performance in the erasure channel. The RS code is a well-known class of cyclic MDS codes that has been widely applied to compact disc, satellite communication, and DSSs.

I propose another two-stage decoding scheme to decode cyclic MDS codes, by modifying the TS-AGD by stopping redundancy, where the proposed TS-AGD offers an explicit decoding method achieving decoding performance of ML decdoer. Further, several lower bounds on the stopping redundancy for the perfect decoding of cyclic MDS codes have been derived. Furthermore, the proposed TS-AGD with submatrix inversion of the parity check matrix is analyzed as a generalization of the previous method.

On the construction of the erasure codes, their intrinsic properties of the channels are considered in the design procedure, which makes definite difference from the codes optimized from the Gaussian error channels. In this research, new locally repairable codes (LRCs) for DSSs and LDPC codes for follower noise jamming (FNJ) and Block interference (BI) are proposed.

For a decade, coding for DSSs has attracted a considerable amount of attention from researchers as the demand for data centers based on DSS grows exponentially. In particular, regenerating codes and LRCs are mostly studied. LRC is designed to minimize the number of storage nodes to be accessed during the repair process, called the locality r.

For practical usefulness, attempts to construct the optimal binary LRCs have also

been made, some of which achieve the optimality [29], [32]. For example, some constructions of optimal binary LRCs using bipartite graph [39], partial spread [40], [41] and cyclic codes [29], [31] but their constructions are limited cases with some codelength and Hamming distance upto 6. Similarly, ternary LRCs were studied [33], where several constructions were proposed. Moreover, cyclic LRCs were introduced in [34].

In this work, new constructions of binary and ternary LRCs are proposed. In order to construct binary LRCs. new construction methods using cyclic codes and existing LRCs are proposed. For the binary case, it is shown that the proposed LRCs with $4 \leq d \leq 10$ and some r for large portion of n are shown to be optimal or near-optimal in terms of the upper bounds in [35] and [37]. The similar construction of binary case is applied to the ternary ones, where ternary LRCs with good parameters are constructed.

Channels with block interference (BI) have been researched in wireless communication systems after the channel models were proposed in [53], where interference occurs in the block unit, called a hop. If length of each hop is not small compared to codelength, BI cannot be averaged over other hops and nature of BI channel is said to be nonergodic.

Two-state binary symmetric channel with block interference (TS-BSC-BI) [54] is a BSC channel with two states in each hop according to existence of BI. It was adopted in the Gilbert-Elliot channel [55] and the channel with jamming in the military communication [56]. In this dissertation, I propose a new class of flexible high-rate LDPC codes, termed generalized root protograph (GRP) LDPC codes for a channel with one BI hop among L hops. For the proposed construction, the techniques of low-rate root [57] and root protograph (RP) LDPC codes [58] for block fading (BF) channel are used. Note that design methods of protograph LDPC codes for BF channel have also been researched recently in many literatures [59], [62], [63], [64].

Jamming and anti-jamming (AJ) schemes are considered as crucial issues in the electric warfare environment and the military communication systems. As an efficient anti-jamming schemes, information-theoretic approaches using channel modeling and

coding schemes were extensively researched. For a channel modeling, channels with jamming can be considered as a class of channels with BI but there also exists difference because jamming is intentionally designed to disrupt the communication link.

For one of AJ schemes, frequency hopping spread spectrum (FHSS) selects one of frequency band using pseudorandom sequence, which can make it hard to know the frequency hopping pattern and thus the system can obtain anti-jamming capability. Therefore, jammer attempts to send jamming signal in partial band randomly, called partial band jamming. Prior works to mitigate jamming use inter-hop interleaving, called bit interleaving coded modulation and iterative decoding (BICM-ID) [46] and RS concatenated coding [56] that can correct burst error caused by jamming. The prior techniques have high anti-jamming performance but they can increase computational complexity by decoding process.

However, there are more efficient jamming strategies one of which is a follower jamming. In follower jamming scenario, jammer scans the occupied frequency bands and send the jamming signal in the band found. To this end, jammer uses the frequency scanner called determinator [47] or communication electronic support measure (CESM) [48] that can guarantee certain level of the scanning probability. Slow frequency hopping (SFH) can be vulnerable to follower jamming environment. SFH is required to lengthen the hop period or decrease hopping speed, both of which are inevitable for high data rate communication.

In classes of the LDPC codes, partially regular LDPC (PR-LDPC) codes are the codes which have small irregularity of degree distribution and is designed for unequal error protection (UEP) [50]. PR-LDPC codes can also be used to AJ communication systems. Simplified erasure-based channel environment and the corresponding density evolution (DE) are proposed for construction of PR-LDPC codes.

1.2 Overview of Dissertation

This dissertation is organized as follows.

In Chapter 2, basic concepts of IED and AGD for erasure channel, LRCs for distributed storage system, and generalized RP and PR LDPC codes in the channels with block interference and jamming are discussed. In the first section, operations of IED and AGD are analyzed in the component of bipartite graph such as variable nodes, check nodes, and their edges. Also, conventional algorithm in AGD is introduced. In the next section, definition, bound, and existing optimal constructions of LRCs are discussed, especially for the binary case. Last, the channels with block interference and jamming and system models are introduced. Note that block fading such as Rayleigh and Nakagami-m fading can be considered together with BI and jamming. For a channel with BI, channel outage analysis using instantaneous input-output mutual information is proposed as both implicit and explict form.

In Chapter 3, new two-stage automorphism group decoders for cyclic codes in the erasure channel are proposed. In Section 3.1, some definitions used in Chapter 3 are introduced. In Section 3.2, the proposed decoding algorithm for the binary cyclic codes in the erasure channel is introduced by modifying the parity check matrix and the AGD algorithm, called TS-AGD. For perfect codes, the proposed TS-AGD achieves the perfect decoding in the erasure channel. For the BCH codes with long codelength, the proposed TS-AGD achieves the near-ML performance. The numerical analysis of the performance of the proposed decoding algorithm is used to verify the performance improvement. In Section 3.3, the proposed TS-AGD algorithms are modified for the cyclic MDS codes by using stopping redundancy and submatrix inversion. Then, several lower bounds on the stopping redundancy and submatrix inversion for the perfect decoding are derived.

In Chapter 4, new constructions of binary and ternary LRCs using cyclic codes and existing LRCs are proposed. In Section 4.1, it is shown that some of the proposed LRCs with Hamming distance larger than 4 are shown to be optimal in terms of the

upper bounds in [35] and [37]. In Section 4.2, the similar construction of binary case is applied to the ternary ones, where ternary LRCs with good parameters are constructed. In Section 4.3, new constructions of binary LRCs with disjoint repair group using existing LRCs are proposed. For new constructions, it is shown that some existing LRCs can also be made by the new construction. In Section 4.4, The proposed construction method is applied to construct optimal or near-optimal binary LRCs with large Hamming distance larger than or equal to 8.

In Chapter 5, new constructions of high-rate GRP LDPC codes for a nonergodic BI and PR LDPC codes for FNJ are proposed. In Section 5.1, a new design method of GRP LDPC codes is proposed. In Section 5.2, their asymptotic analysis is given and finite-length performances of the proposed LDPC codes are compared with the full-diversity LDPC codes and the channel outage probability by numerical analysis. In Section 5.3, follower noise jamming model is introduced. In Section 5.4, construction method of PR-LDPC codes for anti-jamming is proposed and numerical analysis is done. In the simulation, the same codelength as LDPC codes of the IEEE 802.16e standards are compared. The result shows that the proposed codes have superior performance than the standard for all the symbol sizes and jamming environments.

Finally, the concluding remarks are given in Chapter 6.

1.3 Notations

In this section, mathematical notations throughout the dissertation are summarized. All operations are based on the finite field $\mathbb{F}_q$, where q is the size of the field. Let $\mathbf{v}$ be a row-wise vector and v_i be the i-element of $\mathbf{v}$. Then, the support of $\mathbf{v}$ is denoted by $\text{supp}(\mathbf{v}) = \{i; v_i \neq 0\}$ and the Hamming weight of $\mathbf{v}$ by $\text{wt}_H(\mathbf{v}) = |\text{supp}(\mathbf{v})|$. Let $[a, b] = \{i \in \mathbb{N}; a \leq i \leq b\}$ and $[i] = [1, i]$ for the set of positive integers $\mathbb{Z}_+$. Let $\mathbf{1}$ and $\mathbf{0}$ be all-one and all-zero vectors, respectively.

Chapter 2

Preliminaries

2.1 IED and AGD for Erasure Channel

In this section, I introduce the decoding procedure of IED and AGD for erasure channel. First, I will introduce the concept of IED.

2.1.1 Iterative Erasure Decoder

An (n, k) error correcting code has an $(n-k) \times n$ parity check matrix H, which can be represented by a bipartite graph $\mathcal{G}$ with n variable nodes and $n-k$ check nodes. Let V and U be sets of variable nodes and check nodes and let d_{v_i} and d_{u_j} be degrees of a variable node $v_i \in V$ and check node $u_j \in U$, respectively, for $0 \leq i \leq n-1$ and $0 \leq j \leq n-k-1$. The bipartite graph is then denoted by $\mathcal{G}$=(V, U, H). In the erasure channel, the variable nodes have two different states, i.e., erasure and non-erasure states, while the check nodes have three states, i.e., decodable, non-decodable, and non-erasure states. The decoding procedure of IED consists of several iterations, where each iteration performs check node update (CNU) and variable node update (VNU) operations sequentially. It is assumed that CNU and VNU in the decoding procedure are operated in a parallel way, known as flooding decoding.

The CNU operation is the procedure that each check node finds its state by count-

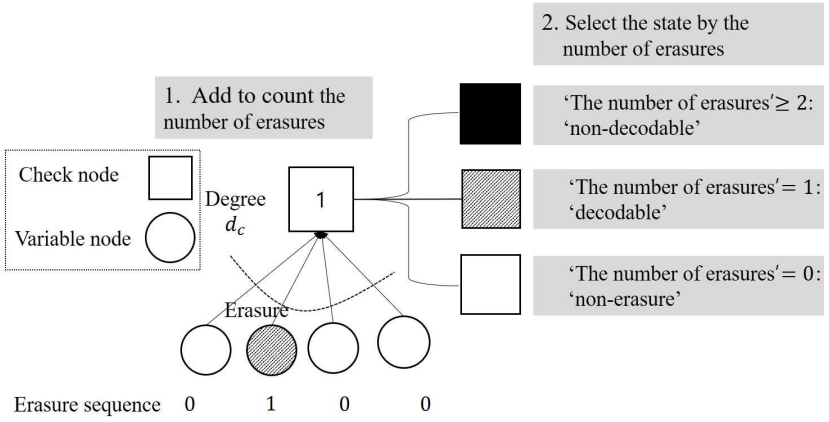

Figure 2.1: CNU operation of IED.

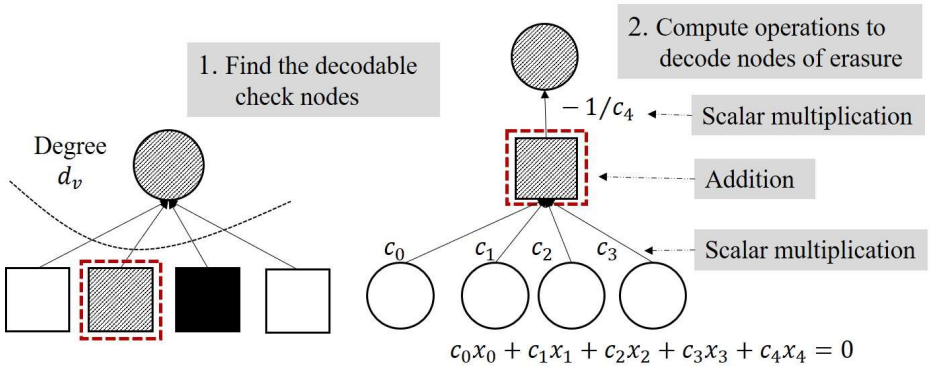

Figure 2.2: VNU operation of IED.

ing the number of the erasure states of the variable nodes connected to itself. A decodable state of a check node is declared when the number of the connected variable nodes in the erasure state is 1. If the check node is connected to two or more variable nodes in the erasure state, then a non-decodable state is declared for the check node. The check nodes with no connected variable nodes in the erasure state are called non-erasure states. The VNU operation is a procedure by which variable nodes in the erasure state are decoded using connected decodable check nodes. Fig. 2.1 and 2.2 shows the CNU and VNU operations. Note that after the decodable check nodes decode the corresponding variable nodes in the erasure state, then state of the check node is changed to the non-erasure state. In the next subsection, I will introduce the concept of AGD.

2.1.2 Automorphism Group Decoder

AGD can be applied to cyclic codes, where AGD consists of the repeated IED and cyclic shift operations of the received vectors. That is, if there is no decodable check node, then the received vector is cyclically shifted until decodable check nodes are found. If it is found, the IED algorithm is repeatedly applied to the cyclically shifted received vectors.

It is known that the cyclic shift operation is easy to implement with negligible complexity and delay. In the AGD, IED should be performed for each cyclically shifted received vector until the decoding is successful or the number of cyclic shifts is equal to the length of codeword. Although the decoding complexity and delay of the AGD are much higher than those of the IED, the decoding performance of the AGD is much better than that of the IED.

The decoding performance and complexity of AGD can be improved by using an optimized parity check matrix. For binary case, Hehn uses cyclic orbit generator (cog) and cog family to construct parity check matrix [6]. Two vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ are said to be cyclically indistinguishable if the cyclic shift of $\mathbf{v}_1$ is identical to $\mathbf{v}_2$, and otherwise,

cyclically distinguishable. Then, the cog is defined as the cyclically distinguishable binary codeword of a dual code with minimum Hamming weight, which can be used as a row of the parity check matrix. Cog family is the set of cogs that has the same Hamming autocorrelation property, where the Hamming autocorrelations of cog's are defined as

$$\begin{aligned} |OO_\tau| = \mathbf{cog} \cdot \mathbf{cog}^{(\tau)}, |ZO_\tau| = (\mathbf{1} - \mathbf{cog}) \cdot \mathbf{cog}^{(\tau)} \\ |OZ_\tau| = \mathbf{cog} \cdot (\mathbf{1} - \mathbf{cog}^{(\tau)}), |ZZ_\tau| = (\mathbf{1} - \mathbf{cog}) \cdot (\mathbf{1} - \mathbf{cog}^{(\tau)}) \end{aligned} \tag{2.1}$$

where $\mathbf{cog}^{(\tau)}$ is right cyclic shift of **cog** by τ and $\cdot$ denotes inner product. Then, the parity check matrix is constructed by $n-k$ cogs, where it is desirable to select the $n-k$ cogs from cog families minimizing the upper bounds of Theorem 3.9 in [6]. In fact, selecting such desirable cogs and cog families requires exponential-time complexity because all the minimum weight codewords should be searched. Moreover, there is no guarantee that it is the optimal construction. Therefore, it is very difficult to construct the parity check matrix for AGD of the cyclic codes with long codelength. Therefore, new construction methods with lower polynomial-time complexity and good decoding performance needs to be researched.

2.2 Binary Locally Repairable Codes for Distributed Storage System

In the distributed storage system, lots of storage nodes are distributed in order to store and process a big data. In DSS, single node can fail either temporarily or permanently. Repair process, replacing a failed node with a new node with same functionality, is common phenomena. Computation and communication cost of the repair process are not small compared to the total cost. Therefore, LRC is designed to minimize the number of storage nodes to be accessed during the repair process which can reduce the communication bandwidth and latency, called the locality r. In general, LRC has the parameters (n, k, d, r). It is clear that LRC with locality r should satisfy the condition

that the union of supports of checks whose Hamming weights are less than or equal to $(r+1)$ is equal to $[n]$.

As a class of LRCs, binary LRCs [32] have been researched actively for advantage of low-complex XOR operation. In [30], the advantage of binary LRCs are numerically analyzed. In specific, $(n,k,d,r) = (15,10,4,6)$ binary LRC is compared with $(16,10,4,5)$ nonbinary LRC, $(14,10)$ RS codes, and 3-replications which were used in the conventional DSSs. In the four metrics; encoding complexity, repair complexity, mean time to data loss (MTTDL), and storage capacity, the proposed binary LRC has evenly good performance whereas the conventional codes have fatal drawbacks for some metrics.

In addition, it is called disjoint repair group if supports of local checks are pairwisely disjoint. It is widely used in the constructions because DSS has the hierarchical structure from the storage unit to the whole data center, which makes correlated and burst erasures among the same region. Therefore, the repair group should be constructed disjointly so that the only independent nodes are used in the repair process.

For following subsections, I introduce bounds, optimality, and existing constructions of LRCs.

2.2.1 Bounds and Optimalities of Binary LRCs

In this subsection, bounds and optimalities of binary LRC are discussed. Especially, bounds and optimalities between two parameters such as d and r have been studied, where (n,k,d,r) LRC is said to be r-optimal if (n,k,d,r') LRC does not exist for $r' < r$. Similarly, (n,k,d,r) LRC is said to be d-optimal and k-optimal if (n,k,d',r) and (n,k',d,r) LRCs do not exist for $d' > d$ and $k' > k$, respectively. LRC is said to be optimal if LRC is r-optimal, d-optimal, and k-optimal. Moreover, the well-known bounds for LRCs are listed as follows.

Proposition 2.1 (Singleton-like bound [35]). *For an* (n, k, d, r) *LRC,*

$$d \leq n - k - \left\lceil \frac{k}{r} \right\rceil + 2. \tag{2.2}$$

Proposition 2.2 (Cadambe-Mazumda (C-M) bound [36]). *For an* (n, k, d, r) *linear LRC,*

$$k \leq \min_{x \in Z^+} \{xr + k_{opt}^{(q)}(n - x(r+1), d)\} \tag{2.3}$$

where $k_{opt}^{(q)}(a, b)$ *is the largest possible code dimension for codelength* a *and minimum distance* b *of the linear code over* $\mathbb{F}_q$*.*

For linear binary LRCs, explicit bound with $d \geq 5$ was proposed as follows.

Proposition 2.3 ($\mathcal{L}$-space bound [37]). *For an* (n, k, d, r) *linear binary LRC with* $d \geq 5$ *and* $2 \leq r \leq \frac{n}{2} - 2$*, it holds*

$$k \leq \frac{rn}{r+1} - \min\left\{ \log_2\left(1 + \frac{rn}{2}\right), \frac{rn}{(r+1)(r+2)} \right\}. \tag{2.4}$$

For for linear binary LRCs with disjoint repair groups, more improved bound from (2.4) was also proposed as follows.

Proposition 2.4 ($\mathcal{L}$-space bound with disjoint repair group [41]). *For the* (n, k, d, r) *binary linear LRC with* $d = 2t + 2$ *and* $n = (r+1)l$*, it holds if* $t + 1$ *is odd,*

$$k \leq \frac{rn}{r+1} - \left\lceil \log_2 \left(\sum_{0 \leq i_1 + \ldots + i_l \leq \lfloor \frac{d-1}{4} \rfloor} \prod_{j=1}^{l} \binom{r+1}{2i_j} \right) \right\rceil, \tag{2.5}$$

else if $t + 1$ *is even,*

$$k \leq \frac{rn}{r+1} - \left\lceil \log_2 \left(\sum_{0 \leq i_1 + \ldots + i_l \leq \lfloor \frac{d-1}{4} \rfloor} \prod_{j=1}^{l} \binom{r+1}{2i_j} + \frac{\sum_{i_1 + \ldots + i_l = \frac{d}{4}} \prod_{j=1}^{l} \binom{r+1}{2i_j}}{\lfloor \frac{n}{t+1} \rfloor} \right) \right\rceil. \tag{2.6}$$

There exist many optimal and near-optimal constructions of LRCs. For $q \geq n + 1$, there exist optimal LRCs for almost all parameters. However, constructions of optimal LRCs on the $q < n + 1$ including the binary cases are not completely known.

Table 2.1: Optimal existing binary LRCs with disjoint repair group

(d, r)	Constructions
$(4, r \leq 3)$	bipartite [39]
$(6, 2)$	cyclic code(limited n) [29], partial spread [41]
$(6, 3)$	partial spread [41]
$(10, 2)$	cyclic code($n = 2^m \pm 1$) [29], [31]

2.2.2 Existing Optimal Constructions of Binary LRCs

Optimal binary LRCs with $d \leq 6$ have been found by various methods, which are listed in Table 2.1. Especially for $(d, r) = (6, 2)$, recent works in [41] can find the optimal LRCs of almost all cases. However, it is still open problem to find optimal or near optimal binary LRCs with disjoint repair group for $d \geq 8$ or $r \geq 3$. There are few constructions of optimal LRCs by (2.4) for $d \geq 8$ or $r \geq 3$ in [41]. In [41], some k-optimal LRCs with $d \geq 8$ are proposed using partial spread which requires exponential complexity and thus, it is hard to find optimal LRCs with long codelength. Therefore, constructions of binary LRCs with new parameters and lower complexity needs to be researched.

2.3 Channels with Block Interference and Jamming

In this section, I introduce channels with block interference with binary phase shift keying (BPSK) modulation and jamming with M-ary frequency shift keying (MFSK) modulation. In fact, they are similar in many sense. First, I will introduce the channels with block interference.

2.3.1 Channels with Block Interference

The channel models of block interference are divided into the two cases of TS-BSC-BIs with and without BF. Here, k information bits are encoded into the (n, k) binary

protograph LDPC code with $n-k$ check nodes (CNs), n variable nodes (VNs), and code rate $R=\frac{k}{n}$. The codewords are indexed as

$$\mathbf{c}=(\mathbf{c}_1,...,\mathbf{c}_L)=(c_{1,1},c_{1,2},...c_{1,h},c_{2,1},$$
$$c_{2,2},...,c_{L,1},...,c_{L,h})\in F_2^n \quad (2.7)$$

where $\mathbf{c}_i$ denotes the partial codeword in the i-th hop and h denotes the length of the hop with $h|n$ and the number of hops $L=\frac{n}{h}$. Binary phase shift keying (BPSK) modulation converts $\mathbf{c}$ to $\mathbf{x}=(\mathbf{x}_1,...,\mathbf{x}_L)$ with elements $x_{i,j}=(-1)^{c_{i,j}}$ for $i\in[L],j\in[h]$. The received signal can be expressed as

$$y_{i,j}=\alpha_i(1-\beta_i)x_{i,j}+n \quad (2.8)$$

where α_i denotes the fading coefficients, $\beta_i\in\{0,1\}$ denotes the indicator of BI with probability $\rho=\Pr(\beta_i=1)$, and $n_{i,j}$ is additive Gaussian noise with $\mathcal{N}(0,\sigma^2)$. Note that existence of BI is often treated in the binomial distribution independent from BF and noise [55], [56]. In this model, BI hops only contain additive Gaussian noise independent from the desired signal. In this dissertation, two specific channel models are considered as follows:

1) Channel with BI and without BF

$$\alpha_i=\begin{cases}0 & \text{with BI and wihtout BF in the } i\text{-th hop}\\ 1 & \text{without BI and BF in the } i\text{-th hop}\end{cases} \quad (2.9)$$

2) Channel with BI and BF

$$\alpha_i=\begin{cases}0 & \text{with BI and BF in the } i\text{-th hop}\\ \alpha_i & \text{without BI and with BF in the } i\text{-th hop.}\end{cases} \quad (2.10)$$

Clearly, the hops with BI contain only interference and the hops without BI and BF contain the desired signal and additive Gaussian noise. Here, I assume that α_i follows the Nakagami-m distribution as

$$f_{\alpha_i}(\xi)=\frac{2m^m\xi^{2m-1}}{\Gamma(m)}e^{-m\xi^2} \quad (2.11)$$

where $\Gamma(m)$ is the Gamma function. Then, α_i^2 follows Gamma distribution with $E(\alpha_i^2) = 1$. The Nakagami-m fading model has the advantages of simple computation and good matching with theoretical and practical fading models when selecting m properly. The following cases show the relationships between Nakagami-m and other theoretical fading models as:

1) $m = 1$; Rayleigh fading

2) $m = \frac{(K+1)^2}{2K+1}$; approximated Rician fading with parameter K

3) $m \to +\infty$; additive Gaussian channel.

At the receiver, the LDPC decoder uses the BP algorithm using the CSIs of noise variance and fading coefficients, where perfect CSI can be known to the decoder. Also, the initial log-likelihood ratio (LLR) message value of the BP decoding algorithm is written as

$$m_{i,j} = \begin{cases} \frac{2\alpha_i y_{i,j}}{\sigma^2}, & \text{without BI in the } i\text{-th hop} \\ 0, & \text{with BI in the } i\text{-th hop} \end{cases} \tag{2.12}$$

where α_i is the estimate of the channel coefficient. Although the BP algorithm was initially proposed only for the Gaussian noise channel, it is known to be universally good for fading channels if I use the channel LLR as in (2.12).

The channel analysis with BI can be given using the information-theoretic method, that is, with mutual information (MI). Note that the MI of each hop with Gaussian noise is expressed as $I_G\left(\frac{\alpha_i^2 E_s}{N_0}\right)$, where $\frac{E_s}{N_0}$ denotes the signal to noise ratio per each codeword bit and

$$I_G(x) = -\int_{-\infty}^{\infty} \phi(\tau, x) \log_2[\phi(\tau, x)] d\tau - \frac{1}{2} \log_2 \frac{\pi e}{x} \tag{2.13}$$

with $\phi(\tau, x) = \frac{1}{2\sqrt{\pi/x}}(e^{-x(\tau+1)^2} + e^{-x(\tau-1)^2})$ [58]. Note that $I_G(x)$ is upper bounded by $\min\{1, \log_2(1+x)\}$. The channel capacity is the expectation of MI of each hop, which is expressed as

$$I_A = \sum_{i \in [L]} \frac{1}{L} I_G\left(\frac{\alpha_i^2 E_s}{N_0}\right). \tag{2.14}$$

Then, the outage probability is defined as $P(I_A < R)$, which is the lower bound for the frame error rate (FER) of any coding scheme for the given channel and R. It is possible to derive outage probability by generating α_i randomly and checking $I_A < R$, but the outage probability can be analyzed using the following method.

For a channel without BF, if a codeword has L_u hops with BIs, MI reduces to

$$I_A = \frac{L - L_u}{L} I_G\left(\frac{E_s}{N_0}\right). \tag{2.15}$$

The outage probability can be asymptotically given as

$$P_{out,L_u} = P(I_A < R) = u\left(\frac{LR}{L - L_u} - I_G\left(\frac{E_s}{N_0}\right)\right) \tag{2.16}$$

where $u(x)$ returns 1 if $x > 0$ and 0, otherwise. Equation (2.16) shows the outage probability of the channel with BI and without BF in the Gaussian noise channel. Thus, the channel threshold is defined as $\frac{E_b}{N_0} = \frac{E_s}{RN_0}$ satisfying $I_G\left(\frac{RE_b}{N_0}\right) = \frac{LR}{L-L_u}$, where the outage probability goes to zero. Using ρ, the outage probability is given as

$$\begin{aligned} P_{out,\rho} &= \textstyle\sum_{l'=0}^{L} p(l') P_{out,L_u=l'} \\ &= \textstyle\sum_{l'=0}^{L} \binom{L}{l'} \rho^{l'} (1-\rho)^{L-l'} u\left(\frac{LR}{L-l'} - I_G\left(\frac{RE_b}{N_0}\right)\right) \\ &= 1 - \textstyle\sum_{l'=0}^{l'_{max}} \binom{L}{l'} \rho^{l'} (1-\rho)^{L-l'} \end{aligned} \tag{2.17}$$

where $p(l')$ represents a binomial distribution and l'_{max} is a nonnegative maximum integer satisfying $\frac{LR}{L-l'_{max}} \leq I_G\left(\frac{E_s}{N_0}\right)$. Note that the channel threshold is given as $\frac{E_b}{N_0}$ satisfying $I_G\left(\frac{RE_b}{N_0}\right) = \frac{LR}{L-l'_{max}}$.

For a channel with BF, the outage probability cannot converge to zero even though $\frac{E_s}{N_0}$ is sufficiently large. For Rician fading, the implicit form of the outage probability was derived in [61]. In addition, the lower bound of the outage probability for Nakagami-m fading was also derived in [60]. The lower bound of the outage probability for the channel with BI and BF can be obtained as follows.

The i-th hop in $L - L_u$ hops without BI is called good if $\log_2(1 + \alpha_i^2 \frac{E_s}{N_0}) \geq 1$ and bad, otherwise. In fact, I have $I_G(\alpha_i^2 \frac{E_s}{N_0}) \leq 1$ for good hops because $I_G(\alpha_i^2 \frac{E_s}{N_0}) \leq \min\{1, \log_2(1 + \alpha_i^2 \frac{E_s}{N_0})\}$ [60]. Then, the probability of good hop is given as $p' =$

$\Pr(\alpha_i^2 \geq 1/\frac{E_s}{N_0})$. Let $\mathcal{U}$, $\mathcal{G}$, and $\mathcal{B}$ be sets of hops with BI, good hops, and bad hops, respectively and thus $|\mathcal{U}| = L_u$ and $|\mathcal{G} \cup \mathcal{B}| = L - L_u$. Then, the probability of t good hops for L_u hops with BI is expressed as

$$\Pr(|\mathcal{G}| = t) = \binom{L - L_u}{t} (p')^t (1 - p')^{L - L_u - t}, t \in [L - L_u]. \tag{2.18}$$

Note that $p' = \frac{\Gamma(m, m/\frac{E_s}{N_0})}{\Gamma(m)}$ for the Nakagami-m distribution, where $\Gamma(a, \xi) = \int_\xi^a t^{a-1} e^{-t} dt$ is an incomplete Gamma function [60]. Clearly, total channel capacity of L hops, I_A is upper bounded as

$$I_A \leq \frac{|\mathcal{G}| + \sum_{i \in \mathcal{B}} \log_2 \left(1 + \alpha_i^2 \frac{E_s}{N_0}\right)}{L}. \tag{2.19}$$

Thus, the outage probability can be lower bounded as

$$P_{out, L_u, |\mathcal{G}|} \geq \Pr\left(|\mathcal{G}| + \sum_{i \in \mathcal{B}} \log_2 \left(1 + \alpha_i^2 \frac{E_s}{N_0}\right) \leq LR\right). \tag{2.20}$$

Let A_i be a random variable $\log_2 \left(1 + \alpha_i^2 \frac{E_s}{N_0}\right)$, whose pdf A_i is given as [60]

$$f_{A_i}(\xi) = \begin{cases} \frac{f_{\alpha_i^2}(\frac{2^\xi - 1}{E_s/N_0})}{F_{\alpha_i^2}(\frac{1}{E_s/N_0})} \frac{2^\xi \log 2}{E_s/N_0}, & 0 \leq \xi \leq 1 \\ 0, & \text{otherwise} \end{cases} \tag{2.21}$$

where $F_{\alpha_i^2}$ represents the cumulative distribution function of α_i^2. Then, the lower bound of the outage probability of L_u hops with BI is rewritten as

$$P_{out, L_u} \geq \sum_{t=0}^{L - L_u} F(LR - t) \Pr(|\mathcal{G}| = t) \tag{2.22}$$

where $F(LR - t) = \Pr(\sum_{k=1}^{L - L_u - t} A_k < LR - t)$, which can be calculated numerically by the fast Fourier transform (FFT). Suppose that the hops with BI are generated with probability ρ and the lower bound of the outage probability is derived as

$$P_{out, \rho} \geq \sum_{l'=0}^{L} \binom{L}{l'} \rho^{l'} (1 - \rho)^{L - l'} P_{out, L_u = l'}. \tag{2.23}$$

2.3.2 Channels with Jamming with MFSK and FHSS Environment

From now on, the channel with jamming with MFSK modulation and FHSS environment are introduced. For block Rayleigh fading channel, I consider an i-th symbol in the k-th hops, where $0 \leq i \leq I-1$ and $0 \leq k \leq K-1$. Suppose that the messages are sent on the $\bar{m}$-th tone of M-ary FSK, Then the received symbol without jamming $y_{\bar{m},k,i}$ is expressed as

$$y_{\bar{m},k,i} = \alpha_k \sqrt{\mathcal{E}_{k,i}} + n \tag{2.24}$$

where $\mathcal{E}_{k,i}$ is the energy value of the symbol, n is an additive white Gaussian noise with zero mean and variance $\frac{N_0}{2}$ and α_k is normalized Rayleigh fading factor with $E[{\alpha_k}^2] = 1$ and probability density function $p(a) = 2\alpha_k e^{-{\alpha_k}^2}$. Note that α_k depends on the hop in block fading. For MFSK demodulation aspects, cosine and sine integrators detect phase ϕ with uniformly distribution over $[-\pi, \pi]$. The power, occurrence, and interval of jamming rely on the category of jamming, which will be discussed in the next subsection. In short, jamming signal is expressed as

$$\delta(k,i) = \begin{cases} 1, & \text{If jamming occurred in } y_{m,k,i} \\ 0, & \text{otherwise.} \end{cases} \tag{2.25}$$

Then, the received signal is expressed as

$$r_{mc,k,i} = \begin{cases} \alpha_k \sqrt{\mathcal{E}_{k,i}} \cos\phi + j\delta(k,i) + n, & m = \bar{m} \\ j\delta(k,i) + n, & \text{otherwise} \end{cases} \tag{2.26}$$

$$r_{ms,k,i} = \begin{cases} \alpha_k \sqrt{\mathcal{E}_{k,i}} \sin\phi + j\delta(k,i) + n, & m = \bar{m} \\ j\delta(k,i) + n, & \text{otherwise.} \end{cases} \tag{2.27}$$

Demodulator calculates the squared values and selects the largest one as demodulated message as

$$m'_{k,i} = \text{argmax}_m(r_{m,k,i}) \tag{2.28}$$

where $r_{m,k,i} = \sqrt{r_{mc,k,i}^2 + r_{ms,k,i}^2}$.

In the decoding procedure, one of the crucial parameter is binary log likelihood ratio (LLR). Binary LLR value in the decoder can be different by existence of side information, but α_k or statistics of j is difficult to know. Here, the decoder uses LLR considering only statistics of n, which is expressed as

$$\Lambda(r_{k,i}) = \log \frac{\sum_{G(m,i)=0} I_0\left(\frac{\sqrt{(\mathcal{E}_{k,i} r_{m,k,i}^2}}{\frac{N_0}{2}}\right)}{\sum_{G(m,i)=1} I_0\left(\frac{\sqrt{(\mathcal{E}_{k,i} r_{m,k,i}^2}}{\frac{N_0}{2}}\right)} \tag{2.29}$$

where $G(m,i)$ is a function that returns 0 when the i-th bit mapped from message m is 0 and otherwise, 1.

Chapter 3

New Two-Stage Automorphism Group Decoders for Cyclic Codes in the Erasure Channel

In this chapter, new two-stage automorphism group decoders for cyclic codes in the erasure channel are proposed. For this, additional definitions are introduced in the following section.

3.1 Some Definitions

In this section, some definitions for the proposed algorithms are presented as follow. First, several definitions of a binary sequence are presented. Let $s_D(t)$ denote a characteristic sequence of index set D such that $s_D(t) = 1$ if $t \in D$ and $s_D(t) = 0$, otherwise. Two binary sequences frequently used in this chapter are defined as follows.

Definition 3.1 (Erasure sequence)**.** *Erasure sequence $s_e(t)$ is defined as a characteristic sequence of the erasure set S_e, which is the set of indices of erasure symbols in the vector received over the erasure channel.*

Definition 3.2 (Parity check sequence)**.** *Parity check sequence $s_p(t)$ of the $(n-k) \times n$ parity check matrix H of the (n, k) cyclic code is a binary sequence of length n defined*

as

$$s_p(t) = \begin{cases} 1, & \textit{if } wt(\boldsymbol{h}_t) = 1 \\ 0, & \textit{otherwise} \end{cases} \tag{3.1}$$

where $\boldsymbol{h}_t$ is the t-th column of H. Furthermore, let S_p denote the support set of $s_p(t)$, i.e., the set of indices of column vectors with Hamming weight 1.

For column indices of the parity check matrix H, the components of S_p are called standard indices and otherwise, non-standard indices. Thus, the number of 1's in a length of $s_p(t)$ is smaller than or equal to $n-k$. The Hamming cross-correlation of two binary $\{0,1\}$ sequences, $s_e(t)$ and $s_p(t)$, is defined as

$$R_H(\tau) = \sum_{t=0}^{n-1} s_e(t) s_p(t+\tau) \tag{3.2}$$

where $R_H(\tau)$ takes values in $\{0, 1, 2, ..., n-k\}$. The stopping redundancy for the parity check matrix is defined as follows.

Definition 3.3 (Stopping redundancy ρ [11])**.** *Stopping redundancy ρ of the code C is the minimum number of check equations for which the decoder can correct all of the erasure patterns with erasure symbols less than or equal to $d-1$, where d is the minimum distance of the code C.*

A mask is a useful notation to represent the parity check matrix of MDS codes because only the location of nonzero values in the parity check matrix is of our interests, which is defined as follows.

Definition 3.4 (Mask)**.** *Mask A is an $(n-k) \times n$ binary matrix whose component $a_{i,j}$ is 1 if the (i,j) component of matrix H is nonzero and otherwise, zero.*

Note that mask A is the same as H for binary codes. It is known that the ML decoding performance in the erasure channel is best, that is, a practical decoder in the erasure channel can have the same or inferior erasure decoding performance to that of the ML decoder. At this point, the perfect decoding is defined as follows.

Definition 3.5 (Perfect decoding)**.** *It is called perfect decoding in the erasure channel if its erasure decoding performance is the same as that of ML decoder.*

In general, perfect decoding is not common because it is rarely possible to show it. In this chapter, the perfect decoding is shown by checking all cases of erasure patterns for some of the cyclic codes with small values of n and k.

There is an example of $(8, 4)$ RS code, where an ML decoder can correct any four erasure symbols regardless of their locations.

Example 3.1 (AGD)**.** *Suppose that an (8,4) RS code is defined in F_{17} and its generator polynomial is given as*

$$g(x) = \prod_{i=0}^{k-1}(x - 2^i) = x^4 + 2x^3 + 2x^2 + 16x + 13. \tag{3.3}$$

Then, the corresponding parity check polynomial is obtained as

$$h(x) = \frac{(x^8 - 1)}{g(x)} = x^4 + 15x^3 + 2x^2 + x + 13 \tag{3.4}$$

and the systematic parity check matrix can be constructed from $h(x)$ as

$$\mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 0 & 4 & 9 & 8 & 4 \\ 0 & 1 & 0 & 0 & 1 & 2 & 11 & 9 \\ 0 & 0 & 1 & 0 & 15 & 5 & 15 & 9 \\ 0 & 0 & 0 & 1 & 15 & 2 & 1 & 13 \end{pmatrix}. \tag{3.5}$$

The parity check matrix H can be described by a bipartite graph as shown in Fig. 3.1. Circles represent variable nodes and squares represent check nodes and the edge between the i-th circle and the j-th square indicates that the (i, j) component of H is nonzero. In Fig. 3.1, the circles with dashed line are variable nodes in erasure states, where there are four erasure symbols. Then, the erasure sequence $s_e(t)$ is $(1, 0, 1, 0, 1, 0, 0, 1)$ and the parity check sequence $s_p(t)$ is $(1, 1, 1, 1, 0, 0, 0, 0)$.

The AGD procedure is described in Fig. 3.1. In the first bipartite graph Fig. 3.1, the decoder performs CNU operations and confirms that there is no check node in a

decodable state. The IED declares a decoding failure, whereas the AGD proceeds to the next decoding procedure by cyclic shifting the received vector. In the second bipartite graph of Fig. 3.1, one right cyclic shift operation for the received vector and CNU operation are done. After four CNU operations, it is found that one check node is in a decodable state, which can proceed to a VNU operation to correct the fourth erasure symbol. In the third bipartite graph of Fig. 3.1, after three CNU operations are performed, the decoder finds that the other check nodes are all in decodable states, which can proceed to three VNU operations to correct the three remaining erasure symbols and then the decoding procedure is completed. In the above decoding procedure for AGD, three IED operations and one cyclic shift are performed.

Example 3.1 shows that the AGD has superior performance to the IED. However, successive IEDs are needed for each cyclic shift operation, which is the main issue of the decoding complexity and delay in the AGD. This example shows that it is needed to select cyclic shift values and construct parity check matrix carefully to reduce the decoding complexity. In the next section, I propose a new decoding scheme for cyclic codes.

3.2 Modification of Parity Check Matrix and Two-Stage AGD

In this section, I propose a new modification method for the parity check matrix and a two-stage decoding algorithm, and the result of a numerical analysis for the proposed decoding algorithm is discussed.

3.2.1 Modification of the Parity Check Matrix

First, I propose a method to modify the parity check matrix for the proposed two-stage decoding algorithm because the decoding performance of the proposed two-stage decoding algorithm depends on the structure of the parity check matrix. The following criteria are used for the modification of the parity check matrix using Definition 3.2.

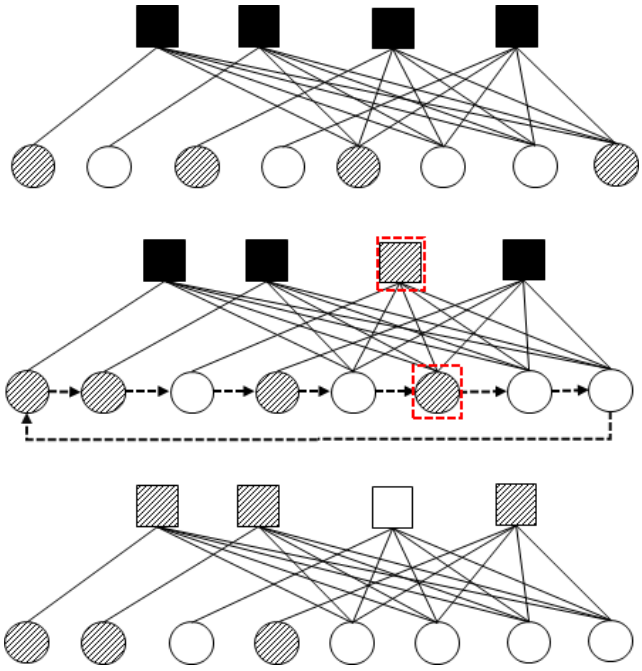

Figure 3.1: AGD for $(8, 4)$ RS code in Example 3.1.

1) Each row of the parity check matrix has as low Hamming weight as possible.

2) The parity check sequence of the parity check matrix has Hamming autocorrelation values as low as possible.

3) Modify the parity check matrix such that as many of its column vectors as possible are the standard vectors.

In fact, the best criteria for the parity check matrix of (n, k) cyclic codes can be described as:

1) The Hamming weights of all rows of the parity check matrix are equal to the minimum Hamming weight of its dual code.

2) All Hamming autocorrelation values of the parity check sequence of the parity check matrix are equal.

3) $n-k$ columns of the parity check matrix are standard vectors.

It is easy to check that in order for the parity check sequences to satisfy the second criterion, they should be the characteristic sequences of cyclic difference sets D_C with parameters (n,k,λ) for (n,k) cyclic codes, if their parameters are allowed for the cyclic difference sets. Note that k-subset D_C of a cyclic group G with order n is a (n,k,λ) cyclic difference set if every nonzero component of G has exactly λ representations as a difference $d_c - d'_c$ with components from D_C [16]. It is known that some cyclic codes satisfy the above best criteria. The other criteria can be compromised if one criterion cannot be achieved due to the other criteria. For these cases, it is recommended to apply the priority as the descending order to the three criteria from the empirical results, the detailed procedures of which are given in the numerical analysis in the next subsection.

The new decoding algorithms together with the proposed modification of the parity check matrix will be explained in the next subsection.

3.2.2 A New Two-Stage AGD

Using AGD algorithm, I propose a new two-stage AGD of (n,k) cyclic codes in the erasure channel as follows.

a) Preprocessing Stage (First Decoding Stage)

Find a $\{0,1\}$ parity check sequence $s_p(t)$ of length n from the parity check matrix H of an (n,k) cyclic code. Find a $\{0,1\}$ erasure sequence $s_e(t)$ of length n from the received vector $\mathbf{r} = (r_0, r_1, ..., r_{n-1})$. Then, calculate a Hamming cross-correlation as

$$R_H(\tau) = \sum_{t=0}^{n-1} s_p(t) s_e(t+\tau),\ 0 \le \tau \le n-1. \tag{3.6}$$

Clearly, $R_H(\tau)$ takes values of the nonnegative integers less than or equal to $\min\{|S_e|, |S_p|\}$ because $|S_e|$ is the number of erasure symbols and $|S_p|$ is the number of standard vectors of the parity check matrix. It can be assumed that the decoding complexity of the

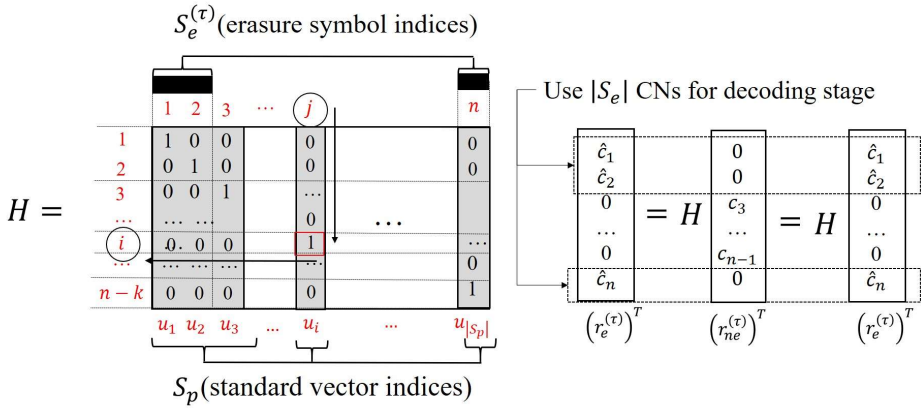

Figure 3.2: The second stage decoding procedure of the TS-AGD of τ such that $R_H(\tau) = |S_e|$.

preprocessing stage for each τ is analogous to the CNU of one check node. If there exists τ such that $R_H(\tau) = |S_e|$, then proceed to the second decoding stage. If not found, cyclically shift the received vector and proceed to the second decoding stage for $\mathbf{r}^{(\tau)}$ in the order of τ's such that values of $R_H(\tau)$ are decreasing, where $\mathbf{r}^{(\tau)}$ is a cyclic shift of $\mathbf{r}$ by τ.

b) IED Decoding Stage (Second Decoding Stage)

In the second decoding stage, the IED algorithm is used for decoding of the cyclically shifted received vector according to the values of $R_H(\tau)$. Recall that S_p is the support set of $s_p(t)$. Let $\mathbf{r}^{(\tau)} = (r_{n-\tau}, r_{n-\tau+1}, ..., r_{n-\tau-1})$ be a received vector cyclically shifted by τ, where erasure symbols are located in the indices in $S_e^{(\tau)} = \{t | s_e(t-\tau) = 1, 0 \leq t \leq n-1\}$.

1) For τ such that $R_H(\tau) = |S_e|$: It is clear that $S_e^{(\tau)} \subseteq S_p$, that is, all of the erasure symbols in $\mathbf{r}^{(\tau)}$ are located in the indices of standard vectors. Note that the i-th component of the received vector $\mathbf{r}$ is expressed as the transmitted symbol c_i for a non-erasure symbol and $\hat{c}_i$ for an erasure symbol. Suppose that $\mathbf{r}^{(\tau)}$ can be

split into two n-tuple vectors as

$$\mathbf{r}^{(\tau)} = \mathbf{r}_e^{(\tau)} + \mathbf{r}_{ne}^{(\tau)} \tag{3.7}$$

where the j-th component of $\mathbf{r}_e^{(\tau)}$ is denoted as $\hat{c_j}$ for $j \in S_e^{(\tau)}$ and otherwise, 0 and the j-th component of $\mathbf{r}_{ne}^{(\tau)}$ is equal to the j-th component of $\mathbf{r}^{(\tau)}$ for $j \notin S_e^{(\tau)}$ and otherwise, 0. In general, the syndrome vector should be zero as

$$S = H(\mathbf{r}^{(\tau)})^T = H(\mathbf{r}_e^{(\tau)})^T + H(\mathbf{r}_{ne}^{(\tau)})^T = 0 \tag{3.8}$$

and thus

$$H(\mathbf{r}_e^{(\tau)})^T = H(\mathbf{r}_{ne}^{(\tau)})^T. \tag{3.9}$$

If the j-th column vector of H is the i-th standard vector u_i, $\hat{c_j}$ is equal to the i-th component of $H(\mathbf{r}_{ne}^{(\tau)})^T$ because $R_H(\tau) = |S_e|$. Clearly, each j-th column for $j \in S_e \subset S_p$ has a different standard vector u_i. In this case, I can recover all of the erasure symbols by $H(\mathbf{r}_{ne}^{(\tau)})^T$ in one iteration, which is described in Fig. 3.2.

2) For τ such that $R_H(\tau) = |S_e| - 1$: In this case, I have one erasure symbol in the non-standard vector of H and the other erasure symbols are located in the column indices in S_p. Here, the decoding process is done in two steps, that is, one for one erasure symbol in the non-standard vector of H and the other for the other erasure symbols with indices in S_p. Suppose that the set of erasure symbol indices is given as $\{e_0, e_1, e_2, ..., e_{z-1}\}$, where z is the number of erasure symbols. Suppose that the e_j-th column is the i_j-th standard vector u_{i_j}, $0 \leq j \leq z-2$, and the e_{z-1}-th column of H is a non-standard vector. I also have $|S_p| - z + 1$ standard vectors in H, where non-erasure symbols are located. In the first decoding step, assume that for $z \leq i \leq |S_p|$, some i_j-th component of the e_{z-1}-th column of H is equal to 1. Then, using the i_j-th row of H, the erasure symbol $\hat{c}_{e_{z-1}}$ can be recovered because there is no erasure symbol except for $\hat{c}_{e_{z-1}}$ at the positions of component 1 in the i_j-th row of H. Then, I

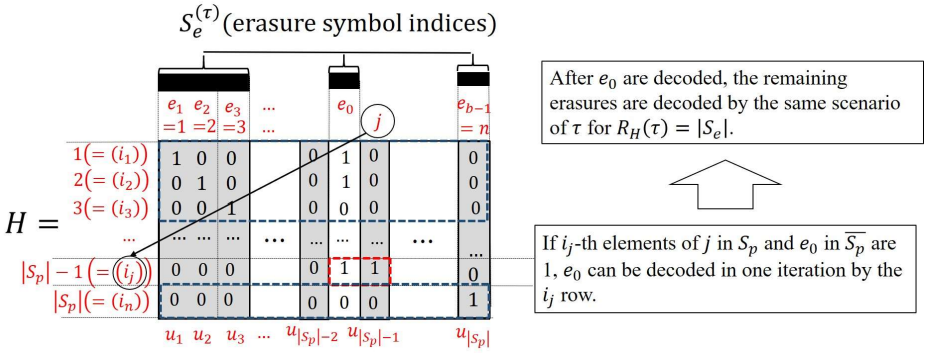

Figure 3.3: The second stage decoding procedure of the TS-AGD of τ such that $R_H(\tau) = |S_e| - 1$.

go to the second decoding stage, which is the same as that of $R(\tau) = |S_e|$. If the i_j-th component of the e_{z-1}-th column of H is 0, decoding of the first step cannot be successful because e_{z-1} disappears in the IED procedure. If the first decoding step is not successful, then I try to decode it for other τ values such that $R_H(\tau) = |S_e| - 1$. The second decoding procedure is described in Fig. 3.3.

3) For τ such that $R_H(\tau) \leq |S_e| - 2$: Let $\bar{S}_p = \{t | s_p(t) = 0\}$, i.e., the complement of S_p. Let $S_p = S_{p_e} \cup S_{p_{ne}}$, where S_{p_e} is a subset of indices such that the erasure symbols exist and $S_{p_{ne}} = S_p \setminus S_{p_e}$. Similarly, let $\bar{S}_p = \bar{S}_{p_e} \cup \bar{S}_{p_{ne}}$ and then clearly, $|S_{p_e}| = R_H(\tau)$. For $j \in S_{p_{ne}}$, suppose that the j-th component of the e_i-th column of H with $e_i \in \bar{S}_{p_e}$ is 1 and that the j-th components of the other columns with indices in $\bar{S}_{p_e} \setminus \{e_j\}$ of H are all zero and further, there exists u_j in the columns with indices in $S_{p_{ne}}$. Then, I can recover the erasure symbol with index e_i. That is, all erasure symbols except for $\hat{c}_{e_i}$ are disappeared in the inner product of the j-th row of H and the received vector cyclically shifted by τ and thus $\hat{c}_{e_i}$ can be recovered. To decode the remaining erasure symbols, it is needed to return to the preprocessing stage to find the values of τ's with higher values of $R_H(\tau)$. The second decoding stage of the proposed two-stage

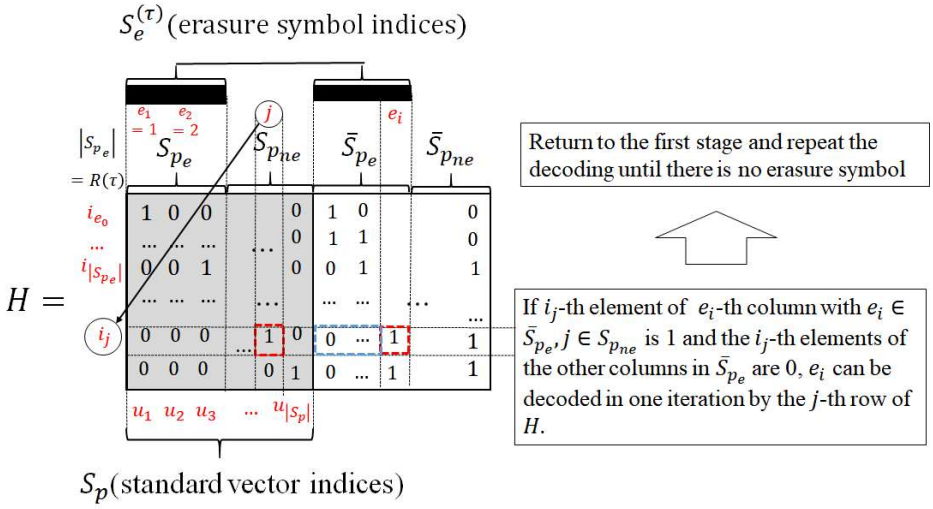

Figure 3.4: The second stage decoding procedure of the TS-AGD of τ such that $R_H(\tau) \leq |S_e| - 2$.

decoding algorithm is described in Fig. 3.4.

The overall decoding procedure is described in Algorithm 3.1 and Fig. 3.5 shows a flowchart to summarize the proposed decoding algorithm.

3.2.3 Analysis of Modification Criteria for the Parity Check Matrix

This subsection analyzes the modification criteria of H for (n, k) cyclic codes. The first criterion is related to the number of standard vectors, that is, the number of t's such that $s_p(t) = 1$, which is less than or equal to $n - k$. As described in the previous subsection, the proposed TS-AGD procedure can be done for the cyclically shifted received vector $\mathbf{r}^{(\tau)}$ such that $R_H(\tau)$ has higher values. As the number of 1's in $s_p(t)$ increases, it is more probable for $R_H(\tau)$ to have high values.

The second criterion is how to locate the standard vectors in the parity check matrix. It is not easy to prove the second criterion and thus the following theorem replaces the proof of the second criterion. First, I need a lemma for proving the following the-

Algorithm 3.1 Two-stage AGD

Require: received codeword $\mathbf{r}$, $s_p(t)$, modified H, and IED

$U = \phi, \mathbf{v} \leftarrow \mathbf{0}$; {$\mathbf{v}$:$n$-tuple vectors}

$\mathbf{r}^{(0)} \leftarrow \mathbf{r}$

a;

for $\tau = 0$ to $n-1$ **do**

$s_e(t) \leftarrow \mathbf{r}^{(0)}$; {obtain $s_e(t)$ from $\mathbf{r}^{(0)}$}

Calculate $R_H(\tau) = \sum_{t=0}^{n-1} s_e(t) s_p(t+\tau)$

if $R_H(\tau) = |S_e|$ **then**

Obtain $\mathbf{r}^{(\tau)}$ by cyclic shifting $\mathbf{r}^{(0)}$ by τ

Do IED for $\mathbf{r}^{(\tau)}$

STOP

end if

$\mathbf{v}_\tau \leftarrow R_H(\tau)$; {$\mathbf{v}_\tau$: τ-th component of $\mathbf{v}$}

end for

for $i = 1$ to n **do**

$\tau' \leftarrow \text{argmax}_{\tau \in [0,n-1] \setminus U} \mathbf{v}_\tau, U \leftarrow U \cup \{\tau'\}$

Obtain $\mathbf{r}^{(\tau')}$ by cyclic shifting $\mathbf{r}^{(0)}$ by τ'

Do IED for $\mathbf{r}^{(\tau')}$

if $\mathbf{v}_{\tau'} = 1$ and the erasures in the non-standard indices are decoded by IED **then**

Do IED to decode the remaining erasure symbols

STOP

else if there exist the decoded erasure symbols by IED **then**

$U \leftarrow \phi$

Obtain $\mathbf{r}^{(0)}$ by cyclic shifting $\mathbf{r}^{(\tau')}$ by $n - \tau'$

Goto a;

end if

Obtain $\mathbf{r}^{(0)}$ by cyclic shifting $\mathbf{r}^{(\tau')}$ by $n - \tau'$

end for

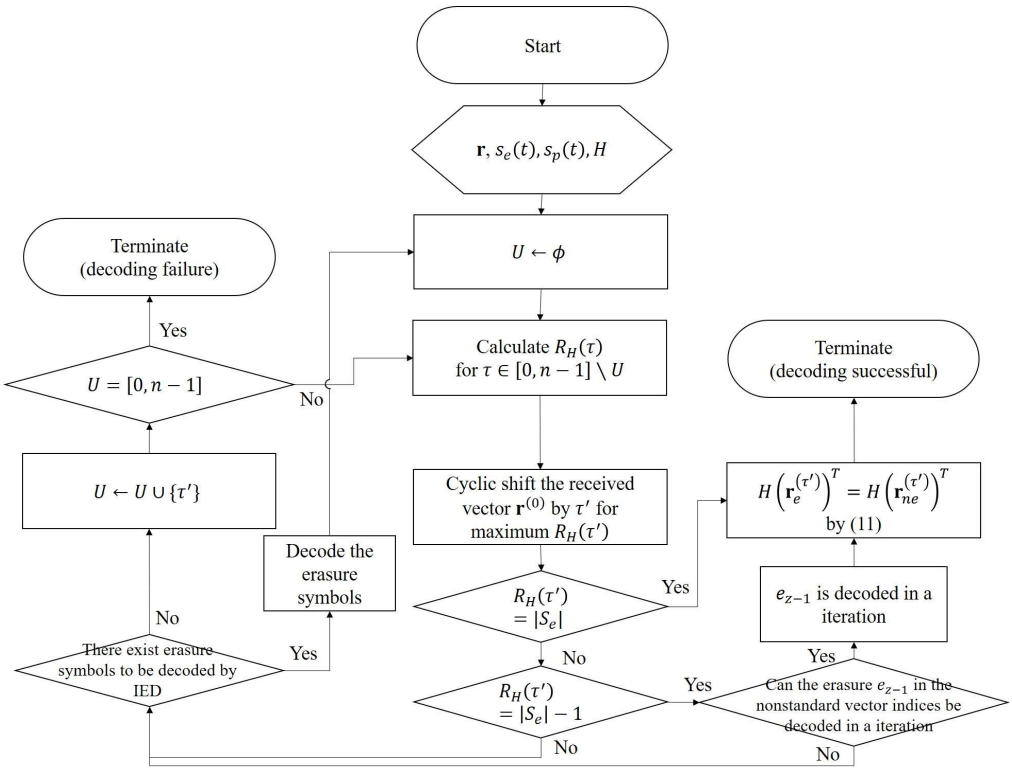

Figure 3.5: Flowchart of the TS-AGD algorithm.

orem.

Lemma 3.1 (Bonferroni inequality [14])**.** *Let E_i, $i \in A$, be sets of components. Then, I have the following inequality as*

$$\sum_{I \subset A, |I|=1} |E_i| - \sum_{I \subset A, |I|=2} \left| \bigcap_{i \in I} E_i \right| \leq \left| \bigcup_{i \in A} E_i \right| \leq \sum_{I \subset A, |I|=1} |E_i| - \frac{2}{|A|} \sum_{I \subset A, |I|=2} \left| \bigcap_{i \in I} E_i \right|. \quad (3.10)$$

Theorem 3.1. *In the upper bound of Lemma 3.1, the number of occurrences of $R_H(\tau) \geq |S_e| - 1$ for $0 \leq \tau \leq n-1$ is maximized if the parity check sequence of the modified parity check matrix has a particular constant dependent on $|S_e|$ autocorrelation value.*

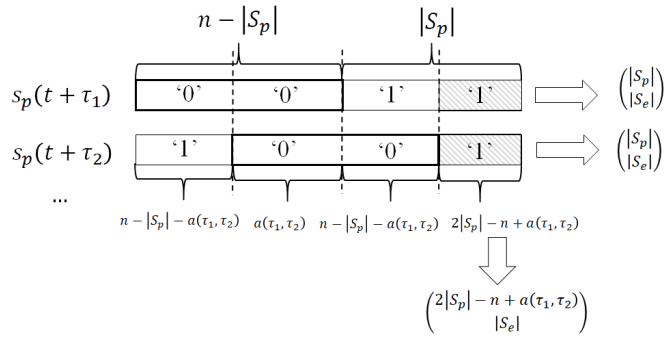

Figure 3.6: Number of doubly counted erasure patterns for τ such that $R_H(\tau) = |S_e|$.

Proof. First, it is desirable for the proposed decoding algorithm to successfully decode more erasure patterns, which is possible if $R_H(\tau) \geq |S_e| - 1$. Thus, I have to modify the parity check matrix, for which $R_H(\tau) \geq |S_e| - 1$ is most common for as many shift values τ as possible. The following two cases are considered.

1) $R_H(\tau) = |S_e|$:

This means that $S_e^{(\tau)} \subseteq S_p$. It is easy to check that in $R_H(\tau)$, it is equivalent to cyclically shift $s_p(t)$ instead of $s_e(t)$. Let $S_p^{(\tau)}$ be the support set of $s_p(t+\tau)$. Let E_τ be the set of erasure patterns which can be successfully recovered by $s_p(t+\tau)$. Then, I have $|E_\tau| = \binom{|S_p|}{|S_e|}$, which leads to

$$\sum_{\tau=0}^{n-1} |E_\tau| \leq n\binom{|S_p|}{|S_e|}. \tag{3.11}$$

It is easy to check that doubly counted erasure patterns are included in (3.11), which should be excluded. If the shaded parts in Fig. 3.6 include all the erasure symbols, those erasure patterns are doubly counted, where $a(\tau_1, \tau_2)$ denotes the number of pairs $(s_p(t+\tau_1), s_p(t+\tau_2)) = (1, 1)$. Thus I have $\binom{2|S_p|+a(\tau_1,\tau_2)-n}{|S_e|}$ doubly counted erasure patterns. Using Lemma 3.1, the number of erasure pat-

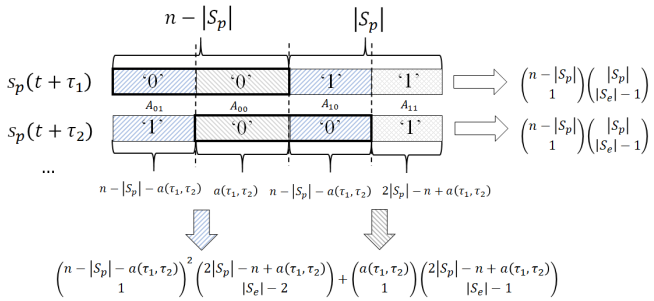

Figure 3.7: Number of doubly counted erasure patterns for τ such that $R_H(\tau) = |S_e| - 1$.

terns which are successfully decoded by $s_p(t)$ is bounded as

$$\left|\bigcup_{\tau=0}^{n-1} E_\tau\right| \leq \sum_{\tau=0}^{n-1} |E_\tau| - \frac{2}{n}\sum_{\tau_1,\tau_2} |E_{\tau_1} \cap E_{\tau_2}| \leq n\binom{|S_p|}{|S_e|} - \frac{2}{n}\sum_{\tau_1,\tau_2} \binom{2|S_p| + a(\tau_1,\tau_2) - n}{|S_e|}. \quad (3.12)$$

2) $R_H(\tau) = |S_e| - 1$:

In this case, the index of one erasure symbol is in $\bar{S}_p$ and the indices of the other erasure symbols are in S_p. Thus, the total number of such erasure patterns is $\binom{n-|S_p|}{1}\binom{|S_p|}{|S_e|-1}$, where doubly counted erasure patterns are included. There are two cases of doubly counted erasure patterns as shown in Fig. 3.7.

1) Each of two erasure symbols is located in A_{10} and A_{01}, respectively and the other erasure symbols are located in A_{00}, which are counted as $\binom{n-|S_p|-a(\tau_1,\tau_2)}{1}^2\binom{2|S_p|-n+a(\tau_1,\tau_2)}{|S_e|-2}$.

2) One erasure symbol is located in A_{11} and the other erasure symbols are located in A_{00}, which are counted as $\binom{a(\tau_1,\tau_2)}{1}\binom{2|S_p|+a(\tau_1,\tau_2)-n}{|S_e|-1}$. Similarly, from Lemma 3.1, the number of erasure patterns which are successfully

decoded by $s_p(t)$ is given as

$$\left|\bigcup_{\tau=0}^{n-1} E_\tau\right| \leq \sum_{\tau=0}^{n-1}|E_\tau| - \frac{2}{n}\sum_{\tau_1,\tau_2}|E_{\tau_1}\cap E_{\tau_2}| \leq n\binom{n-|S_p|}{1}\binom{|S_p|}{|S_e|-1} - \frac{2}{n}$$
$$\sum_{\tau_1,\tau_2\in[0,n-1]}\left(\binom{n-|S_p|-a(\tau_1,\tau_2)}{1}^2\binom{2|S_p|+a(\tau_1,\tau_2)-n}{|S_e|-2} + \binom{a(\tau_1,\tau_2)}{1}\binom{2|S_p|+a(\tau_1,\tau_2)-n}{|S_e|-1}\right). \quad (3.13)$$

In order to maximize the upper bounds in (3.12) and (3.13), the second terms of the right hand sides should be minimized, which can be solved by the convex optimization.

The objective functions to be minimized are as follows:

1) For $R_H(\tau)=0$, the objective function is $\sum_{\tau_1,\tau_2}\binom{2|S_p|+a(\tau_1,\tau_2)-n}{|S_e|}$.

2) For $R_H(\tau)=1$, the objective function is

$$\sum_{\tau_1,\tau_2}\binom{n-|S_p|-a(\tau_1,\tau_2)}{1}^2\binom{2|S_p|+a(\tau_1,\tau_2)-n}{|S_e|-2} + \binom{a(\tau_1,\tau_2)}{1}\binom{2|S_p|+a(\tau_1,\tau_2)-n}{|S_e|-1}. \quad (3.14)$$

It is easy to check that the following constraints are used for optimization:

1) For all τ_1 and $\tau_2, 0\leq a(\tau_1,\tau_2)\leq n-|S_p|$.

2) For any τ_2, $\sum_{\tau_1=0}^{n-1} a(\tau_1,\tau_2)=(n-|S_p|)^2$.

3) For any $\tau, a(\tau,\tau)=n-|S_p|$.

4) $|S_e|\leq|S_p|$.

Let $g(x,y)$ be a function defined by

$$g(x,y)=\begin{cases}\prod_{i=0}^{y-1}\frac{x-i}{i+1}, & \text{if } x\geq y+1\\ 0, & \text{otherwise}\end{cases} \quad (3.15)$$

where x and y are real numbers. In fact, I have that $g(x,y) = \binom{x}{y}$ for $x, y \in Z^+$. It is easy to check that $g(x,y)$ is a convex function. First, the objective function for $R_H(\tau) = 0$ is convex because $g(2|S_p| - n + a(\tau_1,\tau_2), |S_e|) = \binom{2|S_p|-n+a(\tau_1,\tau_2)}{|S_e|}$.

At this point, I will prove that the objective function for $R_H(\tau) = 1$ is convex for $\frac{1}{9} < \frac{|S_p|}{n} \leq 1$ and $|S_e| \geq 3$ but it does not mean that the case of $\frac{|S_p|}{n} \leq \frac{1}{9}$ is not a convex. Clearly, the convexity of (3.14) can be proved by the convexity of summands. Then, the summand of (3.14) can be rewritten as

$$a(\tau_1,\tau_2)g(2|S_p| - n + a(\tau_1,\tau_2), |S_e| - 1) + (n - |S_p| - a(\tau_1,\tau_2))^2 g(2|S_p| - n + a(\tau_1,\tau_2), |S_e| - 2). \quad (3.16)$$

Using $g(x,y) = \frac{x-y+1}{y} g(x, y-1)$ for $x \geq y - 1$, (3.16) can be modified as

$$(a(\tau_1,\tau_2)(2|S_p| + a(\tau_1,\tau_2) - n - |S_e| + 2) + (|S_e| - 1)(n - |S_p| - a(\tau_1,\tau_2))^2) g(2|S_p| - n + a(\tau_1,\tau_2), |S_e| - 2). \quad (3.17)$$

The convexity of (3.17) can be proved by its second derivative. Let

$$f(a) = a(\tau_1,\tau_2)(2|S_p| + a(\tau_1,\tau_2) - n - |S_e| + 2) + (|S_e| - 1)(n - |S_p| - a(\tau_1,\tau_2))^2. \quad (3.18)$$

Then, (3.17) can be expressed as the product of f and g. Then the convexity of (3.17) can be proved by deriving the following inequality

$$(fg)'' = f''g + 2f'g' + fg'' \geq 0. \quad (3.19)$$

It is not difficult to derive the s-derivative of $g(x,y)$ in terms of x as

$$g^{(s)}(x,y) = \sum_{S,|S|=s} \prod_{i \in [0,y-1] \setminus [S]} \frac{x-i}{i+1}. \quad (3.20)$$

Using the geometric-harmonic mean inequality

$$(x_1 x_2 ... x_n)^{\frac{1}{n}} \geq \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + ... + \frac{1}{x_n}} \quad (3.21)$$

with $x_i = (x - i + 1)$ and $n = y$, I have

$$(g(x,y))^{\frac{1}{y}} \geq \frac{bg(x,y)}{g'(x,y)} \tag{3.22}$$

$$\frac{y}{(g(x,y))^{\frac{1}{y}}} g(x,y) \leq g'(x,y). \tag{3.23}$$

In general, $g^{(s)}(x,y)$ is the summation of polynomials factored into $y - s + 1$ polynomials of degree one. Using (3.22) and (3.23), (3.17) can be modified as

$$\frac{b-s+1}{(g(a,b-s+1))^{\frac{1}{b-s+1}}} g^{(s-1)}(a,b) \leq g^{(s)}(a,b). \tag{3.24}$$

Using (3.24), I have

$$\begin{aligned}(fg)'' = f''g + 2f'g' + fg'' \geq \\ \frac{(|S_e|-3)^2}{g(2|S_p| - n + a(\tau_1,\tau_2), |S_e|-3)^{\frac{2}{|S_e|-3}}} f + \frac{2(|S_e|-3)}{g(2|S_p| - n + a(\tau_1,\tau_2), |S_e|-3)^{\frac{1}{|S_e|-3}}} f' \\ + f''g \geq \left(\frac{(|S_e|-3)^2}{g(|S_p|, |S_e|-3)^{\frac{2}{|S_e|-3}}} f + \frac{2(|S_e|-3)}{g(|S_p|, |S_e|-3)^{\frac{1}{|S_e|-3}}} f' + f'' \right) g.\end{aligned} \tag{3.25}$$

Let $w = \frac{2(|S_e|-3)}{g(|S_p|,|S_e|-3)^{\frac{1}{|S_e|-3}}}$. Then, it is enough to show that

$$w^2 f + 2wf' + f'' \geq 0. \tag{3.26}$$

It is easy to check that w is an increasing function for $|S_e|$ and $\frac{|S_e|}{|S_p|}$ and a decreasing function for $|S_p|$. Then, left hand side of (3.26) can be rewritten as

$$\begin{aligned}L(a) = w^2 \Big((|S_e| - 1) \left(n - |S_p| - a(\tau_1,\tau_2)\right)^2 + \\ a(\tau_1,\tau_2)(-n + 2|S_p| - |S_e| + a(\tau_1,\tau_2) + 2) \Big) + \\ 2w(-2n|S_e| + n + |S_e|(2|S_p| + 2a(\tau_1,\tau_2) - 1) + 2) + 2|S_e|.\end{aligned} \tag{3.27}$$

At this stage, it is necessary to prove that $L(0) > 0$ and that its discriminant is negative in terms of a. It is easy to check that $L(a)$ is linear in terms of $|S_e|$ with a negative

slope. Thus, $L(a)$ has its minimum value at the maximum value of $|S_e|$. If $|S_e| = |S_p|$, I have

$$L(0) = w^2(|S_p|-1)(n-|S_p|)^2+w(n(2-4|S_p|)+4|S_p|^2-2|S_p|+4)+2|S_p| \geq 0. \tag{3.28}$$

Let $z = \frac{|S_p|}{n}$. Then for sufficiently large values of n and p, (3.28) can be written as

$$\begin{aligned}\frac{L(0)}{n^2w} = w(|S_p|-1)(1-z)^2-4|S_p|+4|S_p|^2 &\geq ((w(|S_p|-1)+4)z - w(|S_p|-1))\,(z-1) \\ &= (w(|S_p|-1)+4)\left(z - \frac{w(|S_p|-1)}{w(|S_p|-1)+4}\right)(z-1). \end{aligned} \tag{3.29}$$

Clearly, (3.29) is positive for a sufficiently large p. Thus, I have $L(0) \geq 0$. Next, the discriminant is written as

$$D = w^4n^2 - 10w^4n|S_p| + 4w^4n + 9w^4|S_p|^2 - 4w^4|S_p| + 4w^4 + 8w^2|S_p|^2 < 0. \tag{3.30}$$

It can also be reduced with sufficiently large values of n and p, whose simplified inequality is given as

$$(9w^4 + 8w^2)z^2 - 10w^4z + w^4 < 0. \tag{3.31}$$

For $\frac{1}{9} < z < 1$, it is easy to derive $D < 0$ for a large value of w. Fig. **??** shows the upper bound of convexity region by (3.30) and (3.31), which shows that the two bounds become identical as $|S_e|$ becomes larger. Thus I prove the convexity of (3.17) for the proposed convexity region.

Using the solution of the optimization program cvx for (3.17), its minimum value occurs at

$$a(\tau_1, \tau_2) = \frac{(n-|S_e|+1)^2 - n - |S_e| + 1}{n-1} \text{ for all } \tau_1 \text{ and } \tau_2, \tag{3.32}$$

which means that the autocorrelation values of $s_p(t)$ are constant. Thus, I prove the theorem.

□

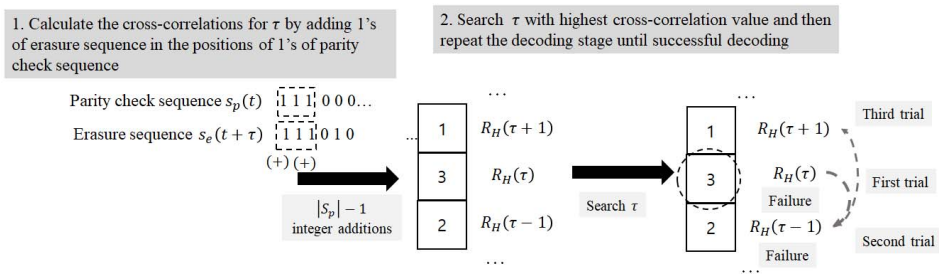

Figure 3.8: Complexity analysis of pre-processing operation for TS-AGD.

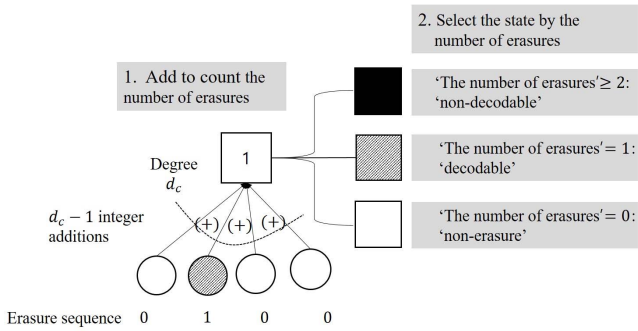

Figure 3.9: Complexity analysis of CNU operation for AGD and TS-AGD.

The third criterion is related to the performance of the decoder, that is, H with the minimum Hamming weight of rows can have better decoding performance in IED as mentioned in [6] as cog, because more erasure symbols are removed in the inner product of the received vector and the rows with the minimum Hamming weight of H.

3.2.4 Analysis of Decoding Complexity of TS-AGD

In the decoding stage, it requires a large number of iterations and high decoding complexity. In the pre-processing stage, TS-AGD derives the order of decoding by computing cross-correlation of the parity check sequence and erasure sequence. TS-AGD decodes in order of more successful decoding cases of cyclic shift values τ but AGD decodes for all the possible τ without considering the decoding order.

I analyze the decoding complexity using integer addition and XOR operation.

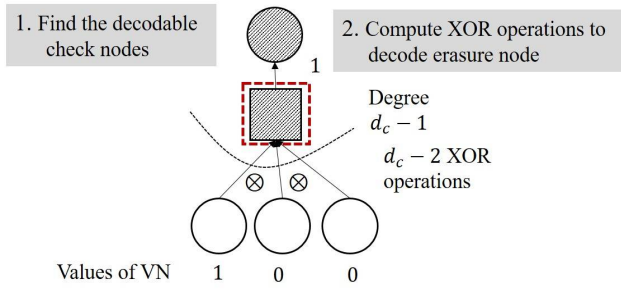

Figure 3.10: Complexity analysis of VNU operation for AGD and TS-AGD.

Complexities of each integer addition and each XOR operation between two binary integer values are considered as 1, respectively and then integer additions and XOR operations among l values are done by $(l-1)$ serialized operations between two integers in the binary form. The complexity of pre-processing stage will be counted by occurrence of integer additions as in Fig. 3.8 and the complexity for search of $\tau's$ with high correlation values is ignored because it has low complexity. For each iteration of decoding process, both CNU and VNU operations are performed, where CNU operation requires $(d_c-1)(n-k)$ integer additions as in Fig. 3.9 and VNU operation requires (d_c-2) XOR for each decodable erasure symbol as in Fig. 3.10. To compute the decoding complexity by the numerical analysis, I consider XOR and integer addition operations as 1, respectively, because complexities of integer addition and XOR operation are equal in the implementation by DSP coding.

3.2.5 Numerical Analysis for Some Cyclic Codes

In this subsection, the proposed TS-AGD is applied to several cyclic codes in the erasure channel, that is, the perfect codes such as Hamming codes, Golay codes, and extended Golay codes, and some triple-error correcting BCH codes. For perfect codes, the proposed modification of parity check matrix can achieve the decoding performance identical to that of the ML decoder, known as *perfect decoding*. For triple-error correcting BCH codes, AGD and TS-AGD have the near-ML decoding performance or

better than regular LDPC code with the similar parameters. Furthermore, the number of iterations of TS-AGD is better than that of AGD. Regarding decoding complexity, additional complexity by pre-processing stage for TS-AGD may increase the decoding complexity but for large codelength, decoding complexity of TS-AGD becomes lower than that of AGD.

a) Proposed Decoding Algorithms for Perfect Codes

1) $(2^m-1, 2^m-1-m, 3)$ Hamming codes: Clearly, Hamming codes have only one cog and thus one row in H is needed to achieve the ML decoding performance as in the following proposition.

Proposition 3.1. *For an (n, k, d) linear code $\mathcal{C}$, the IED of $2^{n-k} \times n$ expanded parity check matrix whose rows consist of all of the codewords of its dual code $\mathcal{C}^{\perp}$ can achieve ML decoding performance.*

Proof. Note that the ML decoder can decode only if S_e of the erasure pattern does not include the support of any codeword. Let H be an $(n-k) \times n$ submatrix with full rank by selecting rows from the expanded parity check matrix. Let H_{S_e} be an $(n-k) \times |S_e|$ submatrix generated by selecting the columns with indices in S_e from H. Let ϵ be an $|S_e|$-tuple erasure vector, that is, ϵ consists of the components with indices in S_e of the transmitted codeword. Then, I have the syndrome of

$$S = H(\mathbf{r}^{(\tau)})^T = H(\mathbf{r}_e^{(\tau)})^T + H(\mathbf{r}_{ne}^{(\tau)})^T = 0, \tag{3.33}$$

which can be modified as

$$H_{S_e}\epsilon^T = H(\mathbf{r}_e^{(\tau)})^T = H(\mathbf{r}_{ne}^{(\tau)})^T. \tag{3.34}$$

If the rank of H_{S_e} is lower than $|S_e|$, (3.34) has multiple solutions, implying that the decoder cannot decode the codeword. Thus, H_{S_e} should have full rank

and then there exist $|S_e|$ linearly independent rows in H_{S_e}. Let H' and H'_{S_e} be $|S_e| \times n$ and $|S_e| \times |S_e|$ matrices constructed from H and H_{S_e} by selecting $|S_e|$ linearly independent rows, respectively. By selecting the components with the same row indices as those of H'_{S_e} from $H(\mathbf{r}_{ne}^{(\tau)})^T$ in (3.34), I can find ϵ by inverting H'_{S_e}. From the properties of linear codes, each row of $H'^{-1}_{S_e} H'$ is actually the codeword of the dual code $\mathcal{C}^{\perp}$. Then, the IED of H whose rows consist of the codewords of $\mathcal{C}^{\perp}$ can correct the erasure patterns, which do not include the codeword. Clearly, this corresponds to the ML decoder. □

From the above proposition, the Hamming codes which have only one cog can achieve the ML decoding performance, because they have the same performance as a $2^{n-k} \times n$ expanded parity check matrix.

2) For the $(23, 12, 7)$ binary Golay code: Using the proposed modification criteria, the parity check matrix of the $(23, 12, 7)$ binary Golay code can be modified as

$$H_m = \begin{pmatrix} 10001000000001100011101 \\ 01001010010000101010001 \\ 00100000001001101010011 \\ 00010010011001100001001 \\ 00001110001000100000111 \\ 00001011000001001001011 \\ 00001000111000000011011 \\ 00000100101010000010111 \\ 00000010001010001011101 \\ 00001000011001011000101 \\ 00000000010000101101111 \end{pmatrix} \tag{3.35}$$

and its parity check sequence is given as

$$s_p(t) = \begin{pmatrix} 11110101100110010100000 \end{pmatrix}, \tag{3.36}$$

Table 3.1: Undecodable erasure patterns by the modified H in the (23,12,7) binary Golay code

Number of erasures	**Total number of erasure patterns**	**TS-AGD and AGD of H_{sys}**	**TS-AGD and AGD of H_m, ML**
≤ 6		0	0
7	245157	253	253
8	490314	4554	4554
9	817190	37973	37950
10	1144066	197754	194810
11	1352078	700488	656558

which corresponds to the characteristic sequence of cyclic difference set with parameters $(23, 12, 5)$. Here, the modified parity check matrix has 12 standard vectors and its rows have the minimum Hamming weights. Thus, (3.35) satisfies the three modification criteria for the parity check matrix. The numerical analysis shows in Table 3.1 that the proposed TS-AGD with H_m can achieve the same performance as the ML decoder and outperform the AGD with H_{sys}, where H_{sys} denotes the systematic form of its parity check matrix defined as $[I_{12}|P]$. Figs. 3.11 and 3.12 show the number of iterations and the decoding complexity of the $(23, 12, 7)$ binary Golay codes. Two decoders, AGD and TS-AGD with H_m and H_{sys} are considered in Figs. 3.11 and 3.12 . The number of iterations of the proposed TS-AGD algorithm can be reduced compared to the AGD as shown in Fig. 3.11 The decoding complexity of TS-AGD with H_m is lower than that of AGD for the low erasure probability, where most of $R_H(\tau)$ by pre-processing stage are higher than $|S_e| - 2$ as in Fig. 3.12. However, decoding complexity of TS-AGD with H_m is higher for the other cases because many cases of $R_H(\tau)$ by pre-processing stage are lower than $|S_e| - 2$ and they cannot be decoded in few iterations.

3) $(24, 12, 8)$ binary extended Golay code: In fact, while this is not a cyclic code, it

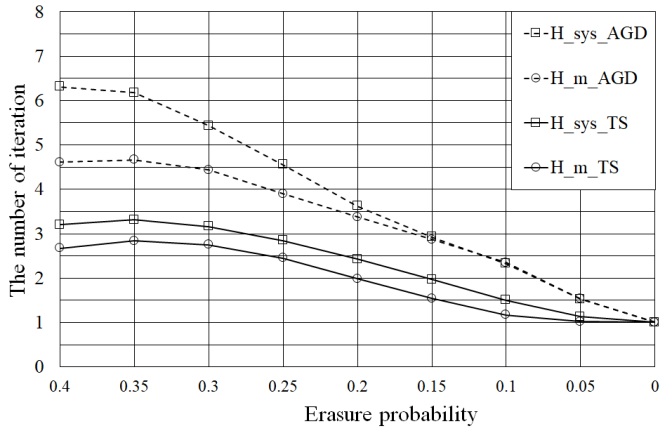

Figure 3.11: Average number of iterations of AGD and TS-AGD with modified H for the $(23, 12, 7)$ Golay code.

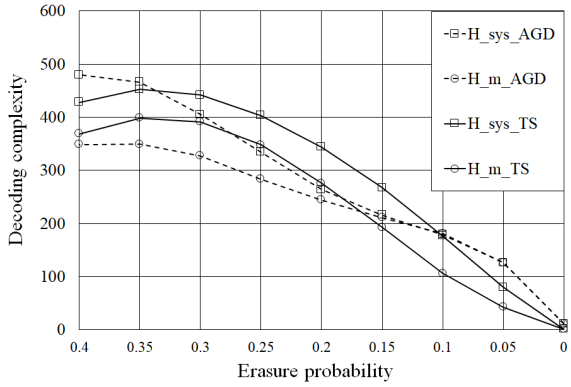

Figure 3.12: Decoding complexity of AGD and TS-AGD with modified H for the $(23, 12, 7)$ Golay code.

is cyclic except for the last parity bit. Thus, I can apply the AGD algorithm and the proposed TS-AGD algorithm. Hehn modified the parity check matrix as [6]

$$H_{Hehn} = \begin{pmatrix} 111000001001100000100001 \\ 110000100001001110000001 \\ 110100101010010000000001 \\ 111000110000000000010101 \\ 110001000101010000000101 \\ 110100010100000010100001 \\ 011000100001010001001001 \\ 110110000001000000011001 \\ 111101000000001001000001 \\ 110010000010001000100101 \\ 001100101001001000010001 \\ 001101010011001000000010 \end{pmatrix} \tag{3.37}$$

and the systematic form of the parity check matrix is given as

$$H_{sys} = \begin{pmatrix} 100000000000110111000101 \\ 010000000000101110001011 \\ 001000000000011100010111 \\ 000100000000111000100101 \\ 000010000000110001011011 \\ 000001000000100010111111 \\ 000000100000000101100111 \\ 000000010000001011011101 \\ 000000001000010110111001 \\ 000000000100101101111001 \\ 000000000010011011100011 \\ 000000000001111111111110 \end{pmatrix}. \tag{3.38}$$

The modified parity check matrix based on the three proposed criteria can be given as

$$H_m = \begin{pmatrix} 10001000000001100011101 0 \\ 01001010010000101010001 0 \\ 00100000001001101010011 0 \\ 00010010011001100001001 0 \\ 00001110001000100000111 0 \\ 00001011000001001001011 0 \\ 00001000111000000011011 0 \\ 00000010010101000010111 0 \\ 00000010001010001011101 0 \\ 00001000011001011000101 0 \\ 00000000010000101101111 0 \\ 00001010011001101011110 1 \end{pmatrix} \tag{3.39}$$

where the first 11 standard column vector indices are determined by the cyclic difference set with parameters $(23, 12, 5)$ as before and the last standard vector is located in the extended bit. The last row of H_m has the Hamming weight of 12, which is larger than the minimum Hamming weight 8. Thus, I can further

modify it by replacing the last row by sum of the first row and the last row as

$$H_A = \begin{pmatrix} 100010000000011000111010 \\ 010010100100001010100010 \\ 001000000010011010100110 \\ 000100100110011000010010 \\ 000011100010001000001110 \\ 000010110000010010010110 \\ 000010001110000000110110 \\ 000000100101010000101110 \\ 000000100010100010111010 \\ 000010000110010110001010 \\ 000000000100001011011110 \\ 100000100110000010000111 \end{pmatrix} \tag{3.40}$$

where the last row has the minimum Hamming weight 8 but the first column is not a standard vector. The further modification is done by replacing the i-th row with the sum of the i-th row and the last row of H_m, $1 \leq i \leq 11$ and the last row with the first row of H_m as

$$H_B = \begin{pmatrix} 100000100110000010000111 \\ 010000000010010000011111 \\ 001010100100000000011011 \\ 000110000000000010101111 \\ 000001000100010010110011 \\ 000000010110001000101011 \\ 000000101000011010001011 \\ 000010000011001010010011 \\ 000010000100111000000111 \\ 000000100000001100110111 \\ 000010100010010001100011 \\ 100010000000011000111010 \end{pmatrix}. \tag{3.41}$$

In fact, the first columns of H_A and H_B have Hamming weight 2. Then the parity check sequences of H_p, H_A, and H_B are given as

$$s_{p,H_m}(t) = \begin{pmatrix} 111101011001100101000001 \end{pmatrix} \tag{3.42}$$

$$s_{p,H_A}(t) = \begin{pmatrix} 011101011001100101000001 \end{pmatrix} \tag{3.43}$$

$$s_{p,H_B}(t) = \begin{pmatrix} 011101011001100101000000 \end{pmatrix}. \tag{3.44}$$

In the $(24, 12, 8)$ extended Golay code, any of the modified parity check matrices cannot achieve the same performance as that of the ML decoder. However, the TS-AGD by adding redundant check equations to H_B can give us the same decoding performance as the ML decoder, which is given as

$$H_C = \begin{pmatrix} H_B \\ H_A^{'} \end{pmatrix} \tag{3.45}$$

where $H_A^{'}$ is a submatrix composed of nine rows out of the first 11 rows of H_A. Fig. 3.13 shows the relationship among the various modified parity check matrices. Table 3.2 shows the decoding performance of the proposed TS-AGD and

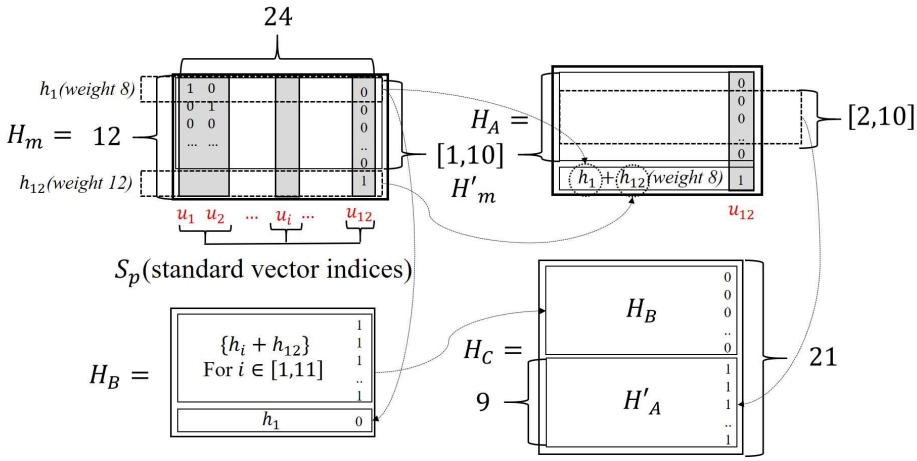

Figure 3.13: Modifications of the parity check matrix in the $(24, 12, 8)$ extended binary Golay code.

AGD with H_{sys}, H_m, H_A, H_B, H_{Hehn}, and H_C, where H_C shows decoding performance identical to that of the ML decoder and better decoding performance than the decoding algorithm by Hehn. In AGD, difference of the number of iterations by H's are small as in Fig. 3).

However, decoding complexity of the modified H's is lower than H_{Hehn} except H_c as in Fig. 3.15, where H_c has the smallest number of iterations but highest decoding complexity due to additional rows of the parity check matrix. For the number of iterations of TS-AGD, H_m and H_A are lower than others because $|S_p| = n - k$, which is larger than the others.

For decoding complexity of TS-AGD, similar tendency of the previous case of $(23, 12, 7)$ Golay codes is shown, where that of TS-AGD is lower than that of AGD for low erasure probabilities. Among the modified H's of TS-AGD, the decoding complexity is proportional to the number of iterations except H_C.

4) Ternary $(11, 6, 5)$ Golay code: For the ternary $(11, 6, 5)$ Golay code, AGD and TS-AGD algorithms with the systematic and the modified form of their parity

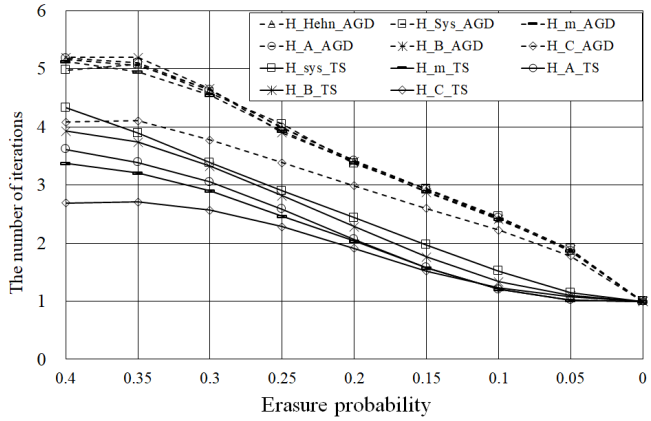

Figure 3.14: Average number of iterations of AGD and TS-AGD with modified H for the $(24, 12, 8)$ extended Golay code.

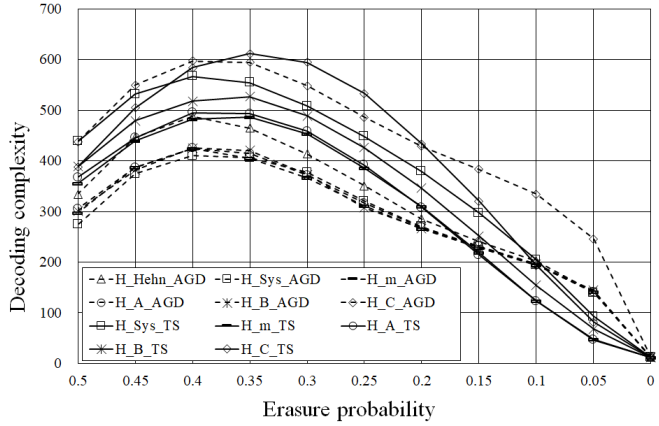

Figure 3.15: Decoding complexity of AGD and TS-AGD with modified H for the $(24, 12, 8)$ extended Golay code.

Table 3.2: Undecodable erasure patterns by the modified H for the $(24, 12, 8)$ binary extended Golay code

Number of erasures	**Total number of erasure patterns**	**TS-AGD and AGD of H_{sys}**	**TS-AGD and AGD of H_m**	**TS-AGD and AGD of H_A**	**TS-AGD and AGD of H_B and H_{Hehn}**	**TS-AGD and AGD of H_C and ML**
≤ 7		0	0	0	0	0
8	735471	759	759	759	759	759
9	1307504	12144	12144	12144	12144	12144
10	1961256	92000	91080	91080	91080	91080
11	2496144	460253	426581	425178	425040	425040
12	2704156	1515792	1344005	1325536	1322179	1313116

check matrices can achieve the ML decoding performance, whereas the number of iterations and the decoding complexity of the modified form are better than those of the systematic form. The systematic parity check matrix of the ternary $(11, 6, 5)$ Golay code is given as

$$H_{sys} = \begin{pmatrix} 10000122210 \\ 01000012221 \\ 00100212012 \\ 00010110111 \\ 00001222101 \end{pmatrix}. \tag{3.46}$$

The modified form of the parity check matrix uses the parity check sequence constructed by the characteristic sequence of the cyclic difference set with pa-

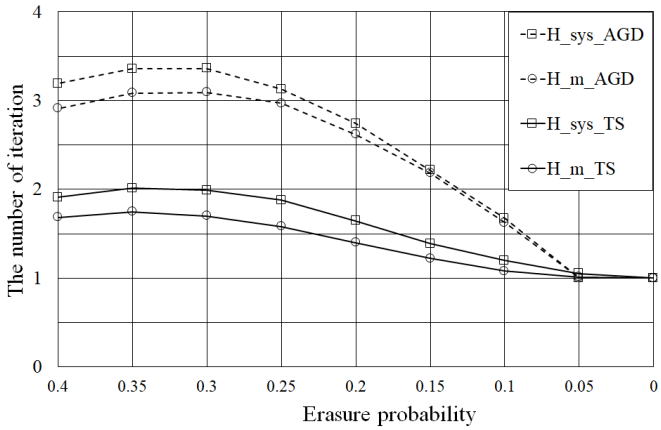

Figure 3.16: Average number of iterations of AGD and TS-AGD with modified H for the $(11, 6, 5)$ ternary Golay code.

rameters $(11, 6, 3)$ as

$$H_m = \begin{pmatrix} 12000110022 \\ 00100212012 \\ 01010122002 \\ 01001201022 \\ 02000021112 \end{pmatrix} \tag{3.47}$$

where the parity check sequence is given as

$$s_{p,H_m}(t) = \begin{pmatrix} 10111000100 \end{pmatrix}. \tag{3.48}$$

The number of iterations and the decoding complexity of the ternary Golay codes are described in Figs. 3.16 and 3.17, which shows that those of TS-AGD has lower than those of AGD. By numerical analysis, there are no undecodable erasure patterns for the number of erasures $e < 6$, there are 66 undecodable erasure patterns for $e = 6$, and there are no decodable erasure patterns for $e > 6$ for the modified form and the ML decoders.

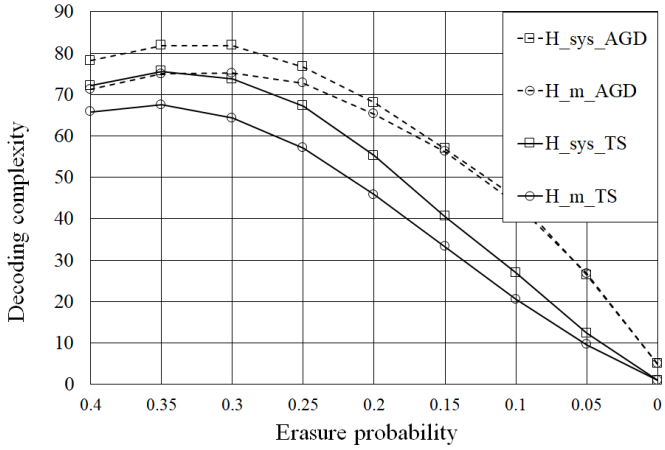

Figure 3.17: Decoding complexity of AGD and TS-AGD with modified H for the $(11, 6, 5)$ ternary Golay code.

b) Proposed TS-AGD Algorithms for Binary Primitive BCH Codes

Binary primitive BCH codes are widely used due to their low-complexity encoding, large designed distance, and guaranteed decoding performance for certain number of erasures. However, BCH codes require inherently high decoding complexity and their decoding performance is degraded for large n and k. The proposed TS-AGD can overcome the disadvantages of BCH codes by the low-complexity decoding with improved performance compared to AGD. Here, the proposed TS-AGD for the triple-error correcting $(63, 45, 5), (255, 231, 5)$, and $(1023, 993, 5)$ BCH codes is numerically analyzed in the erasure channel.

In general, $s_p(t)$ of the BCH code is generated by the cyclic difference set but there are some cases that the cyclic difference set does not exist for the parameters of the BCH code. Instead, S_p can be constructed using the union of cyclotomic cosets of the finite field as an alternative construction method. In this case, $s_p(t)$ does not have constant autocorrelation but has relatively low values of autocorrelation. Thus, this construction method of $s_p(t)$ also results in good decoding performance.

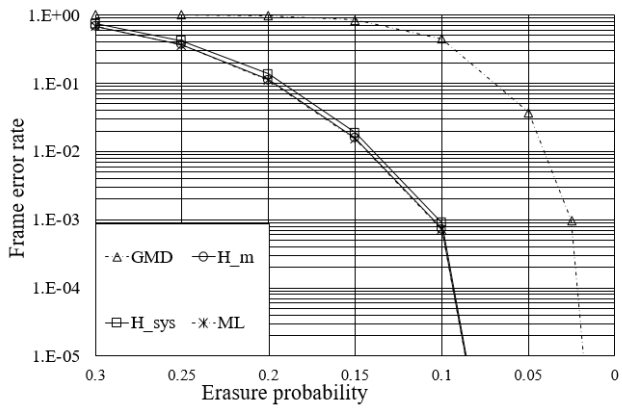

Figure 3.18: Frame error rate of $(63, 45)$ BCH code by GMD, AGD, and TS-AGD for H_m and H_{sys}, and ML.

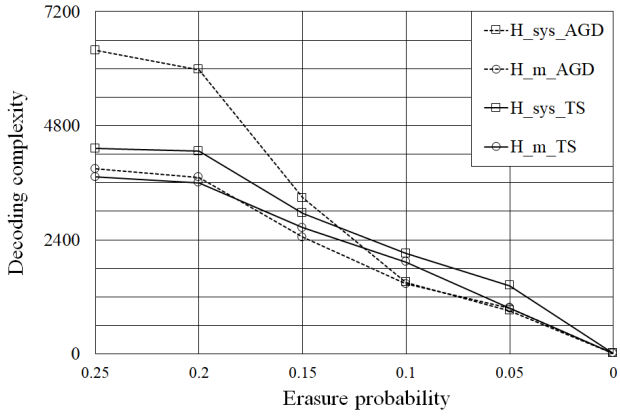

Figure 3.19: Decoding complexity of AGD and TS-AGD for $(63, 45)$ BCH code.

1) $(63, 45, 7)$ BCH code:

For $(63, 45)$ BCH code, the proposed modification of H is done using the cyclotomic cosets of the coset leaders of the finite field F_{2^6} in $\{\alpha, \alpha^3, \alpha^{11}\}$, where α is a primitive element of F_{2^6}. For numerical analysis, H_m has better decoding performance than H_{sys}, but they have near-ML decoding performance as in Fig. 3.18. However, decoding complexity of TS-AGD is slightly higher than that of AGD in some range of erasure probability as in Fig. 3.19 due to additional

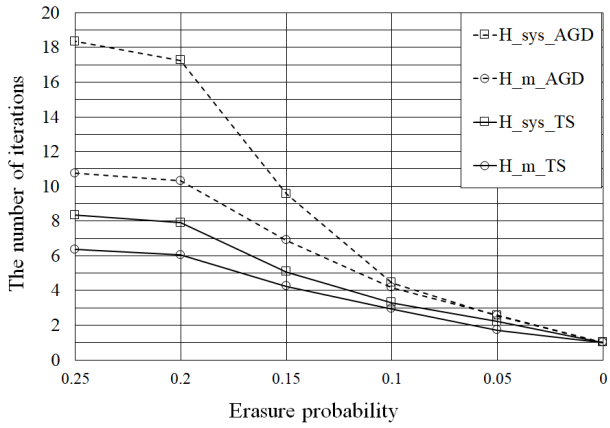

Figure 3.20: Number of iterations of AGD and TS-AGD for $(63, 45)$ BCH code.

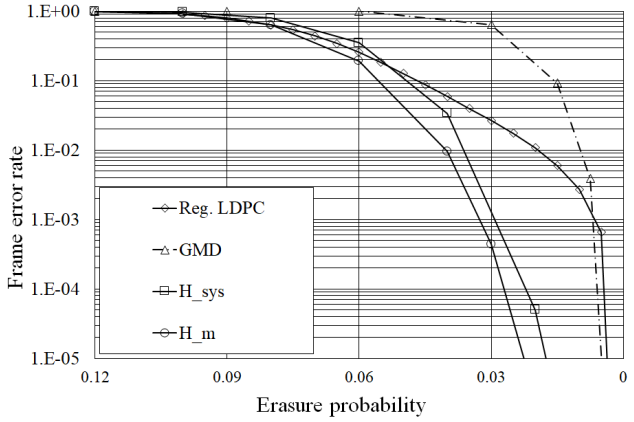

Figure 3.21: Frame error rate of $(255, 231)$ BCH code by GMD, AGD, and TS-AGD for H_m and H_{sys}, and $(260, 234)$ regular LDPC code with $d_v = 3$ by IED.

decoding complexity by pre-processing, whereas the number of iterations in TS-AGD is always lower than that in AGD as in Fig. 3.20 due to the pre-processing stage for $(63, 45)$ BCH code.

2) $(255, 231, 7)$ BCH code:

For $(255, 231)$ BCH code, the proposed modification of H is done using the cyclotomic cosets of the coset leaders of the finite field F_{2^8} in $\{\alpha^3, \alpha^5, \alpha^{11}\}$,

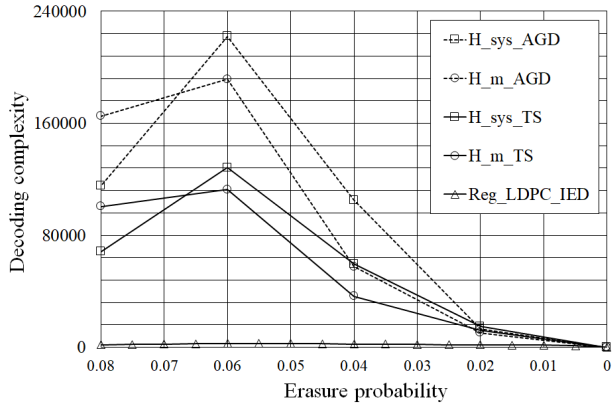

Figure 3.22: Decoding complexity of AGD and TS-AGD for $(255, 231)$ BCH code and IED for $(260, 234)$ regular LDPC code with $d_v = 3$.

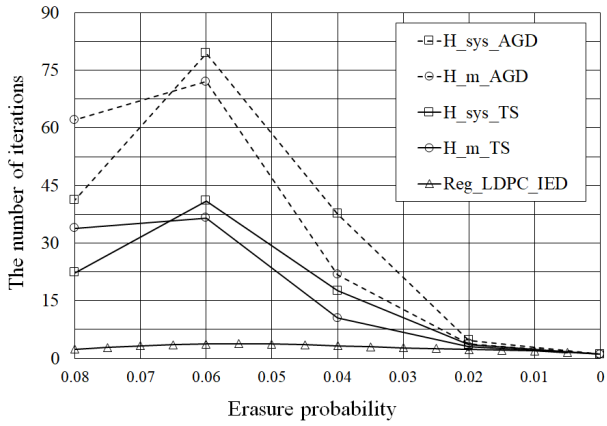

Figure 3.23: Number of iterations of AGD and TS-AGD for $(255, 231)$ BCH code and IED for $(260, 234)$ regular LDPC code with $d_v = 3$.

where α is a primitive element of F_{2^8}. For numerical analysis, H_m has better decoding performance than H_{sys} and $(260, 234)$ regular LDPC codes with $d_v = 3$ as in Fig. 3.21. Decoding complexity and the number of iterations of TS-AGD are lower than those of AGD as in Figs. 3.22 and 3.23, but they are higher than those of IED of regular LDPC code. The reason is that additional decoding complexity by pre-processing stage becomes negligible for large codelength,

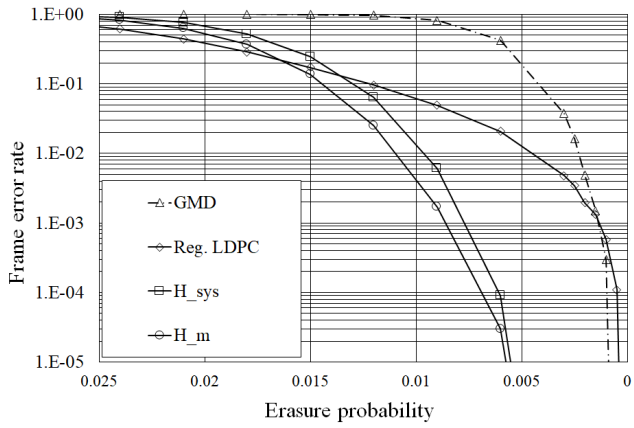

Figure 3.24: Frame error rate of $(1023, 993)$ BCH code by GMD, AGD and TS-AGD for H_m and H_{sys}, and $(1020, 984)$ regular LDPC codes with $d_v = 3$ by IED.

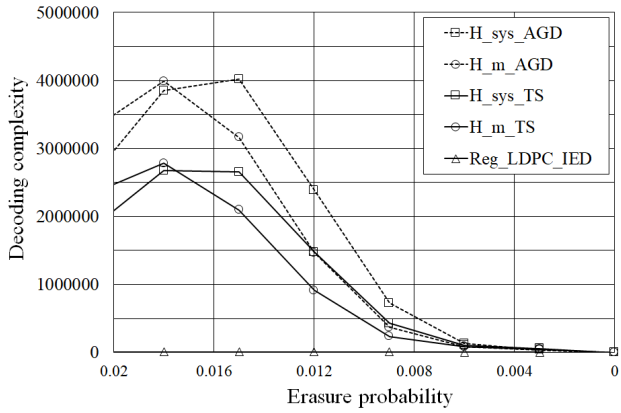

Figure 3.25: Decoding complexity of AGD and TS-AGD for $(1023, 993)$ BCH code and IED for $(1020, 984)$ regular LDPC code with $d_v = 3$.

where the number of correctable erasure patterns is exponentially grown.

3) $(1023, 993, 7)$ BCH code:

For $(1023, 993)$ BCH code, the proposed modification of H is done using the cyclotomic cosets of the coset leaders of the finite field $F_{2^{10}}$ in $\{\alpha^7, \alpha^{11}, \alpha^{13}\}$, where α is primitive element of $F_{2^{10}}$. H_m has better decoding performance than

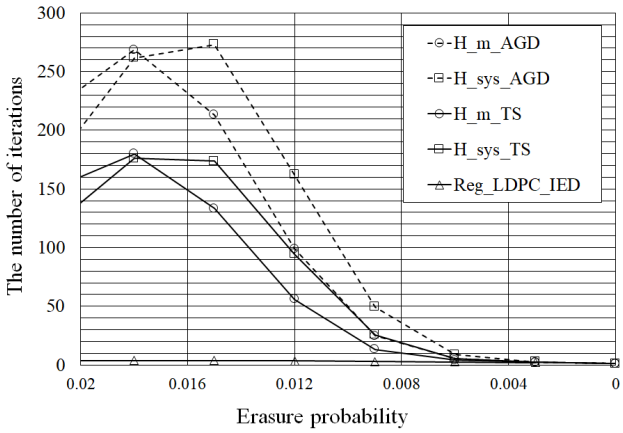

Figure 3.26: Number of iterations of AGD and TS-AGD for $(1023, 993)$ BCH code and IED for $(1020, 984)$ regular LDPC code with $d_v = 3$.

H_{sys} and $(1020, 984)$ regular LDPC code with $d_v = 3$ as in Fig. 3.24. Decoding complexity and the number of iterations of TS-AGD are lower than those of AGD as in Figs. 3.25 and 3.26, but they are higher than those of IED of regular LDPC code. Decoding performance and complexity of $(1023, 993)$ BCH code are shown in the same tendency as those of $(255, 231)$ BCH code.

In short, AGD and TS-AGD have the near-ML decoding performance or better than regular LDPC code with the similar parameters. Furthermore, the number of iterations of TS-AGD is better than that of AGD. Regarding decoding complexity, additional complexity by pre-processing stage for TS-AGD may increase the decoding complexity as in Fig. 3.19, but for large codelength, decoding complexity of TS-AGD becomes lower than that of AGD.

3.3 Construction of Parity Check Matrix and TS-AGD for Cyclic MDS Codes

In this section, the proposed TS-AGD is applied to cyclic MDS codes. In order to achieve the perfect decoding, stopping redundancy and submatrix inversion are also used for the TS-AGD of cyclic MDS codes.

3.3.1 Modification of Parity Check Matrix for Cyclic MDS Codes

The criteria for the modification of the parity check matrices in Section 3.2 can be simplified for the TS-AGD of cyclic MDS codes from the properties of the MDS codes.

Proposition 3.2 (The first and third criteria for cyclic MDS codes)**.** *For the parity check matrix of the* (n, k) *MDS codes,* $n - k$ *standard vectors can be made in any columns of the parity check matrix and the Hamming weight of all rows is* $k+1$*, which is the minimum Hamming weight of its dual codes.*

Proof. It can be easily proved from the theorems in Section 2 of Chapter 11 in [27]. □

Thus, the first and third criteria can always be satisfied in the parity check matrix of the MDS codes but for the second criterion, I have to make the magnitude of the Hamming autocorrelation of the parity check sequence as low as possible.

In order to improve the decoding performance of AGD and IED, the expanded parity check matrix is proposed by expanding the rows of the parity check matrix. That is, the $(n-k) \times n$ parity check matrix can be expanded to a $b(n-k) \times n$ matrix, which is composed of b distinct parity check matrices. Note that each $(n-k) \times n$ parity check matrix has its own parity sequence. Then, the TS-AGD using the expanded parity check matrix decodes the received vector by the first $(n-k) \times n$ parity check matrix. If it fails, successful decoding is possible using the subsequent parity check matrices. Note that if the perfect decoding is possible by the expanded parity check

matrix, the number of rows in the expanded parity check matrix is called the *stopping redundancy*.

3.3.2 Proposed TS-AGD for Cyclic MDS Codes

a) TS-AGD Algorithm for Cyclic MDS Codes

The procedure of the proposed TS-AGD for MDS codes is nearly identical to that of the binary codes introduced in the previous section but the detailed decoding procedure is slightly different. For the binary codes in Fig. 3.3, there is a case that the erasure symbols in the non-standard part cannot be successfully decoded at the first iteration for $R_H(\tau) = |S_e| - 1$. Unlike the binary codes, TS-AGD for the MDS codes can always successfully decode the cyclically shifted received vectors with τ such that $R_H(\tau) \geq |S_e| - 1$ because the non-standard column vectors of the parity check matrix always consist of nonzero components. Therefore, the maximum number of iterations is reduced to 2 if there exists τ which meets the condition of $R_H(\tau) \geq |S_e| - 1$. However, the proposed TS-AGD cannot decode the received vectors of the cyclic MDS codes for the cases of $R_H(\tau) \leq |S_e| - 2$.

b) Performance Analysis of Cyclic MDS Codes and LRCs

For (n, k) cyclic MDS codes, their minimum distance is the largest value $n - k + 1$, which means that the best ML decoding performance of the MDS codes can be obtained in the erasure channel. However, since the minimum Hamming weight of rows in the parity check matrix of the MDS codes is the largest value $k + 1$, this degrades the decoding performance for AGD or IED compared to the binary codes due to the third modification criterion of the parity check matrix.

In order to mitigate the degradation of the decoding performance due to the third criterion without expansion of the parity check matrix, I can also consider cyclic locally repairable codes (LRCs) [15], which can be constructed by slightly modifying the MDS codes as follows. LRC is originally used to reduce the decoding complexity of

the repair process in distributed storage systems. LRCs have slightly shorter minimum Hamming distances than MDS codes, which reduces the decoding performance gap between AGD and the ML decoder. In this subsection, the proposed TS-AGD decoding algorithm can be applied to LRCs as well as cyclic MDS codes in order to achieve the ML decoding performance. For $(d_L^{\perp}-1)|k$ and $d_L^{\perp}|n$, the generator polynomial of the optimal cyclic LRC is given as

$$g(x) = \prod_{i \in \{L \cup M\}} (x - \alpha^i) \tag{3.49}$$

where $d_L^{\perp}$ denotes Hamming distance of the dual code, $L = \{l | l \bmod d_L^{\perp} = 0\}$, and $M = \{0, 1, 2, ..., n - \frac{k}{d_L^{\perp}-1} d_L^{\perp}\}$. For the code parameters $(n, k) = (15, 8)$, there exist a $(15, 8, 8)$ MDS code, a $(15, 8, 7)$ cyclic LRC with $d_{min}^{\perp} = 5$, and a $(15, 8, 5)$ cyclic LRC with $d_{min}^{\perp} = 3$. From (3.49), the generator polynomial of the $(15, 8, 7)$ cyclic LRC has the zeros $\{1, \alpha^1, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^{10}\}$. Similarly, the generator polynomial of the $(15, 8, 5)$ cyclic LRC has the zeros $\{1, \alpha^1, \alpha^2, \alpha^3, \alpha^6, \alpha^9, \alpha^{12}\}$. The characteristic sequence of the cyclic difference set with parameters $(15, 8, 4)$, that is, a bit-inverted m-sequence of length 15 can be used for the parity check sequence as

$$s_p(t) = (111011001010000). \tag{3.50}$$

Then, the corresponding masks A of the parity check matrices of the $(15, 8, 8)$ MDS code, and the $(15, 8, 7)$ and $(15, 8, 5)$ cyclic LRCs are given as

$$A_{MDS} = \begin{pmatrix} 100100110101111 \\ 010100110101111 \\ 001100110101111 \\ 000110110101111 \\ 000101110101111 \\ 000100111101111 \\ 000100110111111 \end{pmatrix} \tag{3.51}$$

Table 3.3: Undecodable erasure patterns for the $(15, 8)$ cyclic MDS code and LRCs

Number of erasures	**Total number of erasure patterns**	**TS-AGD and AGD with** H_{sys}	**TS-AGD and AGD with** H_{MDS}	**TS-AGD, AGD, and ML with** $H_{LRC_{(15,8,5)}}$	**TS-AGD and AGD with** $H_{LRC_{(15,8,7)}}$	**ML with** $H_{LRC_{(15,8,7)}}$
≤ 3		0	0	0	0	0
4	1365	90	0	0	0	0
5	3003	1128	168	60	3	0
6	5005	3520	2380	820	400	0
7	6435	5820	5680	3600	3570	405

$$A_{LRC_{(15,8,7)}} = \begin{pmatrix} 100100100100100 \\ 010100110101111 \\ 001100010101111 \\ 000110110101111 \\ 000101110101111 \\ 000100111101111 \\ 000100110111111 \end{pmatrix} \tag{3.52}$$

$$A_{LRC_{(15,8,5)}} = \begin{pmatrix} 100100110101111 \\ 010000100001000 \\ 001000010000100 \\ 000010000100001 \\ 000101110101111 \\ 000100001000010 \\ 000100110111111 \end{pmatrix}. \tag{3.53}$$

The erasure decoding performance of the above three $(15, 8)$ codes is shown in Table 3.3. Clearly, the ML decoding performance of LRC with lower $d_L^{\perp}$ is degraded compared to that of MDS codes but the performance gap between TS-AGD and the ML decoder becomes smaller. For the $(15, 8, 5)$ cyclic LRC, TS-AGD performs the

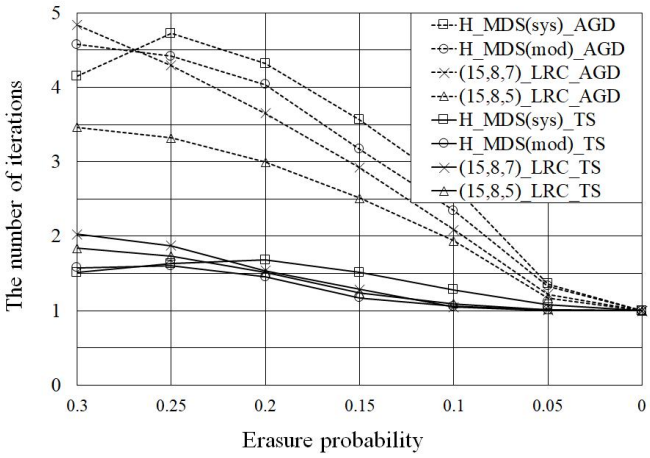

Figure 3.27: Average number of iterations of the $(15, 8)$ cyclic MDS code and cyclic LRCs.

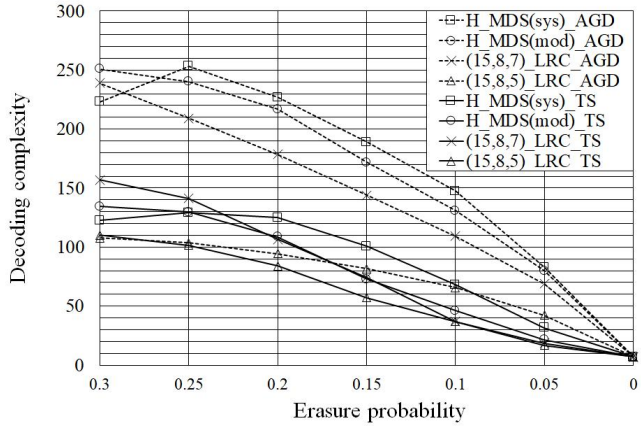

Figure 3.28: Decoding complexity of the $(15, 8)$ cyclic MDS code and cyclic LRCs.

perfect decoding, that is, there is no difference in the decoding performance between TS-AGD and the ML decoder. TS-AGD has the best decoding performance for the $(15, 8, 7)$ cyclic LRC which can replace the $(15, 8, 8)$ MDS code. Figs. 3.27 and 3.28 show that TS-AGD has lower decoding complexity and the fewer iterations than those of AGD for the MDS code and LRCs. For TS-AGD, MDS codes have lower decoding complexity than LRCs because MDS codes use the TS-AGD algorithm. That is, TS-AGD decoding for LRCs is not always successful when $R_H(\tau) = |S_e| - 1$, because LRC is not an MDS code and some components of the non-standard column vectors of the parity check matrix of LRC are zero.

Instead of mitigating the strict condition of the cyclic MDS codes by the cyclic LRCs, the expanded parity check matrix can be used to enhance the erasure decoding performance of the cyclic MDS codes. Numerical analysis of the expanded parity check matrix using m-sequences is introduced in the following subsection.

c) Performance Analysis of TS-AGD with Expanded Parity Check Matrix for Cyclic MDS Codes

To analyze the erasure decoding performance of TS-AGD with expanded parity check matrix for cyclic MDS codes, it is necessary to know the Hamming auto- and cross-correlations of the parity check sequences of the expanded parity check matrix. By counting the number of decodable $(n-k)$-erasure patterns by Lemma 3.1, the decoding performance of the TS-AGD with expanded parity check matrix can be estimated. Each term of the expanded parity check matrix in (3.10) can be modified as in the following proposition and theorems.

Proposition 3.3 (The first term in the Bonferroni inequality in the expanded parity check matrix)**.** *The first term in (3.10) is modified in the expanded parity check matrix as follows:*

$$\sum_{I \subset A, |I|=1} |E_i| = bn(k(n-k)+1). \tag{3.54}$$

Proof. Suppose that the expanded parity check matrix has b parity check sequences. The τ-shifted parity check sequence $s_p(t+\tau)$ can correct $n-k$ erasure symbols in the following two cases:

1) $R_H(\tau) = |S_e|$: $n-k$ erasure symbols are located in the $n-k$ standard vector indices and the decoder can correct the $\binom{n-k}{n-k} = 1$ erasure pattern.

2) $R_H(\tau) = |S_e| - 1$: $n-k-1$ erasure symbols are located in the standard vector indices and the decoder can correct the $\binom{n-k}{n-k-1}\binom{k}{1} = (n-k)k$ erasure patterns.

Each parity check sequence of the expanded parity check matrix has up to n cyclically equivalent parity check sequences and therefore, it can correct up to $n(k(n-k)+1)$ erasure patterns. □

Theorem 3.2 (The second term in the Bonferroni inequality in the expanded parity check matrix)**.** *The second term in (3.10) can also be modified in the expanded parity check matrix as*

$$\sum_{I\subset V,|I|=2}\left|\bigcap_{i\in I}E_i\right| = \sum_{\tau_1,\tau_2=0}^{n-1}\sum_{1\le i<j\le b} \Big(4F_{s_{p,i}(t+\tau_1),s_{p,j}(t+\tau_2)}(k-2) + nF_{s_{p,i}(t+\tau_1),s_{p,j}(t+\tau_2)}(k-1)\Big) \quad (3.55)$$

where $F_{s_{p,i}(t+\tau_1),s_{p,j}(t+\tau_2)}(\gamma)$ returns 1 if $\sum_{t=0}^{n-1}(s_{p,i}(t+\tau_1)s_{p,j}(t+\tau_2)) = \gamma$ and 0, otherwise.

Proof. The proof is similar to that of Theorem 3.1. For the i-th and j-th cyclically shifted parity check sequences, the number of doubly counted decodable erasure patterns is expressed as

$$\sum_{I\subset V,|I|=2}\left|\bigcap_{i\in I}E_i\right| = \sum_{\tau_1,\tau_2=0}^{n-1}\sum_{1\le i<j\le b}\sum_{\gamma=0}^{k} c_\gamma F_{s_{p,i}(t+\tau_1),s_{p,j}(t+\tau_2)}(\gamma) \quad (3.56)$$

where c_γ is the number of doubly counted decodable erasure patterns from $s_{p,i}(t+\tau_1)$ and $s_{p,j}(t+\tau_2)$. Equation (3.56) partitions the number of doubly counted decodable

erasure patterns by τ_1, τ_2, $s_{p_i}(t)$, $s_{p_j}(t)$, and γ. The remaining problem is to determine c_γ. For the given τ, $s_{p_i}(t)$, and $s_{p_j}(t)$, the doubly counted decodable erasure patterns can be computed as follows:

1) If $\gamma \leq k-3$: A doubly counted decodable erasure pattern does not occur because there should be at most $n-k-3$ erasure symbols in A_{00} and the remaining three erasure symbols cannot be decoded regardless of their locations of A_{01}, A_{10}, and A_{11}, where the Hamming cross-correlation values of one parity check sequence and the erasure sequence are smaller than or equal to $k-3$.

2) If $\gamma = k-2$: I have $|A_{11}| = k-2$, $|A_{10}| = |A_{01}| = 2$, and $|A_{00}| = n-k-2$. Then, doubly counted decodable erasure patterns occur when one erasure symbol is located in A_{01}, one erasure is in A_{10}, and $n-k-2$ erasure symbols are in A_{00}. Therefore, c_γ is 4.

3) If $\gamma = k-1$: I have $|A_{11}| = k-1$, $|A_{10}| = |A_{01}| = 1$, and $|A_{00}| = n-k-1$. Then, the doubly counted decodable erasure patterns can occur when one erasure symbol is located in A_{01}, one erasure symbol is in A_{10}, and $n-k-2$ erasure symbols are in A_{00}, where $c_\gamma = n-k-1$. In addition, doubly counted decodable erasure patterns occur when one erasure symbol is located in A_{11} and the other $n-k-1$ erasure symbols are in A_{00}, A_{01}, or A_{10}, where $c_\gamma = k+1$. The sum of the two cases gives us $c_\gamma = n$.

Thus, the theorem is proved. □

The distribution of the Hamming auto- and cross-correlation values of the parity check sequences can be used to count the first and the second terms in the Bonferroni inequality by Proposition 3.3 and Theorem 3.2. The Hamming auto- and cross-correlations of pseudorandom sequences, especially the m-sequences of length $n = 2^m - 1$, have been researched. There can be used to analyze the erasure decoding performance of TS-AGD. In this subsection, TS-AGD with the expanded parity check

matrix for $(n, \frac{n+1}{2})$ MDS codes is analyzed, where the parity check sequences for $(n-k) \times n$ parity check matrices are constructed using the bit-inverted m-sequence and its decimated sequences. In fact, it is well known that the m-sequences correspond to the characteristic sequences of the cyclic difference sets.

1) $m \leq 3$: For $m = 3$, only one $(n-k) \times n$ parity check matrix with a parity check sequence constructed by the bit-inverted m-sequence of length 7 can achieve the perfect decoding. It can be easily shown by the numerical analysis.

2) $m = 4$: There are two bit-inverted m-sequences, $s_{p_1}(t)$ and $s_{p_2}(t)$ of length $n = 15$. The distribution of their Hamming cross-correlation values for $\tau \in [0, n-1]$ can be given as [17]

$$\sum_{t=0}^{n-1} s_{p_1}(t) s_{p_2}(t+\tau) = \begin{cases} 3, & 4 \text{ times} \\ 4, & 5 \text{ times} \\ 5, & 4 \text{ times} \\ 6 & 2 \text{ times.} \end{cases} \tag{3.57}$$

In this case, I can derive the number of the decodable 8-erasure patterns for the expanded parity check matrix with $b = 2$ by the inclusion-exclusion principle. For the first term in (3.55), the number of doubly counted erasure patterns is computed as $2 \times 5(7 \times 8 + 1) = 1710$. For the second term, it is given as

$$\begin{aligned} \sum_{I \subset V, |I|=2} \left| \bigcap_{i \in I} E_i \right| &= \sum_{\tau=0}^{14} \sum_{1 \leq i < j \leq 2} \left(4F_{s_{p,i}(t), s_{p,j}(t+\tau)}(5) + 15F_{s_{p,i}(t), s_{p,j}(t+\tau)}(6) \right) \\ &= \sum_{\tau=0}^{14} (4F_{s_{p,i}(t), s_{p,j}(t+\tau)}(5)) = 15 \times 4 \times 2 = 120, \end{aligned} \tag{3.58}$$

which makes at least 1590 decodable erasure patterns and it is the exact value because it has no triply or more counted erasure pattern. Note that the total number of 8-erasure patterns is $\binom{15}{8} = 6435$.

Table 3.4: Hamming cross-correlation distribution of m-sequence of length 31 and its decimated sequences

Decimation	The number of Hamming cross-correlations (values)	Types
3	3(6,8,10)	Gold [18], Kasami [19]
5	3(6,8,10)	Gold [18]
7	3(6,8,10)	Welch [20]
11	3(generally, 5)(6,8,10)	Boston and McGuire [21]
15	6(6,7,8,9,10,11)	

3) $m \geq 5$: For $m = 5$, there are six m-sequences of length 31, whose Hamming cross-correlation distributions are listed in Table 3.4. For these cases, the peak correlation values are either 10 or 11, which means that there are no doubly counted decodable erasure patterns because there are no Hamming correlation values larger than $k - 2 = 14$. Therefore, any expanded parity check matrix with $b \leq 6$ has erasure decoding performance achieving the upper bound. The maximum value of the Hamming cross-correlation of the m-sequence and its decimated sequences can be derived as $\left\lfloor \frac{2^m+2^{\frac{(m+2)}{2}}+3}{4} \right\rfloor$ [22]. Thus, for $m \geq 5$, I have

$$\left\lfloor \frac{2^m + 2^{\frac{(m+2)}{2}} + 3}{4} \right\rfloor < 2^{m-1} - 2 = k - 2. \tag{3.59}$$

Thus, it is easily checked that there are no doubly counted erasure patterns for construction of the expanded parity check matrices for any combinations of an m-sequence and its decimated sequences for $m \geq 5$. Therefore, the total number of decodable erasure patterns of TS-AGD with expanded parity check matrix constructed by bit-inverted m-sequences can be maximized for the cyclic MDS codes. However, the performance by TS-AGD is worse than that of the perfect decoding for cyclic MDS codes.

3.3.3 Perfect Decoding by TS-AGD with Expanded Parity Check Matrix for Cyclic MDS Codes

In order to achieve the perfect decoding by TS-AGD with the expanded parity check matrix for cyclic MDS codes, the required stopping redundancy $\rho = b(n-k)$ is grown exponentially as n and k increase. It is known to be NP-hard to calculate or approximate the exact value ρ for the perfect decoding [28]. For small values of n and k of the cyclic MDS codes, it will be shown that I can find the optimal ρ which meets the lower bound. In this paper, I only consider the case of $\rho \leq 3(n-k)$ and I propose a construction method of the expanded parity check matrix for the perfect decoding in this subsection. First, three lower bounds on the stopping redundancy are proposed.

a) Lower Bounds on ρ for the Perfect Decoding by TS-AGD

The first lower bound is similar to the Gilbert (sphere packing) bound as in the following theorem.

Theorem 3.3 (Gilbert-like lower bound)**.**

$$\rho \geq \left\lceil \frac{\binom{n}{n-k}}{n((n-k)k+1)} \right\rceil (n-k) \tag{3.60}$$

Proof. Suppose that an expanded parity check matrix has b parity check sequences. If there are no doubly counted decodable erasure patterns, the number of decodable erasure patterns is $bn((n-k)k+1)$ from Proposition 3.3, which is larger than or equal to $\binom{n}{n-k}$. Thus, the theorem is proved. □

This bound can be improved by lotto designs [16] and the Bonferroni inequality [14].

Definition 3.6 (Lotto design [16])**.** *An (n,k,p,t)-lotto design is an n-set V of elements and a set $\mathcal{B}$ of k-element subsets (blocks) of V, such that for any p-subset P of V, there is a block $B \in \mathcal{B}$, for which $|P \cap B| \geq t$. $L(n,k,p,t)$ denotes the smallest number of blocks in any (n,k,p,t)-lotto design.*

By using the above lotto design, I can obtain more improved lower bounds on ρ as follows.

Theorem 3.4 (Lower bound by the lotto design)**.**

$$\rho \geq \left\lceil \frac{L(n, n-k, n-k, n-k-1)}{n} \right\rceil (n-k) \tag{3.61}$$

Proof. In order to decode the cyclic MDS codes, it is necessary for the Hamming correlation values to be less than or equal to 1, i.e., $R_H(\tau) \geq |S_e| - 1$. It also means that the intersection between the standard indices and the support set of erasure sequence is larger than or equal to $n-k-1$. Then, the minimum number of parity check sequences in the expanded parity check matrix is lower bounded by $\frac{L(n,n-k,n-k,n-k-1)}{n}$. □

The lotto design improves the lower bound in Theorem 3.4. Moreover, the lower bound for ρ can also be improved by the Bonferroni inequality as follows.

Theorem 3.5 (Lower bounds by the Bonferroni inequality)**.**

$$\rho \geq \left\lceil \frac{\binom{n}{n-k} - 4A(n, 6, n-k)}{n(k(n-k)-3)} \right\rceil (n-k) \tag{3.62}$$

where $A(n, d, w)$ denotes the maximum number of codewords for the (n, d, w) constant weight codes.

Proof. For an expanded parity check matrix with b parity check sequences, the number of decodable erasure patterns follows (3.10), whose right hand side can be used as an upper bound. In this approach, the second term is calculated as in Theorem 3.2 if the Hamming auto- and cross-correlations of the parity check sequences are known. If cyclically shifted parity check sequences are considered, I have bn distinct parity check sequences, which can be considered as constant weight codewords. Now, I have to count the number of two codewords with Hamming distance less than or equal to 4. By definition, $A(n, 6, k)$ is the maximum number of n-tuple binary codewords which have weight of k and the minimum Hamming distance 6. Then, for each codeword, there exist at least $bn - A(n, 6, n-k)$ codewords which have Hamming distance less

than or equal to 4. Thus, the total number of pair of codewords with Hamming distance less than or equal to 4 is at least $\frac{bn}{2}\left(bn - A(n, 6, n-k)\right)$ because all pairs are counted twice. The minimum value of the second term in the RHS of (3.10) can be computed for $n = k - 2$, that is, Hamming distance 4. Thus, I have

$$\sum_{I \subset V, |I|=2} \left| \bigcap_{i \in I} E_i \right| \geq \sum_{\tau_1, \tau_2 = 0}^{n-1} \sum_{1 \leq i < j \leq b} 4R_{s_{p,i}(t+\tau_1), s_{p,j}(t+\tau_2)}(k-2)$$
$$\geq 4 \times \frac{b}{2}(bn - A(n, 6, n-k)). \quad (3.63)$$

From Proposition 3.3, (3.63), $\left|\bigcup_{i \in V} E_i\right| = \binom{n}{n-k}$, and $|V| = bn$, the right inequality in (3.10) can be modified as (3.62). □

The value of $A(n, d, w)$ is not exactly known in general and its upper bounds are used in this chapter.

b) Examples of the Perfect Decoding for $\rho \leq 3(n-k)$

Table 3.5 lists the required values b for the perfect decoding by TS-AGD with expanded parity check matrix for (n, k) cyclic MDS codes. The underlined values denote the maximum values among the previously derived three lower bounds and the values in parenthesis refer to the lower bounds on b, which are different from the numerically obtained values of b.

Algorithm 3.2 shows one of the simple construction method of the expanded parity check matrix for (n, k) cyclic MDS codes using the set of $(n-k)$-erasure patterns. Using Algorithm 2, the values of b for the perfect decoding are numerically derived for (n, k) cyclic MDS codes in Table 3.5.

To obtain specific values of the lower bounds, the upper bounds of $A(n, d, w)$ in [23] and the lower bounds of $L(n, k, p, t)$ in [24] are used.

Some (n, k) MDS codes in Table 3.5 can be analyzed as follows.

1) $(n, k) = (10, 5)$: The lower bound by Theorem 3.5 shows a stricter bound compared to the other bounds. The values b by Theorems 3.3 and 3.4 are computed

Algorithm 3.2 Greedy algorithm for the construction of the expanded parity check matrix H

Require: $b(n-k) \times n$ expanded parity check matrix H, the set of all $(n-k)$ erasure sequences E, $S = \phi$, $b = 1$, and $\tau = 0$

while $E \setminus S \neq \phi$ **do**

$v \in E \setminus S$

$s_{p,b}(t) \leftarrow \bar{v}$

for $\tau = 0$ to $n-1$ **do**

$C \leftarrow \{s_e(t) | \sum_{t=0}^{n-1} s_e(t) s_{p,b}(t+\tau) \geq |S_e| - 1, s_e(t) \in E\}$

$S \leftarrow S \cup C$

end for

$b \leftarrow b + 1$

end while

Table 3.5: Required $b \leq 3$ for perfect decoding with expanded parity check matrix for (n, k) cyclic MDS codes with $3 \leq k \leq 8$ and $8 \leq n \leq 20$

k/n	8	9	10	11	12	13	14	15	16	17	18	19	20
3	1	1	1	1	1	**2(1)**	1	2	2	2	2	2	**3(2)**
4	1	1	2	**3(2)**	**3(2)**	3	**4(3)**	**4(3)**					
5	1	1	2	2	3	**5(3)**							
6	1	1	2	2	**4(3)**								
7	1	1	1	2	3								
8	1	1	1	1	**3(2)**	**5(3)**							

as

$$b_{Thm.3} = \left\lceil \frac{\binom{10}{5}}{10(5 \times 5 + 1)} \right\rceil = \lceil 0.969 \rceil = 1 \tag{3.64}$$

$$b_{Thm.4} = \left\lceil \frac{L(10, 5, 5, 4)}{10} \right\rceil = \left\lceil \frac{10}{10} \right\rceil = 1 \tag{3.65}$$

whereas Theorem 3.5 gives us a tighter lower bound as

$$b_{Thm.5} = \left\lceil \frac{\binom{10}{5} - 4 \times 7}{10(5 \times 5 - 3)} \right\rceil = \lceil 1.0181 \rceil = 2. \tag{3.66}$$

Using Algorithm 3.2, the expanded parity check matrix can be constructed with two parity check sequences as

$$\begin{aligned} s_{p,1}(t) &= (0101100101) \\ s_{p,2}(t) &= (1000011011). \end{aligned} \tag{3.67}$$

2) $(n, k) = (11, 5)$: The values b of the three lower bounds are equal to 2. Construction of the expanded parity check matrix can be realized by the characteristic sequences of the cyclic difference sets with parameters $(11, 5, 2)$ as

$$\begin{aligned} s_{p,1}(t) &= (10100011101) \\ s_{p,2}(t) &= (11011100010). \end{aligned} \tag{3.68}$$

3) $(n, k) = (13, 4)$: The values of b by Theorems 3.3, 3.4, and 3.5 are given as

$$b_{Thm.3} = \left\lceil \frac{\binom{13}{4}}{13(4 \times 9 + 1)} \right\rceil = \lceil 1.486 \rceil = 2 \tag{3.69}$$

$$b_{Thm.4} = \left\lceil \frac{L(13, 4, 4, 3)}{13} \right\rceil = \lceil 2.153 \rceil = 3 \tag{3.70}$$

$$b_{Thm.5} = \left\lceil \frac{\binom{13}{4} - 4 \times 13}{13(4 \times 9 - 3)} \right\rceil = \lceil 1.545 \rceil = 2. \tag{3.71}$$

Using Algorithm 3.2, the optimal expanded parity check matrix of the $(13, 4)$ cyclic MDS code can be constructed by the following three parity check sequences as

$$\begin{aligned} s_{p,1}(t) &= (1100011111110) \\ s_{p,2}(t) &= (1111011101001) \\ s_{p,3}(t) &= (1010111100111). \end{aligned} \tag{3.72}$$

3.3.4 TS-AGD with Submatrix Inversion for Cyclic MDS Codes

Matrix inversion is not widely used in the erasure decoding but for some codes in the erasure channel, it is permissible for small submatrix inversion. In particular, raptor codes [25] or regenerating codes for distributed storage systems [26] often use an inversion operation of a small submatrix for decoding. The conventional assumption of stopping redundancy for IED is not an inversion-based decoding, but it requires lots of additional check nodes for a large value of n. However, TS-AGD allowing submatrix inversion up to a $u \times u$ matrix dramatically reduces the stopping redundancy for the perfect decoding. The operation of submatrix inversion in the proposed TS-AGD for cyclic MDS codes is always guaranteed by the following proposition.

Proposition 3.4 (The nonsingularity of parity check matrix of cyclic MDS codes)**.** *For any square submatrix of the modified parity check matrix for MDS codes is nonsingular.*

Proof. It can be proved by Theorem 8 in Chapter 11.4 in [27]. □

Thus, Algorithm 3.1 becomes Algorithm 3.3 for the perfect decoding by TS-AGD with expanded parity check matrix and submatrix inversion for cyclic MDS codes. In Algorithm 3.3, the u elements of the syndrome vector with indices $i_{j_1}, i_{j_2}, ..., i_{j_u}$, where for $k \in [1, u]$, j_k is in the $S_e \cap \bar{S}_p$ can be computed as

$$s_{i_{j_k}} = e_{j_1} h_{i_{j_k}, j_1} + e_{j_2} h_{i_{j_k}, j_2} + ... + e_{j_u} h_{i_{j_k}, j_u} + a_{i_{j_k}} = 0, \text{for } k \in [1, u] \quad (3.73)$$

where $a_{i_{j_k}}$ denotes the symbols recovered by the received vector and the parity check matrix in columns whose indices are not in $S_e \cap \bar{S}_p$. By solving the above system of linear equations by submatrix inversion, the erasure symbols $e_{j_1}, e_{j_2}, ..., e_{j_u}$ can be recovered. Then, the remaining erasure symbols are decoded by the inversionless VNU.

Three lower bounds on b for the perfect decoding by TS-AGD with expanded parity check matrix and submatrix inversion for the cyclic MDS codes are derived.

Algorithm 3.3 TS-AGD for the expanded parity check matrix with submatrix inversion

Require: $b(n-k) \times n$ parity check matrix H, parity check sequences $s_{p,i}(t)$, erasure sequence $s_e(t)$

for $i = 0$ to u **do**
  for $j = 0$ to b **do**
    for $\tau = 0$ to $n-1$ **do**
      if $\sum_{t=0}^{n-1} s_e(t+\tau)s_{p,j}(t) = |S_e| - i$ **then**
        if $i \leq 1$ **then**
          Follow Algorithm 3.1 for cyclic MDS codes
          STOP
        else
          Select columns of H with indices in $S_e \cap \bar{S}_p$
          Select $|S_e \cap \bar{S}_p|$ rows whose indices are indices of "1" in the j-th column of the standard vector, $j \in \bar{S}_e \cap S_p$
          Invert $|S_e \cap \bar{S}_p| \times |S_e \cap \bar{S}_p|$ submatrix
          Find erasure symbols with indices in $S_e \cap \bar{S}_p$
          Decode the other $|S_e \cap S_p|$ erasure symbols by additional iterations without inversion
          STOP
        end if
      end if
    end for
  end for
end for

a) Bonferroni Inequality for TS-AGD with Expanded Parity Check Matrix and Submatrix Inversion for Cyclic MDS Codes

The Bonferroni inequality in (3.10) can be modified as in the following theorems.

Theorem 3.6 (The first term of the Bonferroni inequality with submatrix inversion)**.** *The first term in (3.10) is modified in the expanded parity check matrix with submatrix inversion as*

$$\sum_{I\subset V,|I|=1}|E_i| = bn\sum_{i=0}^{u}\binom{n-k}{i}\binom{k}{i}. \tag{3.74}$$

Proof. Suppose that the expanded parity check matrix has b parity check sequences. The τ-shifted parity check sequence $s_p(t+\tau)$ can correct $n-k$ erasure symbols if $R_H(\tau) \geq n-k-u$. If $R_H(\tau) = n-k-i$, $n-k$ erasure symbols are in the $n-k-i$ standard indices and the decoder can correct $\binom{n-k}{n-k-i}\binom{k}{i}$ erasure patterns. The number of decodable erasure patterns is the sum of all $i \in [0,u]$, which proves the theorem. □

Theorem 3.7 (The second term in the Bonferroni inequality with submatrix inversion)**.** *The second term can also be modified in (3.10) in the expanded parity check matrix with submatrix inversion as*

$$\begin{aligned}\sum_{I\subset V,|I|=2}\left|\bigcap_{i\in I}E_i\right| = \sum_{1\leq i<j\leq b}\sum_{\mu=0}^{u-1}\sum_{\tau_1,\tau_2=0}^{n-1}\sum_{0\leq\zeta+\eta_1\leq u,0\leq\zeta+\eta_2\leq u}\\ \binom{k-2u+\mu}{\zeta}\binom{2u-\mu}{\eta_1}\binom{2u-\mu}{\eta_2}\\ \binom{n-k-2u+\mu}{n-k-\eta_1-\eta_2-\zeta}F_{s_{p,i}(t+\tau_1),s_{p,j}(t+\tau_2)}(k-2u+\mu)\end{aligned} \tag{3.75}$$

where $F_{s_{p,i}(t+\tau_1),s_{p,j}(t+\tau_2)}(\gamma)$ *returns 1 if* $\sum_{t=0}^{n-1}(s_{p,i}(t+\tau_1)s_{p,j}(t+\tau_2)) = \gamma$ *and otherwise, 0.*

Proof. The proof is the generalization of that of Theorem 3.2. For the i-th and the j-th parity check sequences cyclically shifted by τ_1 and τ_2, the function $F_{s_{p,i}(t),s_{p,j}(t+\tau_2)}(\gamma)$ is computed as follows. If $\gamma = k-2u+\mu$ for $\mu \in [0,u]$, I have $|A_{11}| = k-2u+\mu$,

$|A_{10}| = |A_{01}| = 2u - \mu$, and $|A_{00}| = n - k - 2u + \mu$. Let ζ, η_1, and η_2 be the numbers of erasure symbols in A_{00}, A_{10}, and A_{01}, respectively. To decode the received vector in two parity check sequences, the Hamming correlation of each parity check sequences is less than or equal to u, where $\zeta + \eta_1 \leq u$ and $\zeta + \eta_2 \leq u$. This provides the proof. □

b) Lower Bounds of the Stopping Redundancy for TS-AGD in an Expanded Parity Check Matrix with Submatrix Inversion

The three lower bounds on b for TS-AGD with expanded parity check matrix and $u \times u$ submatrix inversion for the cyclic MDS codes can be modified as in the following theorems.

Theorem 3.8 (Gilbert-like lower bound of TS-AGD with expanded parity check matrix and submatrix inversion)**.**

$$\rho \geq \left\lceil \frac{\binom{n}{n-k}}{n \sum_{i=0}^{u} \binom{n-k}{i} \binom{k}{i}} \right\rceil (n-k) \tag{3.76}$$

Proof. It manifests from Theorem 3.3. □

Theorem 3.9 (Lower bound by the lotto design for the TS-AGD with expanded parity check matrix and submatrix inversion)**.**

$$\rho \geq \left\lceil \frac{L(n, n-k, n-k, n-k-u)}{n} \right\rceil (n-k). \tag{3.77}$$

Proof. It manifests from Theorem 3.4. □

Theorem 3.10 (Lower bound by the Bonferroni inequality for the TS-AGD with expanded parity check matrix and submatrix inversion)**.**

$$\rho \geq \left\lceil \frac{\binom{n}{k} - \binom{2u}{u}^2 A(n, 4u+2, n-k)}{n(\sum_{i=0}^{u} \binom{n-k}{i} \binom{k}{i} - \binom{2u}{u}^2)} \right\rceil (n-k) \tag{3.78}$$

where $A(n, d, w)$ is the maximum number of codewords for (n, d, w) constant weight codes.

Proof. The proof is the generalization of that of Theorem 3.5. Two parity check sequences that have Hamming correlation less than $k - 2u$ have no doubly counted decodable erasure patterns, because two parity check sequences cannot be simultaneously decoded regardless of their locations of erasure symbols for $|A_{00}| \leq n - k - 2u - 1$. For $|A_{00}| = n - k - 2u$, the doubly counted decodable erasure patterns exist only when u erasure symbols are located in A_{10} and A_{01}, respectively, where $|A_{10}| = |A_{01}| = 2u$. Then, the number of cases is $\binom{2n}{n}^2$. The remaining part is similar to the proof of Theorem 3.5. □

Chapter 4

New Constructions of Binary and Ternary LRCs Using Cyclic Codes and Existing LRCs

In this chapter, new constructions of binary and ternary LRCs are proposed using cyclic codes and existing LRCs. First, new constructions of binary LRCs using cyclic codes are introduced.

4.1 Constructions of Binary LRCs Using Cyclic Codes

In this section, new binary LRCs are proposed by using cyclic codes, where some of them are optimal in terms of bounds in (2.2), (2.3), and (2.4).

Construction 4.1 (Cyclic binary LRCs with $d = 4$)**.** *For $(r+1)|n$, let $v = \frac{n}{r+1}$ and $u = r + 1$, where gcd$(u, v) = 1$ and $u, v \geq 2$. Let $g(x)$ be a generator polynomial of the cyclic binary LRC and β' be an u-th root of unity. Then, $(uv, uv-$deg$(g(x)), 4, u-1)$ binary LRCs can be constructed by the following generator polynomials:*

1) *For $2|r$, $g(x) = (x^v + 1)g_1(x)$, where $g_1(x)$ is the minimal polynomial of β' over F_2.*

2) *For $r = 2^m - 1$, $g(x) = (x^v + 1)(x + 1)^{2^{m-1}}$, where m is a positive integer.*

Proof. First, I have to prove that they are LRCs, that is, there are at least one check with Hamming weight $r+1$. Since $(x^v+1)|g(x)$, $1+x^v+...+x^{(u-1)v}$ can be a check of H. Thus, the proposed codes are LRC with $r=u-1$.

Next, it is necessary to prove that the minimum distance of the proposed LRCs is 4. It is easy to check that there is no codeword with odd Hamming weight for both cases of $g(x)$. Subsequently, I have to prove the nonexistence of codewords with Hamming weight 2 and the existence of codewords with Hamming weight 4.

For $2|r$, suppose that I have a codeword $c(x)$ with Hamming weight 2. Since $(x^v+1)|c(x)$, $c(x)=1+x^{vl}, l\in[u]$. Further, $c(\beta')=0$, that is, $(\beta')^{vl}=1$ and $u|(vl)$ and thus, $l=0$ from $\gcd(u,v)=1$. Then, $c(x)=1+1=0$, which contradicts the assumption. Thus, there is no codeword with Hamming weight 2. It is easy to check that $g(x)$ divides $(1+x^u)(1+x^v)$, which is a codeword with Hamming weight 4.

For $r=2^m-1$, suppose that there exists a codeword $c(x)$ with Hamming weight 2. Since $(x^{2^{m-1}}+1)|c(x)$, $c(x)=x^{l2^{m-1}}+1, l\in[0,2v-1]$. Since $c(x)=(x^{2^{m-1}}+1)(x^{(l-1)2^{m-1}}+x^{(l-2)2^{m-1}}+...+x^{2^{m-1}}+1)$, $(x+1)$ should divide $(x^{(l-1)2^{m-1}}+x^{(l-2)2^{m-1}}+...+x^{2^{m-1}}+1)$ and thus l should be even. Let $l=2l'$. Then, $c(x)=x^{l'2^m}+1$, which satisfies $c(\beta'')=0$, where β'' is a v-th root of unity. Then, $(\beta'')^{l'2^m}=1$. Given that $u=2^m$, $\gcd(u,v)=1$, and $l'\in[0,v-1]$, I have $l=0$ and $c(x)=1+1=0$, which contradicts the assumption. Thus, there is no codeword with Hamming weight 2. Since $g(x)=x^{2^{m-1}+v}+x^{2^{m-1}}+x^v+1$, there exists a codeword with Hamming weight 4. □

Some classes in Construction 4.1 are optimal or r-optimal as in the following proposition.

Proposition 4.1 (Optimality of Construction 4.1)**.** *For LRCs in Construction 4.1, two classes of* $(2l, l-1, 4, 1)$ *and* $(4l, 3l-2, 4, 3)$ *cyclic binary LRCs are optimal for* $l\geq 3$, $gcd(2,l)=1$ *and one class of* $(3l, 2l-2, 4, 2)$ *proposed cyclic binary LRC is* r*-optimal by (2.2) for* $l\geq 4$, $gcd(3,l)=1$.

Proof. Two classes of the proposed $(2l, l-1, 4, 1)$ and $(4l, 3l-2, 4, 3)$ LRCs have the same parameters as the optimal binary LRCs in Construction 4.1 in [38]. If the class of proposed $(3l, 2l-2, 4, 2)$ LRC is not r-optimal, the inequality $\frac{k}{n} \leq \frac{n-2}{2n}$ derived from (2.2) for LRC with $r = 1$ should be satisfied but it does not hold for $l \geq 4$, which tells that the class of proposed $(3l, 2l-2, 4, 2)$ LRC is r-optimal. □

Optimal or r-optimal LRCs with the same parameters were also introduced in [38], [39], but they were not cyclic.

Noncyclic binary LRCs with larger minimum distance are also proposed as in the following construction.

Construction 4.2 (Linear binary LRC with $d \geq 6$ and $r = 2$)**.** *Let β be a primitive element of the finite field F_{2^m} and n a positive integer larger than or equal to 9 and divisible by 3 such that $\frac{2n}{3} \leq 2^m - 1$. Let $\mathcal{C}_E$ be a $(2^m - 1, 2^m - m - 2, 4)$ expurgated Hamming code with generator polynomial $g(x) = (x+1)g_1(x)$, where $g_1(x)$ is the minimal polynomial of β over $\mathbb{F}_2$. A $(\frac{2n}{3}, \frac{2n}{3} - m - 1, \geq 4)$ shortened expurgated Hamming code $\mathcal{C}_S$ can be generated by shortening the first $(2^m - \frac{2n}{3} - 1)$ information bits of $\mathcal{C}_E$. Then, concatenation of $\mathcal{C}_S$ and an $(n, \frac{2n}{3})$ cyclic code with parity check polynomial $x^{\frac{2n}{3}} + x^{\frac{n}{3}} + 1$ as an inner code makes an $(n, \frac{2n}{3} - \lceil \log_2 (\frac{2n}{3} + 1) \rceil - 1, d \geq 6, 2)$ LRC $\mathcal{C}_C$.*

Proof. Let H_E and H_S be the parity check matrices of $\mathcal{C}_E$ and $\mathcal{C}_S$, respectively. Let $H_E = [H_1' \; H_2' \; H_3']$ and $H_S = [H_2' \; H_3']$, where H_1' is the $(m+1) \times (2^m - 1 - \frac{2n}{3})$ matrix and H_2' and H_3' are the $(m+1) \times \frac{n}{3}$ matrices. The parity check matrix of the cyclic inner code can be given as $H_I = [I_{\frac{n}{3}} \; I_{\frac{n}{3}} \; I_{\frac{n}{3}}]$. The parity check matrix of the proposed LRC is then given as

$$H = \begin{bmatrix} H_O \\ H_I \end{bmatrix} = \begin{bmatrix} H_2' & H_3' & O \\ I_{\frac{n}{3}} & I_{\frac{n}{3}} & I_{\frac{n}{3}} \end{bmatrix} \tag{4.1}$$

where O denotes the $(m+1) \times \frac{n}{3}$ zero matrix. It is easily verified that the locality of LRC is 2 from the lower part of H, H_I. Adding all of the rows of H_I makes an all-one

vector and thus, there is no codeword with odd Hamming weight. At this stage, I have to prove that there is no codeword with Hamming weight 4. Suppose that there is a codeword with Hamming weight 4. Since the minimum distance of $\mathcal{C}_S$ is larger than or equal to 4, the nonzero elements of the codeword with Hamming weight 4 should be located in the first $\frac{2n}{3}$ elements of the codeword. In order to satisfy H_I, the codeword polynomial should be a form of $c(x) = x^i + x^j + x^{i+\frac{n}{3}} + x^{j+\frac{n}{3}} = (x^i + x^j)(1 + x^{\frac{n}{3}})$, $0 \leq i < j < \frac{n}{3}$. Clearly, $g_1(x)|c(x)$ and thus $c(\beta) = 0$ but $\beta^{\frac{n}{3}} \neq 1$ and $\beta^i + \beta^j \neq 0$. Thus, there is no codeword with Hamming weight 4. □

Using (2.4), the optimality of the LRCs in Construction 4.2 can be stated as follows:

Proposition 4.2 (Optimality of Construction 4.2)**.** *Let k_{opt} and k be the dimensions of the LRCs satisfying the equality in (2.4) and the proposed LRCs in Construction 4.2, respectively. If $n \geq 33$, the proposed LRCs are r-optimal. Further, if $n \geq 33$ and $\lceil \log_2(\frac{2n}{3} + 1) \rceil + 1 = \lceil \log_2(1 + n) \rceil$, the proposed LRCs are r-optimal and k-optimal.*

Proof. If $n = 33$ or 36, $k = k_{opt}$ by (2.4) and thus LRC is r- and k-optimal. For $n \geq 39$, the proposed LRC is also r-optimal by (2.2), because (2.2) is rewritten as $\frac{k}{n} \leq \frac{k_{opt}}{n} = \frac{2}{3} - \frac{\lceil \log_2(n+1) \rceil}{n} \leq \frac{n-2}{2n} \leq \frac{1}{2}$ for $r = 1$, that is, $\frac{n}{6} \leq \lceil \log_2(n + 1) \rceil$. Thus it does not hold for $n \geq 39$ and $r = 1$, which tells that the proposed LRC with $n \geq 33$ is r-optimal. Also, $k_{opt} = \frac{2n}{3} - \lceil \log_2(n + 1) \rceil$ and $k = \frac{2n}{3} - \lceil \log_2(\frac{2n}{3} + 1) \rceil - 1$ and thus if $\lceil \log_2(\frac{2n}{3} + 1) \rceil + 1 = \lceil \log_2(1 + n) \rceil$, the proposed LRCs are r-optimal and k-optimal, i.e., $k = k_{opt}$. □

Note that (2.4) does not gaurantee d-optimal because (2.4) is valid for $d \geq 5$. Table 4.1 shows the parameters of optimal LRCs with $d = 6$ and $r = 2$ from the existing works [29] [40]. Thus, Construction 4.2 gives us new binary r- and k-optimal LRCs.

For $r \geq 3$, there is a construction of LRC based on nonlinear codes as in the following construction.

Table 4.1: Optimality of the existing binary LRCs with $d = 6$ and $r = 2$

	(n, k), conditions	r-opt	k-opt	d-opt
Thm. 1 in [29]	$(2^m - 1, \frac{2n}{3} - m), 2\|m$	O	O	O
Cor. 1 in [40]	$(3s, 2s - 4), 4 \leq s \leq 5$	O	O	O

Construction 4.3 (Nonlinear binary LRC with $d \geq 5$ and $r \geq 3$)**.** *Let β be a primitive element of the finite field F_{2^m} and n a positive integer such that $n + 1$ is divisible by $r + 1$. Let v be $\frac{n+1}{r+1}$ and m should satisfy $rv \leq 2^m - 1$. Let $\mathcal{C}_E$ be a $(2^m - 1, 2^m - m - 2, 4)$ expurgated Hamming code as defined in the previous construction. An $(rv, rv-m-1, \geq 4)$ shortened expurgated Hamming code $\mathcal{C}_S$ can be generated by shortening the first (2^m-rv-1) information bits of $\mathcal{C}_E$. Then, the $(n+1, rv-m-1, \geq 4, r)$ linear LRC $\mathcal{C}_C$ can be constructed by concatenating $\mathcal{C}_S$ and an $(n + 1, rv)$ cyclic code with parity check polynomial $x^{rv} + x^{(r-1)v} + ... + x^v + 1$. By selecting all codewords with the i-th element 1 for a fixed $i \in [rv, n]$ and deleting the i-th elements from the selected codewords, an $(n, 2^{rv-m-2}, \geq 5, r)$ nonlinear binary LRC can be constructed.*

Proof. Let H_S be a parity check matrix of $\mathcal{C}_S$. A parity check matrix of the (n, rv) cyclic code can be given as $H_I = [I_v \; ... \; I_v] = [H'_L \; I_v]$, where H'_L is a $v \times rv$ matrix consisting of r I_v's. Then, the parity check matrix of the proposed LRC is given as

$$H = \begin{bmatrix} H_O \\ H_I \end{bmatrix} = \begin{bmatrix} H_S & O \\ H'_L & I_v \end{bmatrix}$$

where O denotes the $(m + 1) \times v$ zero matrix. It is easily checked that the locality of LRC is r from the lower part of H, H_I. Adding all of the rows of H_I makes an all-one vector and thus, there is no codeword with odd Hamming weight. Now, I have to prove that there is no codeword with Hamming weight 4. Suppose that there is a codeword with Hamming weight 4. Since the minimum distance of $\mathcal{C}_S$ is larger than or equal to four, the nonzero elements of the codeword with Hamming weight 4 should be located in the first rv elements of the codeword. If the codewords with nonzero elements in

the index $[rv, n]$ exist, their Hamming weights are larger than or equal to six. Since I select the codewords with the i-th element 1 for a fixed $i \in [rv, n]$, the selected code has the minimum Hamming weight larger than or equal to six. By deleting the i-th elements from all selected codewords, their minimum Hamming weight is larger than or equal to five. At this point, I have to prove the number of codewords of the proposed LRCs. First, I can decompose $\mathcal{C}_C$ into two classes, that is, a set of codewords with the i-th element 1, $\mathcal{C}_1^{(i)}$ and a set of the remaining codewords, $\mathcal{C}_0^{(i)}$. Let $\mathbf{c}_i$ be a codeword of $\mathcal{C}_1^{(i)}$. Then for any $\mathbf{c}_j$ in $\mathcal{C}_1^{(i)}$, $\mathbf{c}_i \bigoplus \mathbf{c}_j$ belongs to $\mathcal{C}_0^{(i)}$ and thus $|\mathcal{C}_1^{(i)}| \leq |\mathcal{C}_0^{(i)}|$. Further, for any $\mathbf{c}_k$ in $\mathcal{C}_0^{(i)}$, $\mathbf{c}_i \bigoplus \mathbf{c}_k$ belongs to $\mathcal{C}_1^{(i)}$ and thus $|\mathcal{C}_0^{(i)}| \leq |\mathcal{C}_1^{(i)}|$. Accordingly, $|\mathcal{C}_0^{(i)}| = |\mathcal{C}_1^{(i)}| = 2^{rv-m-2}$, which is the number of codewords of the proposed LRCs. □

In order to encode the proposed LRCs in Construction 4.3, the parity check matrix of $\mathcal{C}_C$ in Fig. 4.1 is used, where $m + 1 < v$. Assume that index i is set to rv, whose element will be 'deleted' from all codewords of $\mathcal{C}_1^{(i)}$ later. For the message vector $\mathbf{m} = (m_1, m_2, ..., m_{rv-m-2})$, the codeword can be given as $\mathbf{c} = (p_0, m_1, m_2, ..., m_{rv-m-2}, p_1, p_2, ..., p_{(r+1)v-1})$, where $p_{m+2} = 1$ will be deleted for the proposed LRCs. First, the value of p_0 is computed by the $(m+1)$-th row of H_C, $\mathbf{m}$, and $p_{m+2} = 1$. Then, the values of $p_1, p_2, ..., p_{m+1}$ are computed using H_S and the values of $p_{m+3}, p_{m+4}, ..., p_{m+v+1}$ can also be computed by H_I. The codeword of the proposed LRC is then given as $(p_0, m_1, m_2, ..., m_{rv-m-2}, p_1, p_2, ..., p_{m+1}, p_{m+3}, p_{m+4}, ..., p_{m+v+1})$. Thus, the encoding procedure of the proposed LRCs is identical to that of the linear code C_C.

Using (2.2), the optimality of the LRCs in Construction 4.3 can be stated as follows:

Proposition 4.3 (Optimality of Construction 4.3)**.** *For some n, the proposed binary LRCs in Construction 4.3 is r-optimal.*

Similar to the proof of the previous cases, it can be easily checked that (2.2) does not hold for $(n, k, d, r-1)$ LRC in Construction 4.3. Table 4.2 lists the codelength of

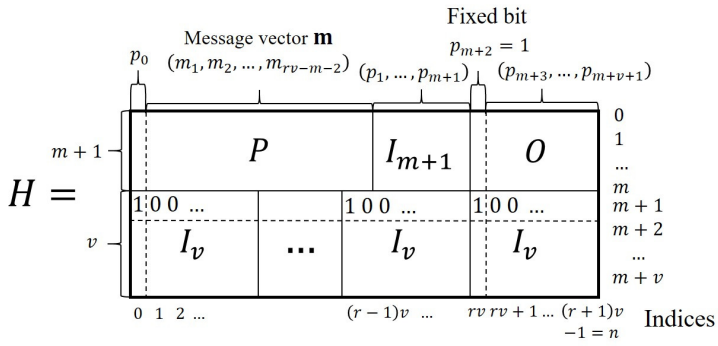

Figure 4.1: Parity check matrix of the LRC in Construction 3 with $i = rv$.

Table 4.2: Codelength of r-optimal LRCs in Construction 3 with $r \in [3, 8]$

r	3	4	5	6	7	8
n	≥ 67	≥ 124	≥ 209	≥ 286	≥ 431	≥ 620

r-optimal LRCs for $r \in [3, 8]$ in Construction 4.3.

As a special case for the short codelength, I can modify Constructions 4.3 as in the following example to construct binary LRC.

Example 4.1 (Construction of linear binary LRC with $r = 3$)**.** *Linear binary LRC with $d = 6$ and $r = 3$ can be constructed for short codelength without deleting the i-th bit in Construction 4.3. A $(20, 10, 6, 3)$ LRC is constructed by using a $(15, 10, 3)$ binary cyclic code with $g(x) = (x^4 + x + 1)(x + 1)$, which is optimal by (2.3) and better than dimension 9 from [40].*

Table 4.3 summarizes the optimality of the proposed binary LRCs.

4.2 Constructions of Linear Ternary LRCs Using Cyclic Codes

Using the construction method of the binary LRCs, two linear ternary LRCs are proposed in the following constructions.

Table 4.3: Optimality of the proposed binary LRCs using cyclic codes

	Conditions	r-opt	k-opt	d-opt
Const.4.1	Class with $d=4$, $r=2$ of Prop. 4.1	O	?	?
Const.4.1	Classes with $d=4$, $r=1,3$ of Prop. 4.1	O	O	O
Const.4.2	$d\geq 6$, $r=2$, $n\geq 33$, $3\|n$	O	?	?
Const.4.2	$d\geq 6$, $r=2$, $n\geq 33$, $3\|n$ $\lceil\log_2(\frac{2n}{3}+1)\rceil+1=\lceil\log_2(1+n)\rceil$	O	O	?
Const.4.3	$d\geq 5$, $(r+1)\|(n+1)$, Prop. 4.3	O	?	?
Exam.4.1	$(n,k,d,r)=(20,10,6,3)$	O	O	O

Theorem 4.1 (Linear ternary LRCs of $d\geq 5$ and $r=2$)**.** *Let β be a primitive element of the finite field F_{3^m} and n a positive integer divisible by 3 such that $\frac{2n}{3}\leq 3^m-1$. Let $\mathcal{C}_E$ be a $(3^m-1, 3^m-m-2, 3)$ cyclic code with generator polynomial $g(x)=(x-1)g_1(x)$, where $g_1(x)$ is the minimal polynomial of β over F_3. A $(\frac{2n}{3}, \frac{2n}{3}-m-1, \geq 3)$ shortened code $\mathcal{C}_S$ can be generated by shortening the first $3^m-1-\frac{2n}{3}$ information bits of $\mathcal{C}_E$. Then, concatenation of $\mathcal{C}_S$ and an $(n, \frac{2n}{3})$ cyclic code $\mathcal{C}_C$ with parity check polynomial $x^{\frac{2n}{3}}+x^{\frac{n}{3}}+1$ as an inner code makes an $(n, \frac{2n}{3}-\lceil\log_3(\frac{2n}{3}+1)\rceil-1, d\geq 5, 2)$ linear ternary LRC.*

Proof. It is easy to check that $\mathcal{C}_E$ has $d\geq 3$ by BCH bound and two consecutive zeros $\{1,\beta\}$. Let H_E and H_S be parity check matrices of $\mathcal{C}_E$ and $\mathcal{C}_S$, respectively. Similarly to Construciton 4.2, the parity check matrix of the proposed LRCs is given as (4.1).

Then, the locality of the proposed LRC is 2 and I have to prove that there is no codeword with Hamming weights of 3 and 4. Suppose that there is a codeword with Hamming weight 3. Since the minimum Hamming weight of $\mathcal{C}_S$ is larger than or equal to 3, the nonzero elements of the codeword with Hamming weight 3 should be located in the first $\frac{2n}{3}$ elements of the codeword. Then, the codeword polynomial should be $c_1(x)=a_1x^i+a_2x^j+a_3x^{i+\frac{n}{3}}$ or $c_2(x)=a_1x^i+a_2x^j+a_3x^k, 0\leq i<j<k<\frac{n}{3}$ and $a_1,a_2,a_3\in\{-1,1\}$. It is easy to check that the checksum of H_I cannot be satisfied for the both cases. If the codeword with Hamming weight 4 has three nonzero elements

located in $[\frac{2n}{3}-1]$ and the other in $[\frac{2n}{3}, n-1]$, it cannot be a codeword because the check $[\mathbf{0}_{\frac{2n}{3}}, \mathbf{1}_{\frac{n}{3}}]$ can be obtained by subtracting $[\mathbf{1}_{\frac{2n}{3}}, \mathbf{0}_{\frac{n}{3}}]$ by $(x-1)$ in $g(x)$ from $[\mathbf{1}_n]$ by H_I. Thus, the codeword with Hamming weight 4 has four nonzero elements in $[\frac{2n}{3}-1]$, which can be represented as $c(x) = (a_1x^i + a_2x^j)(1-x^{\frac{n}{3}})$, $0 \le i < j < \frac{n}{3}$, $a_1, a_2 \in \{-1, 1\}$ because it should satisfy H_I. Then, $c(\beta) \neq 0$ because $\beta^{\frac{n}{3}} \neq 1$ and $\beta^{(j-i)} \neq \pm 1$. Thus, there is no codeword with Hamming weight 4. □

Note that the $(12, 5, 6, 2)$ ternary LRC constructed in Construction 4.1 has the same parameters as those of the eight classes of the optimal ternary LRCs in [33] by (2.2).

In addition, the linear ternary LRC of $r = 3$ can also be constructed as follows.

Theorem 4.2 (Linear ternary LRC of $d \ge 5$ and $r = 3$)**.** *Let β be a primitive element of the finite field F_{3^m} and n a positive integer divisible by 4 such that $\frac{3n}{4} \le 3^m - 1$. Let $\mathcal{C}_E$ be a $(3^m - 1, 3^m - 1 - 2m, 4)$ ternary BCH code with $g(x) = g_1(x)g_2(x)$, where $g_1(x)$ and $g_2(x)$ are the minimal polynomials of β and β^2 over F_3, respectively. A $(\frac{3n}{4}, \frac{3n}{4} - 2m, \ge 5)$ shortened code $\mathcal{C}_S$ can be generated by shortening the first $3^m - 1 - \frac{3n}{4}$ information bits of $\mathcal{C}_E$. Then, concatenation of $\mathcal{C}_S$ and an $(n, \frac{3n}{4})$ cyclic code $\mathcal{C}_C$ with parity check polynomial $x^{\frac{3n}{4}} + x^{\frac{2n}{4}} + x^{\frac{n}{4}} + 1$ as an inner code makes an $(n, \frac{3n}{4} - 2\lceil \log_3(\frac{3n}{4}+1) \rceil, d \ge 5, 3)$ linear ternary LRC.*

Proof. It is easily checked that the locality of the proposed LRC is 3 by the parity check polynomial of $\mathcal{C}_S$. Then, I have to prove that there is no codeword with Hamming weight 4. If there is a codeword with Hamming weight 4, its nonzero element should be located in $[\frac{3n}{4} - 1]$ because C_S has the minimum Hamming weight 4 by BCH bound and the three consecutive zeros. For $a_1, a_2 \in \{-1, 1\}$ and $i \neq j \in [\frac{n}{4} - 1]$, the codewords with Hamming weight 4 can be expressed as in the following five cases;

1) $c_1(x) = x^l(a_1x^i + a_2x^j - a_1x^{i+\frac{n}{4}} - a_2x^{j+\frac{n}{4}}) = x^l(1-x^{\frac{n}{4}})(a_1x^i + a_2x^j)$ for $l \in \{0, \frac{n}{4}\}$.

2) $c_2(x) = a_1x^i + a_2x^j - a_1x^{i+\frac{2n}{4}} - a_2x^{j+\frac{2n}{4}} = (1-x^{\frac{2n}{4}})(a_1x^i + a_2x^j)$.

3) $c_3(x) = a_1x^i + a_2x^j - a_1x^{i+\frac{n}{4}} - a_2x^{j+\frac{2n}{4}} = (1-x^{\frac{n}{4}})(a_1x^i + a_2x^j(1+x^{\frac{n}{4}}))$.

4) $c_4(x) = a_1x^i + a_2x^{j+\frac{n}{4}} - a_1x^{i+\frac{2n}{4}} - a_2x^{j+\frac{2n}{4}} = -x^{\frac{2n}{4}}(1-x^{-\frac{n}{4}})(a_1x^i + a_2x^j(1+x^{-\frac{n}{4}}))$.

5) $c_5(x) = a_1x^i + a_2x^{j+\frac{n}{4}} - a_1x^{i+\frac{n}{4}} - a_2x^{j+\frac{2n}{4}} = (1-x^{\frac{n}{4}})(a_1x^i - a_2x^{j+\frac{n}{4}})$.

For the proofs of 1) and 2), it is easy to check that $c_1(\beta) \neq 0$ because $\beta^{\frac{n}{4}} \neq 1$ and $\beta^{j-i} \neq \pm 1$ and $c_2(\beta) \neq 0$ because $\beta^{\frac{2n}{4}} \neq 1$. For 3), I have to prove $c_3(\beta^k) \neq 0$ for at least one $k \in [1,3]$. Suppose $\beta^{\frac{n}{4}}, \beta^{\frac{2n}{4}}, \beta^{\frac{3n}{4}} \neq -1$. Clearly, $\beta^{\frac{kn}{4}} \neq 1$ for $k \in [1,3]$ and I have $a_1\beta^i + a_2\beta^j(\beta^{\frac{n}{4}}+1) = a_1\beta^{2i} + a_2\beta^{2j}(\beta^{\frac{2n}{4}}+1) = a_1\beta^{3i} + a_2\beta^{3j}(\beta^{\frac{3n}{4}}+1) = 0$ by $c_3(\beta) = c_3(\beta^2) = c_3(\beta^3) = 0$. Then, $\beta^i = \frac{a_1\beta^{2i}}{a_1\beta^i} = \frac{-a_2\beta^{2j}(\beta^{\frac{2n}{4}}+1)}{-a_2\beta^j(\beta^{\frac{n}{4}}+1)}$ and also $\beta^i = \frac{a_1\beta^{3i}}{a_1\beta^{2i}} = \frac{-a_2\beta^{3j}(\beta^{\frac{3n}{4}}+1)}{-a_2\beta^{2j}(\beta^{\frac{2n}{4}}+1)}$. Thus, I have $(\beta^{\frac{2n}{4}}+1)^2 = (\beta^{\frac{n}{4}}+1)(\beta^{\frac{3n}{4}}+1)$, which can be rewritten as $\beta^{\frac{n}{4}}(\beta^{\frac{n}{4}}-1)^2 = 0$ and it contradicts. If $\beta^{\frac{kn}{4}} = -1$ for some $k \in [1,3]$, $c_3(\beta^k) = (1+1)(a_1\beta^{ki}) \neq 0$. Similarly, 4) can be proved. For 5), $\beta^{j-i+\frac{n}{4}} = 1$ by $\beta^i = \frac{a_1\beta^{2i}}{a_1\beta^i} = \beta^{j+\frac{n}{4}}$ and $c_5(\beta) = c_5(\beta^2) = 0$, but there is a contradiction. Thus, there is no codeword with Hamming weight 4. □

4.3 Constructions of Binary LRCs with Disjoint Repair Groups Using Existing LRCs

In this section, a new construction method of binary linear LRCs is proposed using existing LRCs as in the following theorem.

Theorem 4.3 (Construction of new binary LRCs using existing LRCs)**.** *Suppose that for $(r+1)|n$, an $(\frac{nr}{r+1}, k_1, d_1)$ and $(\frac{n}{r+1}, k_2, d_2)$ linear binary codes $\mathcal{C}_1$ and $\mathcal{C}_2$ exist, respectively. Then, $(\frac{nr}{r+1}, k_1 - \frac{n}{r+1}, d_1', r-1)$ binary LRC $\mathcal{C}_1'$ is constructed by adding $\frac{n}{r+1}$ disjoint repair groups in parity check matrix of $\mathcal{C}_1$, which should satisfy $\mathcal{C}_1' \subset \mathcal{C}_1$. By combining $\mathcal{C}_1'$ and $\mathcal{C}_2$, $(n, k_1+k_2-\frac{n}{r+1}, \geq \min(d_1', d_1+d_2), r)$ binary LRC $\mathcal{C}$ with $\frac{n}{r+1}$ disjoint repair groups is constructed.*

Proof. Let $v = \frac{n}{r+1}$. Suppose that $\mathcal{C}'_1$ has a parity check matrix as

$$H_{\mathcal{C}'_1} = \begin{bmatrix} H_{\mathcal{C}_1} \\ H_L \end{bmatrix} \tag{4.2}$$

where $H_L = [I_v, I_v, ..., I_v]$ for identity matrix I_v with size v. Assume that the parity check matrix of $\mathcal{C}$ is given as

$$H_{\mathcal{C}} = \begin{bmatrix} H_{\mathcal{C}_1} & O \\ O & H_{\mathcal{C}_2} \\ H_L & I_v \end{bmatrix}. \tag{4.3}$$

First, it is easy to check that the locality of $\mathcal{C}$ is r from submatrix $[H_L\ I_v]$, the lower part of $H_{\mathcal{C}}$. Similarly, it is also easy to derive n and k from $H_{\mathcal{C}}$ for $\mathcal{C}$. In order to prove Hamming distance, it can be partitioned to three cases by the location of nonzero elements for the each codeword of $\mathcal{C}$ as;

1) All the nonzero elements exist in $[\frac{rn}{r+1}]$.

2) All the nonzero elements exist in $[\frac{rn}{r+1} + 1, n]$.

3) Nonzero elements exist in both $[\frac{rn}{r+1}]$ and $[\frac{rn}{r+1} + 1, n]$.

For the first case, Hamming weight of the corresponding codewords are at least d'_1. There is no second case because submatrix $[H_L\ I_v]$ of $H_{\mathcal{C}}$ cannot be satisfied. For the last case, the number of nonzero elements in $[\frac{rn}{r+1}]$ is at least d_1 due to the checks $[H_{\mathcal{C}_1}\ O]$ and the number of nonzero elements in $[\frac{rn}{r+1} + 1, n]$ is at least d_2 due to the checks $[O\ H_{\mathcal{C}_2}]$. Therefore, Hamming weight of the corresponding codewords is at least $d_1 + d_2$ and the minimum Hamming weight of $\mathcal{C}$ is minimum of the above two cases, which completes the proof. □

The $(20, 10, 6, 3)$ binary LRC in Example 4.1 can be generalized by using Theorem 4.3 as follows.

Construction 4.4 ($d \geq 6, r = 3$)**.** *For* $4|n$*, suppose that* $\mathcal{C}_1$ *is an* $(\frac{2n}{3}, \frac{2n}{3} - \lceil \log_2(1 + \frac{2n}{3}) \rceil - 1, \geq 3)$ *shortened Hamming code obtained by shortening* $\left[1, \frac{2^{2m}-1}{3} - \frac{n}{4}\right]$, $\left[\frac{2^{2m}-1}{3} + 1, 2\left(\frac{2^{2m}-1}{3} - \frac{2n}{4}\right)\right]$*, and* $\left[2\left(\frac{2^{2m}-1}{3}\right), 2^{2m} - 1 - \frac{3n}{4}\right]$ *information bits from* $(2^{2m} - 1, 2^{2m} - 2m - 1, 3)$ *Hamming code, where* m *is a smallest integer satisfying* $\frac{3n}{4} \leq 2^{2m} - 1$*. Then,* $(\frac{3n}{4}, \frac{2n}{4} - 2\lceil \frac{\log_2(1+\frac{3n}{4})}{2} \rceil - 1, 6, 2)$ *LRC* $\mathcal{C}_1'$ *with* $\frac{n}{4}$ *disjoint repair groups is constructed. Using Theorem 4.3, an* $(\frac{3n}{4}, \frac{3n}{4} - \lceil \frac{\log_2(1+\frac{3n}{4})}{2} \rceil - 1, 6, 3)$ *LRCs* $\mathcal{C}$ *can be with* $\frac{n}{4}$ *disjoint repair groups is constructed.*

Proof. Clearly, the generator matrix of $\mathcal{C}_1'$ is a primitive polynomial $p_\alpha(x)$, where α is a primitive element of $\mathbb{F}_{2^{2m}}$. Note that check polynomial for H_L in (4.2) is $1 + x^{\frac{n}{4}} + x^{\frac{2n}{4}}$. Thus, the generator polynomial of $\mathcal{C}_1'$ is $p_\alpha(x)\ (1 + x^{\frac{3n}{4}})/(1 + x^{\frac{n}{4}} + x^{\frac{2n}{4}}) = p_\alpha(x)(1 + x^{\frac{n}{4}})$ and roots of $\mathcal{C}_1'$ is $\{1, \alpha, \alpha^2, \alpha^3, \alpha^4\}$. Thus, the minimum Hamming distance of $\mathcal{C}_1'$ is larger than or equal to 6, that is , $d_1' \geq 6$. Clearly, $d_1 \geq 3$ and $d_2 = 2$ and thus $\min\{d_1', d_1 + d_2\} = 5$. From the checks $[I\ I\ I\ I]$ in lower part of (4.2), all codewords should be even Hamming weights and thus $d = 6$. □

The optimality of the proposed LRC in Construction 4.4 is given in the following proposition.

Proposition 4.4 (Optimality of Construction 4.4)**.** *The proposed LRC is optimal if* $\lceil \log_2(1 + \frac{3n}{2}) \rceil = 2\lceil \frac{\log_2(\frac{3n}{4}+1)}{2} \rceil + 1$*, Furthermore, the proposed LRC is* r*- and* d*-optimal if* $\lceil \log_2(6n - 16) \rceil > 2\lceil \frac{\log_2(\frac{3n}{4}+1)}{2} \rceil + 1$ *for disjoint repair group.*

It can be easily proved from Proposition 2.4 and thus, I omit it.

4.4 New Constructions of Binary Linear LRCs with $d \geq 8$ Using Existing LRCs

In this section, I will show that the proposed LRCs satisfy LRCs new k-optimality for $k \geq 8$. For ≥ 8, note that there is no guarantee that (2.6) is tight for LRCs with large minimum Hamming distance $d \geq 8$. Instead, I will relax k-optimality as logarithmic

gap for k if $k_{prop1} - k \leq \log_2 n$, where k_{prop1} is maximum dimension satisfying (2.5) or (2.6) in Proposition 2.4. and it is called *near k-optimal for* for LRCs with dimension k_{const}. If it is logarithmic gap, the ratio $\frac{k_{prop1}-k_{const}}{n} = O(\frac{\log_2 n}{n})$, which becomes small for larger n.

By modifying the proof of Theorem 4.3, the other new construction of binary linear LRCs is also proposed as in the following corollary.

Corollary 4.1. *Suppose that for $(r+1)|n$, an $(\frac{n}{2}, k_1, d_1)$ linear binary codes $\mathcal{C}_1$ exists. An $(\frac{n}{2}, k_1 - \frac{n}{r+1}, d_1', \frac{r+1}{2} - 1)$ binary LRC $\mathcal{C}_1'$ can be constructed by adding $\frac{n}{r+1}$ disjoint repair groups in parity check matrix of $\mathcal{C}_1$, satisfying $\mathcal{C}_1' \subset \mathcal{C}_1$. As in Theorem 4.3, an $(n, 2k_1 - \frac{n_1}{r_1+1}, \geq \min(d_1', 2d_1), r)$ binary LRC $\mathcal{C}$ with $\frac{n}{r+1}$ disjoint repair groups is constructed with parity check matrix as*

$$H_{\mathcal{C}} = \begin{bmatrix} H_{\mathcal{C}_1} & O \\ O & H_{\mathcal{C}_1} \\ H_L & H_L \end{bmatrix}. \tag{4.4}$$

It can be proved similar to that of Theorem 4.3 and I omit proof. For $d \geq 8$, the following construction is proposed.

Construction 4.5 ($d \geq 8$, $r = 2, 3$)**.** *For $r = 2, 3$ and $(r+1)|n$, suppose that $\mathcal{C}_\sigma$ and $\mathcal{C}_\delta$ be shortened Hamming codes whose $m \times (2^m - 1)$ parity check matrix $\mathcal{H}_\sigma$ and $\mathcal{H}_\delta$ are additive representation of $[1, \alpha, \alpha^2, ..., \alpha^{\frac{n}{r+1}}]$ and $[\alpha^b, \alpha^{b+1}, \alpha^{b+2}, ..., \alpha^{b+\frac{n}{r+1}}]$, respectively, where α is a root of irreducible trinomial $x^m + x^b + 1$ of $\mathbb{F}_2^m$, $b \in [m-1]$, and m is a smallest integer satisfying $\frac{n}{r+1} \leq 2^m - 1$. Then, a $(\frac{2n}{r+1}, \frac{2n}{r+1} - \lceil \log_2(1 + \frac{n}{r+1}) \rceil - 2, 4)$ binary code $\mathcal{C}_1$ is constructed with parity check matrix*

$$H_{\mathcal{C}_1} = \begin{bmatrix} \boldsymbol{1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{1} \\ H_{\mathcal{C}_\sigma} & H_{\mathcal{C}_\delta} \end{bmatrix} \tag{4.5}$$

and $(n, \frac{2n}{3} - \lceil \log_2(1 + \frac{n}{4}) \rceil - 1, 8, 3)$ LRC $\mathcal{C}_1'$ with $\frac{n}{r+1}$ disjoint repair groups is con-

structed with the parity check matrix

$$H_{\mathcal{C}'_1} = \begin{bmatrix} \boldsymbol{1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{1} \\ H_{\mathcal{C}_\sigma} & H_{\mathcal{C}_\delta} \\ I & I \end{bmatrix}. \tag{4.6}$$

From Theorem 4.3 for $r = 2$, $(n, \frac{2n}{3} - 2\lceil \log_2(1 + \frac{n}{3}) \rceil - 2, \geq 8, 2)$ *binary LRC* $\mathcal{C}$ *is constructed using* C_1 *and an* $(\frac{n}{r+1}, \frac{n}{r+1} - \lceil \log_2(1 + \frac{n}{r+1}) \rceil, 3)$ *binary code* C_2 *whose parity check matrix* $H_{\mathcal{C}_2}$

$$H_{\mathcal{C}_2} = \begin{bmatrix} \boldsymbol{1} \\ H_{\mathcal{C}_\sigma} \end{bmatrix}. \tag{4.7}$$

From Corollary 4.1 for $r = 3$, $(n, k, d, r) = (n, \frac{3n}{4} - 2\lceil \log_2(1 + \frac{n}{4}) \rceil - 3, \geq 8, 3)$ *binary LRCs is constructed using* C_1.

Proof. First, it is easy to check that localities of the proposed LRCs are 2 and 3, respectively. In order to prove minimum Hamming distance of the proposed LRCs, it it necessary to prove two claims.

1) Minimum Hamming distance of the code with parity check matrix $H_{\mathcal{C}_1}$ is 4.

2) Minimum Hamming distance of the code with parity check matrix $H_{\mathcal{C}'_1}$ is 8.

Fo the proof of 1), I divide it to the three cases by the location of nonzero elements as in Theorem 4.3. If all the nonzero elements exist in $[\frac{n}{r+1}]$ or $[\frac{n}{r+1} + 1, \frac{2n}{r+1}]$, Minimum Hamming distance is 4 because its parity check matrix is $\begin{bmatrix} \mathbf{1} \\ H_{\mathcal{C}_\sigma} \end{bmatrix}$. For the third case, minimum Hamming distance is also at least 4 by two upper checks $\begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}$.

For the proof of 2), it is not difficult to check that $H_{\mathcal{C}'_1}$ can be modified to $\begin{bmatrix} \mathbf{1} & \mathbf{0} \\ H_{\mathcal{C}'_\sigma} + H_{\mathcal{C}'_\delta} & O \\ I & I \end{bmatrix}$ by elementary row operation. From $1 + \alpha^b + \alpha^m = 0$, I have $H_{\mathcal{C}_\sigma} + H_{\mathcal{C}_\delta} = [1+\alpha^b, \alpha+\alpha^{b+1}, ..., \alpha^{\frac{n}{r+1}} + \alpha^{b+\frac{n}{r+1}-1}] = [\alpha^m, \alpha^{m+1}, ..., \alpha^{m+\frac{n}{r+1}-1}]$.

Thus, $H_{\mathcal{C}_\sigma} + H_{\mathcal{C}_\delta}$ is also a parity check matrix of shortened Hamming code and its minimum Hamming weight is 3. From the first and last local rows, the minimum Hamming weight of $H_{\mathcal{C}'_1}$ is 8.

Based on the values of $d_1 = 4$ and $d'_1 = 8$ from 1) and 2) and and $d_2 = 3$, it is easily checked that minimum Hamming distance of $\mathcal{C}$ is 8 by Theorem 4.3 or Corollary 4.1.

Note that the check $\begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix}$ for the LRC of $r = 2$ or $\begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix}$ for for the LRC of $r = 3$ in $H_{\mathcal{C}}$ can be generated by of other check and thus, I can increase the dimension of $\mathcal{C}$ by 1. □

Note that primitive trinomial does not exist for all m. The existence of primitive trinomial for each $m < 5000$ was found in [42]. Optimality of the Construction 4.5 can be shown using Proposition 2.4 as in the following proposition and I omit the proof.

Proposition 4.5 (Optimality of Construction 4.5)**.** *If there exists a primitive trinomial with degree* $m = \lceil \log_2(1 + \frac{n}{4}) \rceil$*, the proposed construction is* r*- and* d*-optimal if* $\log_2(\frac{n^2-n+2}{2}) > 2\lceil \log_2(1 + \frac{n}{4}) \rceil + 2$ *for* $r = 2$ *and* $\log_2(\frac{9n^2-22n+8}{8}) > 2\lceil \log_2(1 + \frac{n}{4}) \rceil + 3$ *for* $r = 3$*. Also, they are near* k*-optimal.*

For some k-optimal LRCs with $d = 8$ such as $(16, 6, 8, 2)$, $(20, 8, 8, 3)$, and $(24, 11, 8, 3)$ [41] can be induced by some criteria and exhaustive search, which requires exponential complexity of codelength. However, the proposed method offers explicit constructions regardless of n though they are near k-optimal. First, a lemma is given for the proof of the following construction as:

Lemma 4.1 (Theorem 1 in [43])**.** *Let* $g_2(x)$ *be a factor polynomial of the polynomial* $g_1(x)$*, both over* $\mathbb{F}_2$*. For* $i = 1, 2$*, let* $g_i(x)$ *generate an* (n, k_i, d_i) *cyclic code. Then, the Hamming distance of the cyclic* $(2n, k)$ *code generated by* $g(x) = g_1(x)g_2(x)$ *is given as* $\min(d_1, 2d_2)$ *where* d_1 *and* d_2 *is the Hamming distance of* (n, k) *cyclic code with generator polynomial* $g_1(x)$ *and* $g_2(x)$*, respectively.*

For specific case, I can construct an optimal $(30, 12, 8, 2)$ LRC using different approach as follows.

Construction 4.6. $(n, k, d, r) = (30, 12, 8, 2)$ *optimal binary cyclic LRC can be constructed by generator polynomial* $g(x) = (x^{10} + 1)(x^2 + x + 1)^2(x^4 + x + 1)$.

Proof. First of all, it is easy to check that the locality of the code is 2 for codeword polynomial $c^{\perp}(x) = x^{20} + x^{10} + 1$ of the dual code of the LRC. In order to show minimum Hamming distance, let $g_1(x) = (x^5 + 1)(x^2 + x + 1)(x^4 + x + 1)$ and $g_2(x) = (x^5 + 1)(x^2 + x + 1)$. By Lemma 4.1, d_1 is larger than or equal to 8 by 7 consecutive roots $\{\alpha^0, \alpha^1, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6\}$, where α is a primitive element of $\mathbb{F}_{2^4}$ and d_2 is 4 by roots $\{\alpha^0, \alpha^5\, \alpha^6\}$. Therefore, the Hamming distance is at least 8. In addition, optimality can be verified by (2.6). □

For the following construction, the definition of reciprocal polynomial will be used. For $\mathbb{F}_{2^n}$, reciprocal polynomial $p(x)$ is denoted as $\bar{p}(x) = x^n p(x^{-1})$. Also, $p(x)$ is called reversible if $p(x) = \bar{p}(x)$. Note that binary reversible polynomials will be used as generator polynomial of code $\mathcal{C}_1$ for codelength $2^m + 1$ if m is odd and $2^m - 1$ if m is even. r-optimal LRCs with $d \geq 10$ and $r = 3$ will be proposed in the following theorem.

Lemma 4.2 (Theorem in [44])**.** *For* $3|n$ *and* $2s \leq \frac{n}{3}$*, suppose that an* $(n - 3s, k - 3s)$ *shortened linear code* $\mathcal{C}_s$ *can be obtained by shortening the information bits with indices in* $[s]$*,* $\left[\frac{n}{3} + 1, \frac{n}{3} + s\right]$*, and* $\left[\frac{2n}{3} + 1, \frac{2n}{3} + s\right]$ *from* $(n = 2^m \pm 1, k)$ *cyclic systematic code* $\mathcal{C}_r$ *with roots* $\{\alpha^{-1}, \alpha\}$*, where* α *is a primitive element of* $\mathbb{F}_{2^m}$*. Then, the minimum Hamming distance of* $\mathcal{C}_s$ *is at least 3. Also, the support of* $\boldsymbol{x} \in \mathcal{C}_s$ *with Hamming weight 3 is only supp*$(\boldsymbol{x}) = \{i, i + \frac{n}{3} - s, i + \frac{2n}{3} - 2s\}$ *for* $i \in [0, \frac{n}{3} - s)$ *and there is no codeword with Hamming weight 4.*

Proof. First, the minimum Hamming weight of codeword in $\mathcal{C}_r$ is 3 due to two consecutive roots $\{\alpha, \alpha^2\}$. Suppose that the codeword polynomial with Hamming weight

3 is shown to be $c(x) = 1 + x^i + x^j$. Then, $c(\alpha) = 1 + \alpha^i + \alpha^j = 0$, $c(\alpha^{-1}) = 1 + \alpha^{-i} + \alpha^{-j} = 0$, $0 < i < j < n$ and therefore, $\alpha^i + \alpha^j = 1$ and $\alpha^{i+j} = 1$, which means $i = \frac{n}{3}$ and $j = \frac{2n}{3}$. Similarly, suppose that the Hamming weight of some codeword is four and its codeword polynomial is given as $c(x) = 1 + x^i + x^j + x^k$. Then, $c(\alpha) = 1+\alpha^i+\alpha^j+\alpha^k = 0$ and $c(\alpha^{-1}) = 1+\alpha^{-i}+\alpha^{-j}+\alpha^{-k} = 0$ and thus, $\alpha^{i+j}+\alpha^{j+k}+\alpha^{k+i} = \alpha^{i+j+k}$. Then, the polynomial $a(x) = (x+\alpha^i)(x+\alpha^j)(x+\alpha^k)$ for $0 < i < j < k < n$ can be expressed as $(x^2 + \alpha^{i+j+k})(x + 1)$, which means that one of the roots of $a(x)$ is $\alpha^0 = 1$ and thus, $i = 0$, which contradicts to the assumption of $i > 0$. Therefore, there is no codeword with Hamming weight 4. In addition, it is easy to check support of $\mathcal{C}_s$ with Hamming weight 3 and nonexistence of codeword with Hamming weight 4 by properties and indices of the shortened code. □

Construction 4.7 (Binary LRCs with $d \geq 10$ and $r = 3$)**.** *For $4|n$, suppose that $\mathcal{C}_\sigma$ is an $(n_\sigma, k_\sigma, d_\sigma) = (2^m \pm 1, 2^m - 2m \pm 1, 3)$ systematic cyclic binary code with roots $\{\alpha^{-1}, \alpha\}$, where m is the smallest integer satisfying $\frac{3n}{4} < 2^m \pm 1$ and $2m \leq \frac{n}{4}$ and α is a primitive element of $\mathbb{F}_{2^m}$. Then, $\mathcal{C}_1$ is constructed by shortening the information bits with indices in $\left[\frac{2^m \pm 1}{3} - \frac{n}{4}\right]$, $\left[\frac{2^m \pm 1}{3} + 1, 2\left(\frac{2^m \pm 1}{3}\right) - \frac{n}{4}\right]$, and $\left[2\left(\frac{2^m \pm 1}{3}\right) + 1, 2^m \pm 1 - \frac{n}{4}\right]$ from $\mathcal{C}_\sigma$. Then, a $(\frac{3n}{4}, \frac{2n}{4} - 2\lceil \log_2(\frac{3n}{4} \pm 1)\rceil, d \geq 10, 2)$ LRC $\mathcal{C}_1'$ is constructed by adding $\frac{n}{4}$ local checks with disjoint repair groups. Suppose that $\mathcal{C}_2$ is an $(\frac{n}{4}, \frac{n}{4} - \lceil \log_2(1 + \frac{n}{4})\rceil, d \geq 3)$ shortened Hamming code obtained by shortening the first $2^{m'} - 1 - \frac{n}{4}$ information bits from the $(2^{m'} - 1, 2^{m'} - m' - 1, 3)$ systematic Hamming code, where m' is the smallest integer satisfying $\frac{n}{4} \leq 2^{m'} - 1$. Using Theorem 1, an $(n, \frac{3n}{4} - 2\lceil \log_2(\frac{3n}{4} \mp 1)\rceil - \lceil \log_2(1 + \frac{n}{4})\rceil, d \geq 10, 3)$ LRC $\mathcal{C}$ with $\frac{n}{4}$ disjoint repair groups is constructed.*

Proof. It is easy to check that the locality of the proposed LRC is three and d_1' is 10 by 9 consecutive roots $\{\alpha, ^{-4}, \alpha, ^{-3}, \alpha, ^{-2}, \alpha^{-1}, 1, \alpha, \alpha^2, \alpha, ^3, \alpha^4\}$ of the generator polynomial of $\mathcal{C}'_1$ $g_{\mathcal{C}'_1}(x) = p_\alpha(x)p_{\alpha^{-1}}(x)(x^{\frac{n}{4}} + 1)$ similar to those in [31], which corresponds to the first case in the proof of Theorem 4.3. The third case in the proof of Theorem 4.3 is when nonzero elements exist in both $[\frac{3n}{4}]$ and $[\frac{3n}{4} + 1, n]$. Note that the

minimum Hamming distance of $\mathcal{C}$ is at least 6 because d_1 and d_2 are 3. However, there are no codewords with Hamming weights 6 and 8 and thus, the minimum Hamming weight of $\mathcal{C}$ is at least 10. In order to prove this, Lemma 4.2 will be used.

Suppose that there is a codeword with Hamming weight 6 in $\mathcal{C}$ and then the codeword should be Hamming weight 3 in $[\frac{3n}{4}]$ and $[\frac{3n}{4}+1, n]$, respectively. However, the only support in $[\frac{3n}{4}]$ is $\{i, i+\frac{n}{4}, i+\frac{2n}{4}\}$ for $i \in [\frac{n}{4}]$ by Lemma 4.2. Regardless of the support in $[\frac{3n}{4}+1, n]$, two local checks cannot be satisfied in (4.2). Thus, there is no codeword with its Hamming weight 6. Suppose that there is a codeword with Hamming weight 8. Then, Hamming weight in $[\frac{3n}{4}+1, n]$ should be 3 because it should be less than or equal to 4 by the local checks. Also, there is no codeword with Hamming weight 4 in $[\frac{3n}{4}+1, n]$ because if exists, the support of codeword in $[\frac{3n}{4}]$ has the Hamming weight 4 and in fact, it does not exist by Lemma 4.2. For $[\frac{3n}{4}]$, there are two different cases to satisfy local checks as follows:

1) $\mathbf{x}_1 \in C_1$, where $\text{supp}(\mathbf{x}_1) = \{i, i+\frac{n}{4}, i+\frac{2n}{4}, j, k\}$ for different i, j, k satisfying $i \in [\frac{n}{4}]$ and $j, k \in [\frac{3n}{4}]$.

2) $\mathbf{x}_2 \in C_1$, where $\text{supp}(\mathbf{x}_2) = \{i, i+\frac{n}{4}, j, k, h\}, \{i, i+\frac{2n}{4}, j, k, h\}$, or $\{i+\frac{n}{4}, i+\frac{2n}{4}, j, k, h\}$ for different i, j, k, h satisfying $i \in [\frac{n}{4}]$ and $j, k, h \in [\frac{3n}{4}]$.

For the above cases, I already know that there is a nonzero codeword $\mathbf{x}_3 \in C_1$ such that $\text{supp}(\mathbf{x}_3) = \{i, i+\frac{n}{4}, i+\frac{2n}{4}\}$ from Lemma 4.2. By linearity, the sums of two codewords $\mathbf{x}_1+\mathbf{x}_3$ and $\mathbf{x}_2+\mathbf{x}_3$ should also be codewords of whose Hamming weights are 2 and 4. From Lemma 4.2, there are no codewords with Hamming weights 2 and 4 in C_1, which makes contradiction. Therefore, the minimum Hamming weight of $\mathcal{C}$ is at least 10. □

Note that the optimal LRCs with $d \geq 10$ and $r = 2$ were constructed in [29],[31] using reversible polynomial. The optimality of the Construction 4.7 without proofs is shown as follows:

Table 4.4: Optimality of proposed binary LRCs with disjoint repair group using existing LRCs

(d, r)	k	k-opt	d-opt	r-opt
$(6, 2)$	$\frac{2n}{3} - \lceil \log_2(\frac{2n}{3} + 1) \rceil - 1$	Δ	O	O
$(6, 3)$	$\frac{3n}{4} - 2\lceil \frac{(\log_2(\frac{3n}{4}+1)}{2} \rceil - 1$	Δ	O	O
$(8, 2 \text{ or } 3)$	$\frac{rn}{r+1} - 2\lceil \log_2(1 + \frac{n}{r+1}) \rceil - r$	near	Δ	O
$(8, 2)$	$12 (n = 30)$	O	O	O
$(10, 3)$	$\frac{3n}{4} - 2\lceil \log_2(1 \pm \frac{3n}{4}) \rceil$ $- \lceil \log_2(1 + \frac{n}{4}) \rceil$	near	?	O

Proposition 4.6 (Optimality of Construction 4.7)**.** *The proposed LRC is r-optimal for $n \geq 72$. Also, it is near k-optimal.*

Table 4.4 lists the constructions and optimality of the proposed binary LRCs. The proposed LRC gives us new classes of LRCs which can be explicitly constructed even for long codelength.

Chapter 5

New Constructions of Generalized RP LDPC Codes for Block Interference and Partially Regular LDPC Codes for Follower Jamming

5.1 Generalized RP LDPC Codes for a Nonergodic BI

In this section, I propose new GRP LDPC codes for TS-BSC-BIs with and without BF. First, I explain the motivation of new code design using minimum blockwise Hamming weight.

5.1.1 Minimum Blockwise Hamming Weight

The minimum blockwise Hamming weight of the code [57] is described in the following definition.

Definition 5.1. *Minimum blockwise Hamming weight d_c of code $\mathcal{C}$ is defined as*

$$d_c = \min_{\boldsymbol{c} \in \mathcal{C} \setminus \{\boldsymbol{0}\}} \left(\sum_{i \in [L]} u(wt(\boldsymbol{c}_i)) \right), \tag{5.1}$$

where $u(x)$ returns 1 if $x > 0$ and 0, otherwise.

Note that the minimum blockwise Hamming weight d_c is also called diversity order in the BF channel, which corresponds to a slope of FER curve in the coded performance. In addition, a code with minimum block Hamming weight d_c is designed to be robust against $d_c - 1$ deep fades. It is known that the minimum blockwise Hamming weight of codes with L hops is upper bounded by the Singleton-like bound as [57]

$$d_c \leq 1 + \lfloor L(1-R) \rfloor, \tag{5.2}$$

which implies a trade-off between R and d_c. If a code of $R \leq \frac{1}{L}$ has the minimum blockwise Hamming weight of $d_c = L$ in the BF channel, it is said that the code achieves full diversity. Among LDPC codes, the low-rate root LDPC codes [57] and the RP LDPC codes [58] with $R = \frac{1}{L}$ are designed to have the full diversity $d_c = L$. In contrast, the maximum value of d_c can be achieved as 2 from (5.2) if the code has $R \in (\frac{L-2}{L}, \frac{L-1}{L}]$. For example, the high-rate turbo code with $R = \frac{L-1}{L}$ and $d_c = 2$ was proposed in the BF channel [65].

In fact, the codes robust against deep fades are also advantageous for the BI channel because BI can be considered as deep fade. In addition, the BI channel in the high E_b/N_0 region can be considered as block-erasure channel as in Section 3.A of [57]. However, the high-rate code with $R = \frac{L-1}{L}$ and $d_c = 2$ does not work well in the BI channel because it should be always $I_A < R$ when BI exists and $\frac{E_b}{N_0} < \infty$. Thus, I propose new high-rate GRP LDPC codes with $R \in [\frac{L-2}{L-1}, \frac{L-1}{L})$ and $d_c = 2$ in the next subsection.

5.1.2 Construction of GRP LDPC Codes

In this chapter, I propose a GRP LDPC code with L hops and $R = \frac{b(L-2)+\beta}{b(L-1)}$, $0 \leq \beta < b$. First, the structure of its base matrix is proposed to enhance performance in the high E_b/N_0 region.

Construction 5.1 (GRP LDPC codes)**.** *Let T_i be a $b(L-1) \times b(L-1)$ upper or lower triangular low-density matrix with diagonal elements 1, which is equally row-*

partitioned with a row size b as $T_i = [T_{i,1}^T, T_{i,2}^T, ..., T_{i,(L-1)}^T]^T$, $i \in [L]$. Let D_i be a $(b-\beta) \times b(L-1)$ nonnegative integer matrix. Then, GRP LDPC codes of $R = \frac{bL(L-2)+\beta}{bL(L-1)}$ have the parity check matrix lifted from $(bL-\beta) \times b(L-1)L$ base matrix $B = [B_1, B_2, ..., B_L]$ with $B_i = [T_{i,1}^T, T_{i,2}^T, ..., T_{i,i-1}^T, D_i^T, T_{i,i}^T, ..., T_{i,(L-1)}^T]^T$.

Remind that BI channel can be considered as block-erasure channel in the high E_b/N_0 region. In the erasure channel, it is well-known that the existence of stopping set can degrade the performance of BP decoder. The proposed code is designed to avoid the stopping set in a hop while keeping the maximum values of d_c as 2 in the high E_b/N_0 region.

Theorem 5.1. *Minimum blockwise Hamming weight of the proposed GRP LDPC codes is 2 and the proposed codes do not have stopping set within one hop.*

Proof. Suppose that there exists a binary vector $\mathbf{v} = [\mathbf{v}_1, ..., \mathbf{v}_L]$, where the indices of all nonzero elements are in the i-th hop, i.e., $\text{wt}(\mathbf{v}) = \text{wt}(\mathbf{v}_i) > 0$. Note that a stopping set $\mathcal{S}$ is defined as a subset of VNs, where every CN connected to them has at least two edges emanating from the VNs in $\mathcal{S}$. To prove that there is no stopping set $\mathcal{S}$ in the i-th hop, I will show that some CNs have only one edge emanating from the VNs in $\mathcal{S}$.

For an upper triangular low-density matrix T_i, suppose that the last index of the nonzero element in $\mathbf{v}_i$ is the one lifted from the j-th column of B_i, $j \in [b(L-1)]$. Then, it is easy to prove this because there is a CN with one edge emanating from the VNs with indices in $\text{supp}(\mathbf{v}_i)$ by a permutation matrix P_I lifted from the diagonal elements 1 of $T_{i,\lceil \frac{j}{b} \rceil}^T$ in the base matrix. Therefore, the code has no stopping set in the i-th hop and $d_c = 2$ because $\mathbf{v}$ is not a codeword if the set of VNs with indices in $\text{supp}(\mathbf{v})$ does not contain $\mathcal{S}$ and thus, $d_c \geq 2$ by (5.1). It is easily checked that the maximum value of d_c is two for the code rate larger than or equal to $\frac{L-2}{L-1}$ from (5.2). For a lower triangular low-density matrix T_i, the proof can also be done similarly. □

Some integers of the base matrix remain undetermined in Construction 5.1. In

order to find the best base matrix operating well even in the low E_b/N_0 region, the modified PEXIT algorithm in [58] is applied to the proposed GRP LDPC codes as follows.

a) Modified PEXIT Algorithm

Suppose that there exists a $b_r \times b_c$ base matrix B of the protograph LDPC code $\mathcal{C}$ with $L|b_c$ and let $b_h = \frac{b_c}{L}$. In the previous section, the initial message value of LDPC codes in the fading channel is given as (2.12). Given that an all-zero codeword is transmitted, the message values can be approximately expressed as

$$m_{i,j} \sim \mathcal{N}\left(\frac{2\hat{\alpha}_i^2}{\sigma^2}, \frac{4\hat{\alpha}_i^2}{\sigma^2}\right). \tag{5.3}$$

In order to express the modified PEXIT, the four types of the MIs are used, which are $I_{E_v}(i,j)$, $I_{E_c}(i,j)$, $I_{A_v}(i,j)$, and $I_{A_c}(i,j)$ as follows:

1) $I_{E_v}(i,j)$; extrinsic MI between the message sent by V_j to C_i and the associated codeword bit, on one of the $b_{i,j}$ edges connecting V_j to C_i

2) $I_{E_c}(i,j)$; extrinsic MI between the message sent by C_i to V_j and the associated codeword bit, on one of the $b_{i,j}$ edges connecting C_i to V_j

3) $I_{A_v}(i,j)$; a priori MI between the input LLR of V_j and the associated codeword bit, on one of the $b_{i,j}$ edges connecting C_i to V_j

4) $I_{A_c}(i,j)$; a priori MI between the input LLR of C_i and the associated codeword bit, on one of the $b_{i,j}$ edges connecting C_i to V_j.

Let $J(\sigma)$ be a function given by

$$J(\sigma) = 1 - \int_{-\infty}^{\infty} \frac{e^{\frac{-(\xi-\sigma^2/2)^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}} \log_2\left(1+e^{-\xi}\right) d\xi. \tag{5.4}$$

The inverse function $J^{-1}(x)$ is given in an approximated form as

$$J^{-1}(x) = \begin{cases} 1.09542x^2 + 0.214217x + 2.33737\sqrt{x}, \\ \text{if } 0 \leq x \leq 0.3646 \\ -0.706692 \ln\left[0.386013(1-x)\right] + 1.75017x, \\ \text{otherwise.} \end{cases} \tag{5.5}$$

Suppose that the block fading coefficients are estimated as $\hat{\boldsymbol{\alpha}} = (\hat{\alpha}_1, ..., \hat{\alpha}_{b_c})$ for the $b_r \times b_c$ base matrix, where $\hat{\alpha}_{b_h(l-1)+1} = ... = \hat{\alpha}_{b_h l}$ for all $l \in [L]$. Then, the modified PEXIT is described in Algorithm 5.1.

The distributions of the fading coefficients differ depending on the channel models. The remaining analysis for the GRP and other LDPC codes with different code rates are discussed as follows.

In fact, the detailed method using PEXIT algorithm is the same as that in [58] except the initialization. In order to enhance the performance in the channel with a BI hop, initial fading coefficient in the l-th hop is set to 0, i.e., $\alpha_l = 0$ and $\alpha_i = 1$ for $i \in [L] \setminus \{l\}$. Then, find the best base matrix such that the corresponding BP threshold $(E_b/N_0)_{BP,th}$ has minimum value. By Construction 5.1 and the modified PEXIT algorithm, a GRP LDPC code with $L = 2$, $b = 3$, $\beta = 2$, and $R = \frac{1}{3}$ is constructed as

$$B_{GRP1} = \left(\begin{array}{c|c} 021 & 100 \\ 110 & 110 \\ 011 & 011 \\ 001 & 120 \end{array}\right). \tag{5.9}$$

Similarly, two GRP LDPC codes with $L = 3$ can be constructed as follows. The first

Algorithm 5.1 Modified PEXIT algorithm [58]

Require: Standard deviation of Gaussian channel noise σ, code rate R, block fading coefficients $\hat{\boldsymbol{\alpha}} = (\hat{\alpha}_1, ..., \hat{\alpha}_{b_c})$, $(b_r \times b_c)$ base matrix B, maximum iteration number I_{max}, and set of indices of hops with BI $\mathcal{U} \subset [L]$

1. **Initialization:** Set $\sigma_{ch,j}^2 = \frac{8\hat{\alpha}_j^2 R^2 E_b}{N_0} = \frac{4\hat{\alpha}_j^2}{\sigma^2}$ for $j \in [b_c] \setminus \{b_h(l-1)+i | l \in \mathcal{U}, i \in [b_h]\}$ and $\sigma_{ch,j}^2 = 0$, otherwise, with the iteration number $I = 0$.

2. **Variable node update(VNU):** For all $(i,j) \in [b_r] \times [b_c]$, calculate $I_{E_v}(i,j)$ using $I_{A_v}(i,j)$ and $\sigma_{ch,j}^2$ as

$$I_{E_v}(i,j) = J\left(\sqrt{\sum_{c=1}^{b_r}\{(b_{cj} - \delta_{ci})\left(J^{-1}(I_{A_v}(c,j))\right)^2\} + \sigma_{ch,j}^2}\right) \quad (5.6)$$

where $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$, otherwise. Then, for all $(i,j) \in [b_r] \times [b_c]$, $I_{A_c}(i,j) \leftarrow I_{E_v}(i,j)$.

3. **Check node update(CNU):** For all $(i,j) \in [b_r] \times [b_c]$, calculate $I_{E_c}(i,j)$ using $I_{A_c}(i,j)$ as

$$I_{E_c}(i,j) = 1 - J\left(\sqrt{\sum_{v=1}^{d_c}\{(b_{iv} - \delta_{vj})\left(J^{-1}(1 - I_{A_c}(i,v))\right)^2\}}\right). \quad (5.7)$$

Then, for $(i,j) \in [b_r] \times [b_c]$, $I_{A_v}(i,j) \leftarrow I_{E_c}(i,j)$.

4. **Cumulative mutual information(CMI):** Calculate I_{CMI}^j for all $j \in [b_c]$ as

$$I_{CMI}^j = J\left(\sqrt{\sum_{c=1}^{b_r}\{\left(J^{-1}\left(I_{E_c}(c,j)\right)\right)^2\} + \sigma_{ch,j}^2}\right). \quad (5.8)$$

5. **Stopping criterion:** If $I_{CMI}^j = 1$ for all $j \in [b_c]$, terminate the decoding (decoding successful). Else, if $I_{CMI}^j \neq 1$ for at least one $j \in [b_c]$ and $I = I_{max}$, terminate the decoding (decoding failure). Otherwise, go to step 2 with $I \leftarrow I + 1$.

one is constructed for $b = 2$, $\beta = 1$, and $R = \frac{7}{12}$ as

$$B_{GRP2} = \left(\begin{array}{c|c|c} 2111 & 1000 & 1000 \\ 1000 & 0100 & 0103 \\ 0100 & 2112 & 0010 \\ 3010 & 0010 & 0001 \\ 0001 & 0001 & 1112 \end{array}\right). \tag{5.10}$$

The second one is constructed for $b = 2$, $\beta = 0$, and $R = \frac{1}{2}$ as

$$B_{GRP3} = \left(\begin{array}{c|c|c} 0012 & 1000 & 1000 \\ 1120 & 0100 & 0100 \\ 1000 & 0012 & 0010 \\ 0100 & 1120 & 0001 \\ 0010 & 0010 & 0012 \\ 0001 & 0001 & 1120 \end{array}\right). \tag{5.11}$$

In the next subsection, the proposed codes of (5.10) and (5.11) will be compared with other low-rate full-diversity LDPC codes with $L = 2, 3$ and channel outage probability in (2.14).

5.2 Asymptotic and Numerical Analyses of GRP LDPC Codes

In this section, the proposed GRP LDPC codes are compared with full-diversity rate-compatible RP (RCRP) LDPC codes designed for $L = 2$ and $R = \frac{1}{3}$ in (4) of [62] and irregular RP2 (IRP2) LDPC code designed for $L = 3$ and $R = \frac{1}{3}$ in (18) of [64]. Note that it is not possible to make the same code rate of both the proposed codes of $R \geq \frac{L-2}{L-1}$ and the full-diversity codes of $R \leq \frac{1}{L}$ by definitions, if $L \geq 3$. For additional comparison, the regular protograph LDPC code of $R = 1/3$ and $d_c = 1$, whose base matrix is (4×6)-sized all-one matrix, is also simulated. In order to have asymptotic performance, BP and fading thresholds are shown in the next subsection.

Table 5.1: Channel and BP thresholds of GRP, regular, and full-diversity LDPC codes in the channels with $L = 2, 3$

L	2			3		
BP/Chan. thr.	GRP1	RCRP	Reg.	GRP2	GRP3	IRP2
BP thr. w. BI	4.364	4.736	∞	4.745	3.630	2.923
Chan. thr. w. BI	4.070			4.606	3.387	1.948
BP thr. w.o. BI	1.716	1.946	1.730	1.606	1.266	1.606
Chan. thr. w.o. BI	-0.496			0.590	0.158	-0.496

5.2.1 Asymptotic Analysis of LDPC Codes

Asymptotic analysis of the proposed and full-diversity LDPC codes can be done by fading threshold [58] using PEXIT algorithm and channel threshold by (2.14). However, there is a difference in initialization of the PEXIT algorithm as in the construction because our interest is error performance for the channel with BI. Therefore, I will consider the case where the first hop is with BI, i.e., $\alpha_1 = 0$.

For a channel with a BI hop, Table 5.1 shows that the proposed GRP LDPC codes have smaller gap between channel and BP thresholds with BI than full-diversity codes for both $L = 2$ and $L = 3$. For $L = 2$ and $R = \frac{1}{3}$, GRP1 LDPC codes have lower BP threshold than RCRP LDPC codes and regular LDPC codes do not have the BP threshold for any E_b/N_0 due to small value of $d_c = 1$. Table 5.1 also includes the thresholds for $L = 3$.

Fig. 5.1 shows the fading threshold vector (α_2, α_3) with $L = 3$ and $E_b/N_0 = 12$[dB] for the GRP LDPC codes of $R = \frac{7}{12}$ and $R = \frac{1}{2}$ and IRP2 LDPC code of $R = \frac{1}{3}$, where the first hop is with BI. For a channel with a BI hop, it is shown that the proposed GRP LDPC codes have good performance approaching to channel fading threshold, but IRP2 and RCRP LDPC codes have larger gap to the channel fading threshold.

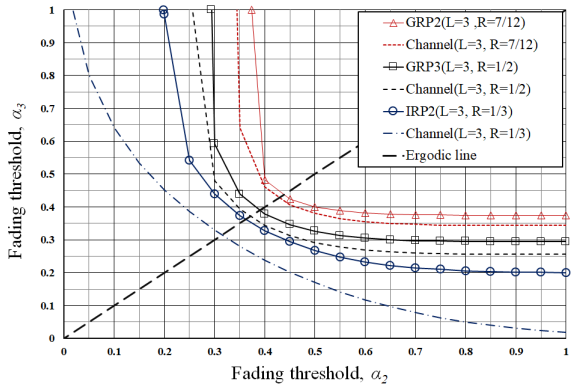

Figure 5.1: Fading threshold vector (α_2, α_3) of the channels of $E_b/N_0 = 12$[dB] and $L = 3$, where the first hop is with BI for the GRP and IRP2 LDPC codes.

5.2.2 Numerical Analysis of Finite-Length LDPC Codes

In this subsection, I show that the finite-length FER performance of the proposed GRP and full-diversity LDPC codes matches the aforementioned asymptotic analyses. All the finite-length LDPC codes have codelength 2304 with two or three hops lifted by circulant permutation matrices, which generate quasi-cyclic LDPC codes. In addition, I generate the shift values for the parity check matrix to avoid girth 6, where girth 6 for LDPC codes can degrade finite-length performance.

Furthermore, the following channel environments are assumed. First, existence of BI for each hop is assumed to follow binomial distribution with $\rho = 0.01$ in the channels without BF and with Rayleigh BF. For BP decoder, the maximum number of iterations is set to $I_{ch} = 100$. For FER, GRP LDPC codes declare an error if at least one of the coded bits are erroneous, but IRP2 and RCRP LDPC codes declare an error if at least one of the information bits are erroneous. Note that the location of information bits is known in RCRP and IRP2 LDPC codes but is not explicitly known in the proposed GRP LDPC codes.

The finite-length FER performance of regular, GRP1, and RCRP LDPC codes of $L = 2$ and GRP2, GRP3, and IRP2 LDPC codes of $L = 3$ with $\rho = 0.01$ without BF is

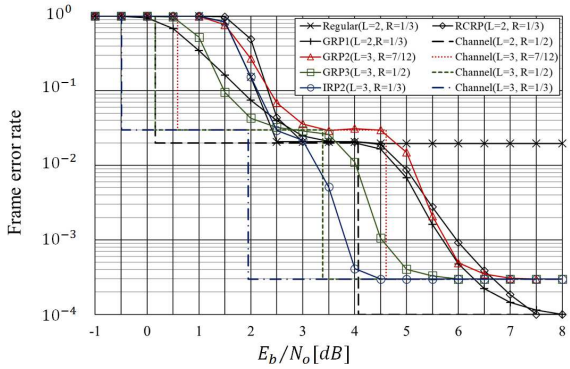

Figure 5.2: FER performance of finite-length regular, GRP1, and RCRP LDPC codes of $L = 2$ and GRP2, GRP3, and IRP2 LDPC codes of $L = 3$ and $n = 2304$ in the BI channel with $\rho = 0.01$ and without BF.

shown in Fig. 5.2. First, FER of LDPC codes has shape of stairs with two levels, which shows that the codes cannot be decoded for the cases without BI and with a BI hop if E_b/N_0 is lower than channel thresholds without BI and with a BI hop. The proposed GRP LDPC codes have good performance approaching to the channel threshold and their gaps are smaller at the lower stair, which is the case of the channel with a BI. For the full-diversity RCRP and IRP2 LDPC codes, they can correct BI for high E_b/N_0, but gap between FER and channel threshold is larger than GRP LDPC code because it has large BP threshold as in Table 5.1. For regular LDPC code, FER does not approach to the channel outage probability even for high E_b/N_0 due to small value of $d_c = 1$.

The finite-length FER performance of regular, GRP1, and RCRP LDPC codes of $L = 2$ and GRP2, GRP3, and IRP2 LDPC codes of $L = 3$ with $\rho = 0.01$ with Rayleigh BF is shown in Fig. 5.3. In this case, FER curves of GRP and IRP2 LDPC codes show high error-floor for high E_b/N_0 due to the existence of BI. GRP LDPC codes have good performance approaching to the channel outage probability but IRP2 and RCRP LDPC codes have larger gap between FER and channel outage probability than the proposed GRP LDPC codes. For regular LDPC codes, FER does not approach to the channel outage probability even for high E_b/N_0 due to small value of $d_c = 1$.

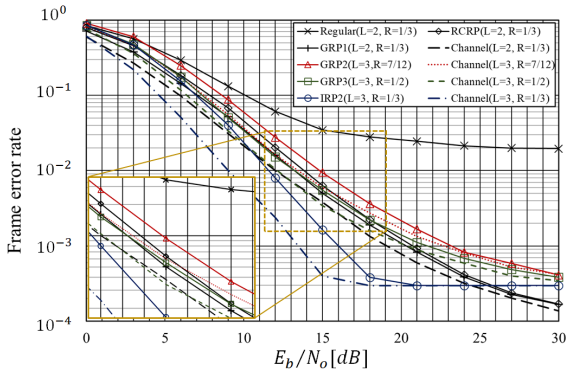

Figure 5.3: FER performance of finite-length regular, GRP1, and RCRP LDPC codes of $L = 2$ and GRP2, GRP3, and IRP2 LDPC codes of $L = 3$ and $n = 2304$ in the BI channel with $\rho = 0.01$ and Rayleigh BF.

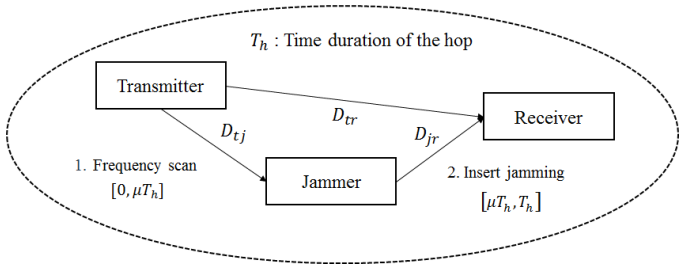

Figure 5.4: Jamming environment of the system model.

5.3 Follower Noise Jamming with Fixed Scan Speed

Follower noise jamming (FNJ), also called as repeater-back jamming, is based on the assumption that the jammer can scan the frequency which the transmitter uses. Generally, it is efficient strategy than partial band jamming in that the jammer can succeed to insert jamming with high probability. However, the jamming interval has the fundamental limitation by geometry of the transmitter, receiver, and jammer, which is explained in [51]. This relation is expressed as in Fig. 5.4.

$$T_p + T_j \leq T_h \tag{5.12}$$

, where $T_p = \frac{D_{tj}+D_{jr}-D_{tr}}{c}$, T_j is the processing time of the jammer and T_h is the interval of one hop.

For this setting, the interval that the jamming cannot approach for fixed geometry exist, which is called jamming eclipse. Two parameters describing FNJ are ρ and μ. ρ is the probability that the jamming can be actually inserted in a hop and μ is the ratio that the jamming possesses in a hop.

The scenario can be more specified by the assumption that the processing time can be variable. [52] also suggest variable jamming interval scenario, where the scenario comes from the processing time. Processing time of the jammer largely depends on the scan time. The jammer wants to find the used frequency as quick as possible, whereas it has to scan randomly due to lack of information about hopping rules. Then, the moment that finds the frequency can be different. Furthermore, I supposes that the jammer has the fixed scan speed v. Then, the processing time T_p can be expressed as

$$T_j = T^* + T_{scan} \tag{5.13}$$

$$T_{scan} = \min\left(\frac{N_{fr}}{v}, (1-\mu)T_h\right). \tag{5.14}$$

From (5.14), μ has to be divided into two terms. One is possibly front initial point of the jamming called as μ_a and the other is back initial point, called as μ_b. Then, I have

$$\mu_a = \frac{T_p + T^*}{T_h} \tag{5.15}$$

$$\mu_b = \mu_a + \frac{N_{fr}}{vT_h} \tag{5.16}$$

$$\mu_k = u[\mu_a, \min(\mu_b, 1)] \tag{5.17}$$

$$\rho = \frac{1-\mu_a}{\mu_b - \mu_a}. \tag{5.18}$$

For convenience, I suppose $T^* = 0$. By using geometry and proper v and T_h, jamming parameters are evaluated. $\delta(k,i)$ of received signal can be defined as

$$\delta(k,i) = \begin{cases} 1 & \frac{i}{K} \geq \mu_k \\ 0 & \text{otherwise.} \end{cases} \tag{5.19}$$

The remained topic is about power of jamming. The follower jamming is more energy efficient in that the jammer only can insert jamming in valid frequency band. For general case, the jammer also can select the tones of message in MFSK modulation or insert jamming into all the tones regardless of the size of M. The difference is that the jammer should divide total power as the size of M is bigger, which weakens the jamming effect. For a latter case, the average jamming statistics are expressed as

$$j \approx \mathcal{N}\left(0, \frac{N_j}{2M(1-\frac{\mu_a+\min(\mu_b,1)}{2})}\right). \tag{5.20}$$

In this chapter, only follower noise jamming and sufficiently highly powered cases are considered because jamming power is not a parameter that can be controlled. In the next section, the procedure from the channel to inducing PR-LDPC codes is introduced.

5.4 Anti-Jamming Partially Regular LDPC Codes for Follower Noise Jamming

Partially regular LDPC codes were firstly designed for unequal error protection [50]. Modified version of PR-LDPC codes for anti-jamming, AJ-PR-LDPC codes are given as below.

Construction 5.2 ((λ, d_c, d_v) AJ-PR-LDPC codes)**.** *The (λ, d_c, d_v) AJ-PR-LDPC codes with rate r is given as below. For an positive location vector $\lambda = [\lambda_1, ..., \lambda_K]$ where $\sum \lambda = 1$ and corresponding variable node degree $d_v = [d_{v,1}, ..., d_{v,K}]$ with $r(\lambda \cdot d_c) = d_v$, AJ-PR-LDPC codes has parity check matrix H that is constant weight d_c in the each row and each h_i, The i-th column of H that has block size I has to satisfy*

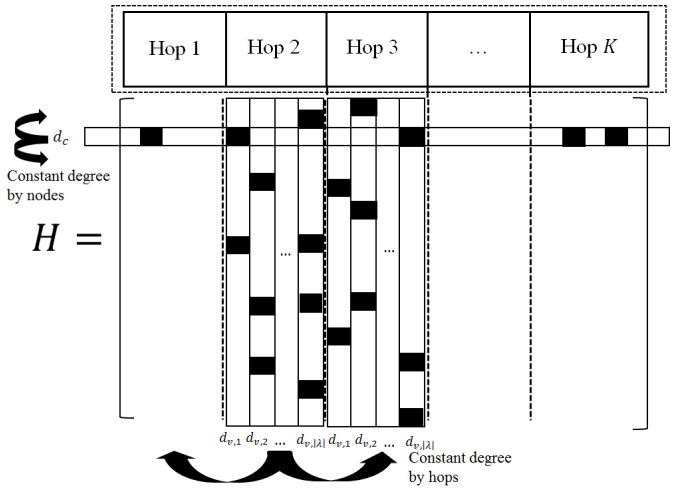

Figure 5.5: Structure of AJ-PR-LDPC codes.

$$wt(h_i) = d_{c,i}, \forall i \pmod I \in [I\sum_{i=1}^{K}\lambda_{c,i}, I\sum_{i=1}^{K+1}\lambda_{c,i}]. \tag{5.21}$$

For constructing AJ-PR-LDPC codes, the simplified channel modeling is needed.

5.4.1 Simplified Channel Model and Corresponding Density Evolution

In order to use density evolution, channel model needs to be determined. Fig. 5.6 shows the model of error distribution of hop under follower jamming. The hop is divided into 3 intervals by the SER. The leftmost interval is called as jamming eclipse, which jamming cannot approach by geometrical issue. Middle interval is where jamming may probably exist and error rate grows linearly. Rightmost interval is where the jamming is always inserted. Note that the exact values of P_a, P_b are not equal between hops due to the existence of block fading.

It is challenging to formulate density evolution of above channel environment, since it has many parameters to be considered. Instead, simplified channel model is proposed in Fig. 5.7. In this model, error is substituted with erasure and the middle interval with linear growth is changed to a series of intervals the whole of which forms

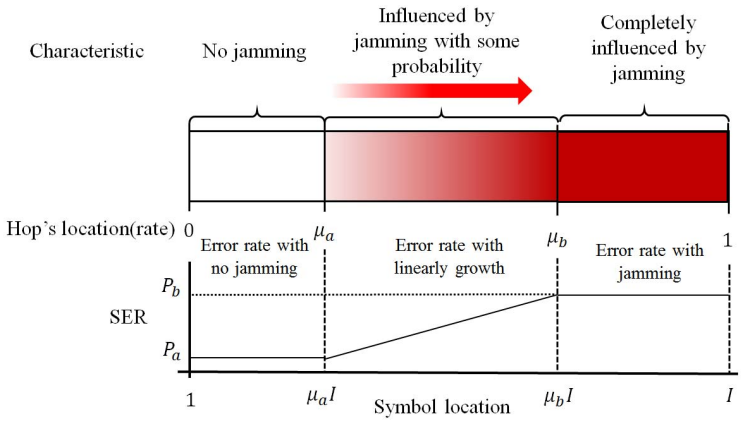

Figure 5.6: Symbol error rate in hop of follower jamming with fixed scan speed in the exact model.

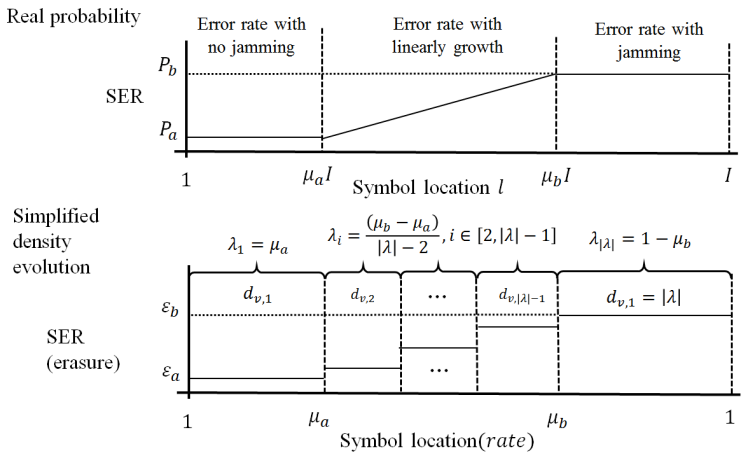

Figure 5.7: Symbol error rate in hop of follower jamming with fixed scan speed in the simplified model.

$|\lambda| - 2$ stair shapes. The corresponding density evolution of the simplified model is induced as below.

$$\epsilon_i = (\epsilon_b - \epsilon_a)\frac{i-1}{|\lambda|-1} + \epsilon_a, i \in [1, |\lambda|] \tag{5.22}$$

$$p_{l+1,i} = \epsilon_i q^{d_{v,i}-1} \tag{5.23}$$

$$q_{l+1} = 1 - (1 - \sum_{i=1}^{|\lambda|} \lambda_i p_{l,i})^{d_c - 1} \tag{5.24}$$

The initial values of ϵ_a, ϵ_b, and λ have to be determined. However their values cannot be induced from real channel rather evaluated heuristic way before constructing PR-LDPC codes, details of which are discussed next subsection.

5.4.2 Construction of AJ-PR-LDPC Codes Based on DE

The procedure of construction is summarized in Algorithm 5.2. Before constructing PR-LDPC codes, initial values of $\epsilon_a, \epsilon_b, \lambda, d_{c,max}, s_a, s_b$, and code rate r should be determined. Then, the maximum degree of variable node $d_{v,max} = [d_{v,1,max}, ..., d_{v,K,max}]$ and the check nodes $d_{c,max}$ are needed. s_a and s_b are incremental factors of simplified channel that makes channel poor. The parameters can be evaluated intuitively by designer's choice but large $\lambda, d_{c,max}$ and small s_a, s_b can make the algorithm time-consuming.

The resulting degree pair does not guarantee convergence in specific channel but has ordinal excellence than other pairs. Partial regular PEG can be implemented with modification of regular PEG or permutation of columns from the irregular PEG. In this section, the specific construction is introduced and compared to the LDPC code IEEE standards of 802.16e in the next section.

Algorithm 5.2 Construction of AJ-PR-LDPC

Require: $\epsilon_a, \epsilon_b, s_a, s_b, \lambda, d_{v,max}, d_{c,max}$, and code rate r.

Generate all the degree pair D of (d_v, d_c) which satisfy $d_v \leq d_{v,max}$, $d_c \leq d_{c,max}$.

while There exist pairs more than one **do**

Set D_n as the degree pairs that are not converged to 0 for each element of D by the proposed density evolution.

$D \leftarrow D \setminus D_n$

$\epsilon_a \leftarrow \epsilon_a + s_a$, $\epsilon_b \leftarrow \epsilon_b + s_b$.

end while

Select the remaining degree pair of D

Use partial regular PEG for generating H by the selected degree pair.

Ensure: the parity check matrix H.

Table 5.2: Jamming environment of the simulation

Cases	*Modulation*	μ_a	μ_b	ρ	E_b/N_j
No jamming	NC-MFSK	X			
Slow scan	with	3/8	11/8	5/8	-50
Fast scan	M=2,4,8,16	3/8	7/8	1	[dB]

5.5 Numerical Analysis of AJ-PR LDPC Codes

The simulation is done by NC-MFSK channel with follower jamming. The symbol sizes $M = 2, 4, 8, 16$ are used and the jamming environment divided into 3 cases; no jamming, the jamming with fast scan speed and slow scan speed. Hop size has 192 bits for all M and the scan speed is defined as proportional values of N_{fr} and T_h. The scan speed of fast speed case is $v = \frac{2N_{fr}}{T_h}$ and one of slow case is $v = \frac{N_{fr}}{2T_h}$. The parameter representing jamming power $\frac{E_b}{N_j}$ is -50[dB] if the jamming exists. It is the environment that the jamming overwhelms the message signal regardless of μ or M. The table summarizing the channel and code criterion is in Table 5.2.

The code criteria are the codelength $N = 2304$ and code rate $r = \frac{1}{2}$, which is

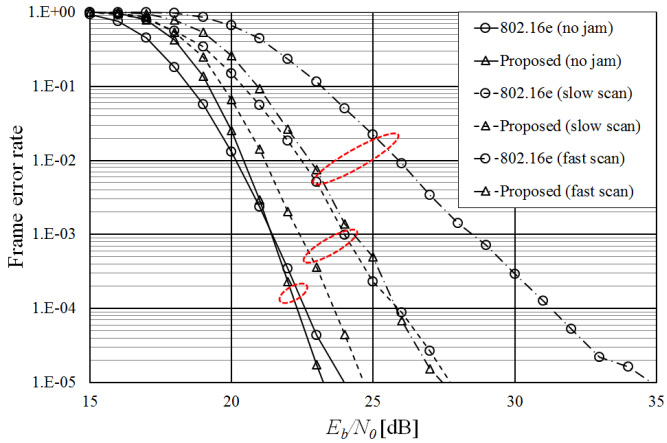

Figure 5.8: Simulation result of the proposed AJ-PR-LDPC codes with MFSK modulation of $M = 2$.

the same as 802.16e standards. There are 12 hops in the code regardless of M. AJ-PR-LDPC codes in this simulation have initial values $\epsilon_a = 0.2, \epsilon_b = 0.9, s_a = s_b = 0.01, \lambda = (\frac{3}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}), d_{c,max} = 8$, and $d_{v,max} = (8, 8, 8, 16, 16, 20)$. λ is chosen according to the fast scan case. The resulting AJ-PR-LDPC codes have parameters of $d_c = 5, \lambda = (\frac{5}{8}, \frac{2}{8}, \frac{1}{8})$, and $d_v = (2, 3, 4)$. Decoder uses BP with LLR values of MFSK and the performance comparison of the codes is in Figs. 5.8, 5.9, 5.10, and 5.11.

In these figures, red circle represents the same jamming environment. The proposed one represents AJ-PR-LDPC codes. With the same M and jamming, 802.16e has superior performance than the proposed one for all M with no jamming case. However, the AJ-PR-LDPC has more gain in two jamming case, which shows the anti-jamming effect. The largest gain is obtained in the slow scan case, which is the base of the proposed one. It is shown that the performances are the better in low $\frac{E_b}{N_0}$ as M is larger, but worse in high $\frac{E_b}{N_0}$.

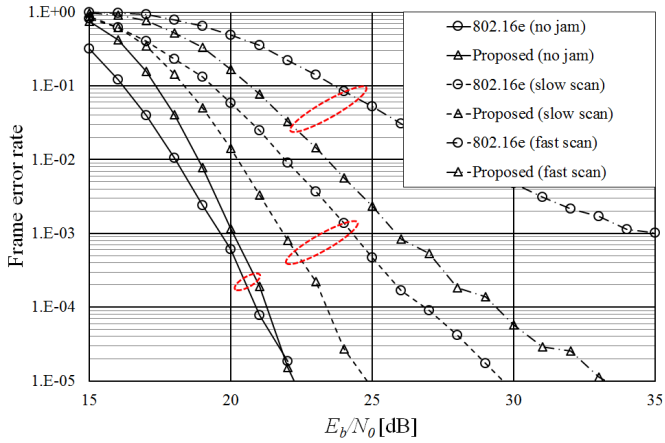

Figure 5.9: Simulation result of the proposed AJ-PR-LDPC codes with MFSK modulation of $M = 4$

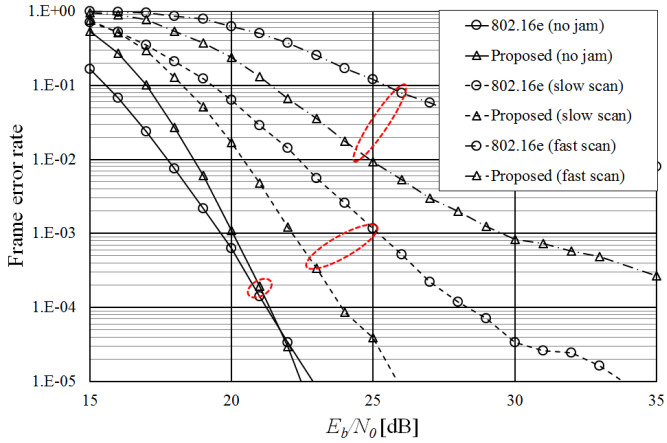

Figure 5.10: Simulation result of the proposed AJ-PR-LDPC codes with MFSK modulation of $M = 8$

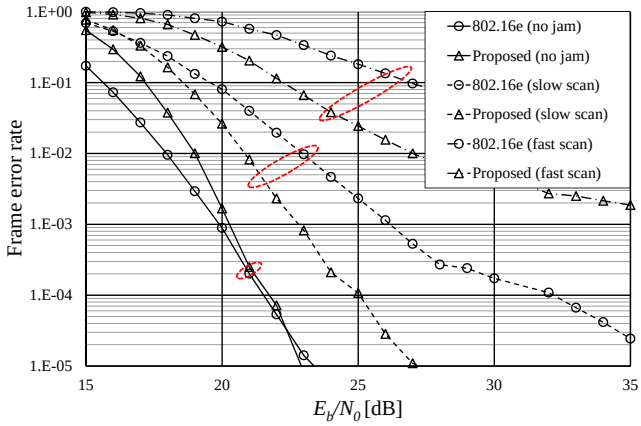

Figure 5.11: Simulation result of the proposed AJ-PR-LDPC codes with MFSK modulation of $M = 16$

Chapter 6

Conclusion

In this dissertation, new TS-AGD for cyclic codes in the erasure channel, new constructions of binary and ternary LRCs using cyclic codes and existing LRCs, and new constructions of high-rate GRP LDPC codes for a nonergodic BI and AJ-PR-LDPC codes for FNJ were studied.

First, TS-AGD algorithms for cyclic binary and cyclic MDS codes were proposed by modifying and expanding the parity check matrix. Modification criteria of the parity check matrix are proposed and the proposed TS-AGD algorithms are shown to be able to reduce the average number of iterations and the decoding complexity. The perfect codes, BCH codes, and MDS codes are considered for the proposed TS-AGD algorithms, where some of them achieve the perfect decoding. For the MDS codes, the modified decoding algorithm with expanded parity check matrix and submatrix inversion for perfect decoding is discussed. It is shown that some cyclic codes achieve the perfect decoding by the proposed TS-AGD with the expanded parity check matrix and submatrix inversion.

Second, several binary and ternary constructions of LRCs by cyclic codes and existing LRCs are proposed. Our constructions can construct binary LRCs with parameters of $4 \leq d \leq 10$ and $2 \leq r \leq 3$, most of which are optimal for k, d, and r or near k-optimal. As a future work, improved construction better than our construc-

tions will be researched. Also, improved bounds and constructions of LRCs with large Hamming distance and locality can be studied.

Third, I proposed high-rate GRP LDPC codes for channels with BI using the concept of minimum blockwise Hamming weight. The design and asymptotic analysis of the proposed GRP LDPC codes were done by the modified PEXIT algorithm. The finite-length GRP LDPC codes show good performance approaching to the channel outage probability. There remains an open problem in the design of good RP LDPC codes for channels with BI and BF, for which the minimum blockwise Hamming weight is larger than 2, that is, $3 \leq d_c \leq L$. Design and analysis of the LDPC codes for channels with BI and BF and with relatively large d_c and L can be researched.

Also, I proposed AJ-PR-LDPC codes for follower noise jamming, where I assumes the SFH and MFSK with Rayleigh block fading channel with follower jamming to imitate tactical environment. Furthermore, a new model for follower jamming with fixed scan speed in FH/SS environment is proposed. The model of probabilistic hop error distribution can be simplified with erasure stair model and it is used for density evolution for AJ-PR-LDPC. Simple algorithm can be used to derive the degree pair with ordinal excellence and PR-PEG are used to generate H. The simulation result shows that the proposed codes have better performance in the presence of jamming than 802.16e. Nonbinary codes can be more optimized solution to the cases with high M which can be future work.

Bibliography

[1] I. S. Reed, "A class of multiple-error-correcting codes and the decoding scheme," *IRE Trans. Inf. Theory*, vol. IT-4, pp. 38-49, Sep. 1954.

[2] F. J. MacWilliams, "Permutation decoding of systematic codes," *Bell Syst. Tech. J.,* vol. 43, pp. 485-505, 1964.

[3] J. Bellorado and A. Kavcic, "Low-complexity soft-decoding algorithms for Reed-Solomon codes Part I: An algebraic soft-in hard-out Chase decoder," *IEEE Trans. Inf. Theory*, vol. 56, no. 3, pp. 945-959, Mar. 2010.

[4] J. Bellorado, A. Kavcic, M. Marrow, and L. Ping, "Low-complexity soft-decoding algorithms for Reed-Solomon codes Part II: Soft-input soft-output iterative decoding," *IEEE Trans. Inf. Theory,* vol. 56, no. 3, pp. 960-967, Mar. 2010.

[5] H. D. L. Hollmann and L. M. G. M. Tolhuizen, "On parity-check collections for iterative erasure decoding that correct all correctable erasure patterns of a given size," *IEEE Trans. Inf. Theory,* vol. 53, no. 2, pp. 823-828, Jan. 2007.

[6] T. Hehn, O. Milenkovic, S. Laendner, and J. Huber, "Permutation decoding and the stopping redundancy hierarchy of cyclic and extended cyclic codes," *IEEE Trans. Inf. Theory,* vol. 54, no. 12, pp. 5308-5331, Dec. 2008.

[7] T. Hehn, J. B. Huber, O. Milenkovic, and S. Laendner, "Multiple-bases belief-propagation decoding of high-density cyclic codes," *IEEE Trans. Commun.,* vol. 58, no. 1, pp. 1-8, Jan. 2010.

[8] C. Chen, B. Bai, X. Yang, L Li, and Y Yang, "Enhancing iterative decoding of cyclic LDPC codes using their automorphism groups," *IEEE Trans. Commun.,* vol. 61, no. 6, pp. 2128-2137, Apr. 2013.

[9] K. Liu, S. Lin, and K. Abdel-Ghaffar, "A revolving iterative algorithm for decoding algebraic cyclic and quasi-cyclic LDPC codes," *IEEE Trans. Commun.,*, vol. 61, no. 12, pp. 4816-4827, Dec. 2013.

[10] M. Schwartz and A. Vardy, "On the stopping distance and the stopping redundancy of codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp.922-932, Mar. 2006.

[11] T. Etzion, "On the stopping redundancy of Reed-Muller codes," *IEEE Trans. Inf. Theory,* vol. 52, no. 11, pp. 4867-4879, Sep. 2006.

[12] J. Han, P. H. Siegel, and R. M. Roth, "Single-exclusion number and the stopping redundancy of MDS codes," *IEEE Trans. Inf. Theory*, vol. 55, no. 9, pp. 4155-4166, Sep. 2009.

[13] J. Zhang, F. W. Fu, and D. Wan, "Stopping sets of algebraic geometry codes," *IEEE Trans. Inf. Theory,* vol. 60, no. 3, pp. 1488-1495, Mar. 2014.

[14] K. Dohmen, *Improved Bonferroni Inequalities via Abstract Tubes.*, Berlin, Germany: Springer-Verlag, 2003.

[15] I. Tamo, A. Barg, S. Goparaju, and R. Calderbank, "Cyclic LRC codes and their subfield subcode," *IEEE Int. Symp. Inf. Theory (ISIT)*, pp. 1262-1266, Jun. 2015.

[16] C. J. Colbourn and J. H. Dinitz, *Handbook of Combinatorial Designs.*, New York, NY, USA: CRC Press, 2006.

[17] Y. Niho, "Multivalued cross-correlation functions between two maximal linear recursive sequence," Ph.D. dissertation, Univ. Southern Calif., Los Angeles, 1970.

[18] R. Gold, "Maximal recursive sequences with 3-valued recursive cross-correlation functions (corresp.)," *IEEE Trans. Inform. Theory*, vol. 14, no. 1, pp. 154-156, 1968.

[19] T. Kasami, "The weight enumerators for several classes of subcodes of the 2nd order binary Reed-Muller codes," *Inf. Control*, vol. 18, no. 4, pp. 369-394, 1971.

[20] A. Canteanut, P. Charpin, and H. Dobbertin, "Binary m-sequences with three-valued crosscorrelation: a proof of Welch's conjecture," *IEEE Trans. Inform. Theory*, vol. 46, no. 1, pp. 4-8, 2000.

[21] N. Boston and G. McGuire, "The weight distributions of cyclic codes with two zeros and zeta functions," *J. Symbolic Comput.*, vol. 45, no. 7, pp. 723-733, 2010.

[22] T. A. Dowling and R. J. McEliece, "Cross-correlation of reverse maximal-length shift register sequences," JPL Space Programs Summary 37-53, vol. 3, pp.192-193, 1968.

[23] E. Agrell, A. Vardy, and K. Zeger, "Upper bounds for constant-weight codes," *IEEE Trans. Inf. Theory,* vol. 46, no. 7, pp. 2373-2395, Nov. 2000.

[24] P. C. Li and G. H. J. Van Rees, "Lotto design tables," *Journal of Combinatorial Designs*, vol. 10, no. 5, pp.335-359, Aug. 2002.

[25] D. Mackay, "Fountain codes," *IEE Proceedings-Communications*, vol. 152, no. 6, pp. 1062-1068, Dec. 2005.

[26] K. V. Rashmi, N. B. Shah, and P. V. Kumar, "Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction," *IEEE Trans. Inform. Theory*, vol. 57, no. 8, pp. 5227-5239, Jul. 2011.

[27] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error Correcting Codes.*, New York, NY, USA: North-Holland, 1977.

[28] A. McGregor and O. Milenkovic, "On the hardness of approximating stopping and trapping sets," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1640-1650, Mar. 2010.

[29] S. Goparaju and R. Calderbank, "Binary cyclic codes that are locally repairable," *in Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Honolulu, USA, Jun. 2014, pp. 676-680.

[30] M. Shahabinejad, M. Khabbazian, and M. Ardakani, "An efficient binary locally repairable codes for Hadoop distributed file system," *IEEE Comm. Lett.*, vol. 18, no. 8, Aug. 2014.

[31] A. Zeh and E. Yaakobi, "Optimal linear and cyclic locally repairable codes over small fields," *Information theory workshop(ITW)*, Jerusalem:Israel, Jun. 2015.

[32] P. Huang, E. Yaakobi, H. Uchikawa, and P. H. Siegel, "Binary linear locally repairable codes," *IEEE Trans. Inf. Theory*, vol. 62 no. 11, pp. 6268-6283, Sep. 2016.

[33] J. Hao, S. T. Xia, and B. Chen, "On optimal ternary locally repairable codes," *in Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Aachen:Germany, 2017, pp. 171-175.

[34] I. Tamo, A. Barg, S. Goparaju, and R. Calderbank, "Cyclic LRC codes and their subfield subcodes," *in Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Hong Kong, China, Jun. 2015, pp. 1262-1266.

[35] I. Tamo and A. Barg, "A family of optimal locally recoverable codes," *IEEE Trans. Inf. Theory*, vol. 60, no. 8, pp. 4661-4676, May 2014.

[36] V. Cadambe and A. Mazumdar, "Bounds on the size of locally recoverable codes," *IEEE Trans. Inf. Theory*, vol.61, no.11, pp. 5785-5794, Nov. 2015.

[37] A. Wang, Z. Zhang, and D. Lin, "Bounds and constructions for linear locally repairable codes over binary fields," *in Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Aachen, Germany, Jun., 2017, pp. 2033-2037.

[38] J. Hao, S. T. Xia, and B. Chen, "Some results on optimal locally repairable codes," in Proc. *IEEE Int. Symp. Inf. Theory (ISIT)*, Barcelona, Spain, Jul. 2016, pp. 440-444.

[39] M. Shahabinejad, M. Khabbazian, and M. Ardakani, "A class of binary locally repairable codes," *IEEE Trans. Commun.*, vol 64, no. 8, pp. 3182-3193, Aug. 2016.

[40] M. Y. Nam and H. Y. Song, "Binary locally repairable codes with minimum distance at least 6 based on partial t-spreads," *IEEE Commun. Letters*, vol 21. no. 8, pp. 1683-1686, Apr. 2017.

[41] J. Ma and G. Ge, "Optimal binar linear locally repairable codes with disjoint repair groups," Arxiv:1711.07138v1 [cs.IT], 20. Nov. 2017.

[42] M. Zivkovic, ''A table of primitive binary polynomials," *Mathematics of Computation*, vol. 62, no. 205, pp.385-386, 1994.

[43] R. Morelos-Zaragoza, "A note on repeated-root cyclic codes," *IEEE Trans. Inf. Theory*, vol. 37 no. 6, pp. 1736-1737, Nov. 1991.

[44] K. Tzeng and C. Hartmann, "On the minimum distance of certain reversible cyclic codes (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 16 no. 5, pp. 644-646, Sep. 1970.

[45] M. K. Simon, J. K. Omura, R. A. Scholtz, and B. K. Levitt, *Spread Spectrum Communications Handbook*, Revised ed. New York, NY, USA: McGraw-Hill, 1994.

[46] J. Dai, D. Guo, and B. Zhang, "A BICM-MD-ID scheme in FFH system for combatting partial-band interference," *Wireless Communications and Signal Processing (WCSP)*. pp. 1-4, Oct. 2010.

[47] E. Felstead, "Follower jammer considerations for frequency hopped spread spectrum," *IEEE MILCOM.*, vol.2, 1998.

[48] A. D. Martino, *Introduction to Modern EW Systems*, Norwood, MA, USA: Artech House, 2012.

[49] J. J. Boutros, A. G. Fàbregas, E. Biglieri, and G. Zemor, "Low-density parity-check codes for nonergodic block-fading channels," *IEEE Trans. Inf. Theory*, vol. 56, pp. 4286-4300, Sep. 2010.

[50] N. Rahnavard, H. Pishro-Nik, and F. Fekri, "Unequal error protection using partially regular LDPC code," *IEEE Trans. on Commmun.*, vol. 55, pp 387-391, March 2007.

[51] D. J. Torrieri, "Fundamental limitations on repeater jamming of frequency-hopping communications," *IEEE Journal on Selected Areas in Communications*, vol. 7. pp. 569-575, May 1989.

[52] A. Hansson, J. N. Senior, and K. Wiklundh, "Performance analysis of frequency-hopping ad hoc networks with random dwell-time under follower jamming," *IEEE MILCOM.*, pp. 848 - 853, 2015.

[53] R. McEliece and W. E. Stark, "Channels with block interference," *IEEE Trans. Inf. Theory*, vol.30, no.1, pp. 44-53, Jan. 1984.

[54] W. Vijacksungsithi and K. A. Winick, "Joint channel-state estimation and decoding of low-density parity-check codes on the two-state noiseless/useless BSC block interference channel," *IEEE Trans. Commun.*, vol.53, no.4, pp. 612-622, Apr. 2005.

[55] E. Lutz, D. Cygan, M. Dippold, F. Dolainsky, and W. Papke, "The land mobile satellite communication channel-recording, statistics, and channel model," *IEEE Trans. Vehi. Tech.*, vol. 40, no.2, pp. 375-386, May 1991.

[56] C. D. Frank and M. B. Pursley, "Concatenated coding for frequency-hop spread-spectrum with partial-band interference," *IEEE Trans. Commun.*, vol.44, no.3, pp. 377-387, Mar. 1996.

[57] J. J. Boutros, A. G. I. Fabregas, E. Biglieri, and G. Zemor, "Low-density parity-check codes for nonergodic block-fading channels," *IEEE Trans. Inf. Theory*, vol.56, no.9, pp. 4286-4300, Sep. 2010.

[58] Y. Fang, G. Bi, and Y. L. Guan, "Design and analysis of root-protograph LDPC codes for non-ergodic block-fading channels," *IEEE Trans. Wireless Commun.*, vol.14, no.2, pp. 738-749, Sep. 2014.

[59] N. U. Hassan, M. Lentmaier, I. Andriyanova, and G. P. Fettweis, "Improving code diversity on block-fading channels by spatial coupling," *IEEE Inter. Symp. on Inf. Theory (ISIT)*, Honolulu:USA, Jun, 2014, pp. 2311-2315.

[60] K. D. Nguyen, A. G. I. Fabregas, and L. K. Rasmussen, "Analysis and computation of the outage probability of discrete-input block-fading channels," *in Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Nice:France, Jun. 2007, pp. 1196-1200.

[61] R. Knopp and P. A. Humblet, "On coding for block fading channels," *IEEE Trans. Inf. Theory*, vol.46, no.1, pp. 189-205, Jan. 2000.

[62] Y. Fang, Y. L. Guan, G. Bi, L. Wang, and F. C. M. Lau, "Rate-compatible root-protograph LDPC codes for quasi-static fading relay channels," *IEEE Trans. Vehi. Tech.*, vol. 65, no.4, pp. 2741-2747, Apr. 2016.

[63] Y. Fang, G. Bi, L. Wang, and F. C. M. Lau, "A survey on protograph LDPC codes and their applications," *IEEE Commun. Surveys and Tutorials*, vol. 17, no.4, pp. 1989-2016, Fourth quarter 2015.

[64] Y. Fang, S. C. Liew, and T. Wang, "Design of distributed protograph LDPC codes for multi-relay coded-cooperative networks," *IEEE Trans. Wireless Commun.*, vol.16, no.11, pp. 7235-7251, Nov. 2017.

[65] G. M. Kraidy, "High-rate turbo code design for block-fading channels," *Mediterranean Electrotechnical Conference (MELECON)*, Lemesos:Cyprus, Apr. 2016.

초 록

이 논문에서는, i) 소실 채널에서 순환 부호의 새로운 이단 자기동형 군 복호기, ii) 분산 저장 시스템을 위한 순환 부호 및 기존의 부분 접속 복구 부호(LRC)를 이용한 이진 혹은 삼진 부분 접속 복구 부호 설계법, 및 iii) 블록 간섭 환경을 위한 고부효율의 일반화된 근 프로토그래프(generalized root protograph, GRP) LDPC 부호 및 추적 재밍 환경을 위한 항재밍 부분 균일 (anti-jamming paritally regular, AJ-PR) LDPC 부호가 연구되었다.

첫번째로, 소실 채널에서 순환 부호의 새로운 이단 자기동형 군 복호기를 제안하였다. 최근 분산 저장 시스템 혹은 무선 센서 네트워크 등의 응용으로 인해 소실 채널에서의 오류 정정 부호 기법이 주목받고 있다. 그러나 많은 복호기 알고리즘은 높은 복호 복잡도 및 긴 지연으로 인해 실용적이지 못하다. 따라서 낮은 복호 복잡도 및 높은 성능을 보일 수 있는 순환 부호에서 이단 자기 동형 군 복호기가 제안되었다. 본 연구에서는 패리티 검사 행렬을 변형하고, 전처리 과정을 도입한 새로운 이단 자기동형 군 복호기를 제안한다. 제안한 복호기는 perfect 부호, BCH 부호 및 최대 거리 분리 (maximum distance separable, MDS) 부호에 대해서 분석되었다. 수치 분석을 통해, 제안된 복호 알고리즘은 기존의 자기 동형 군 복호기보다 낮은 복잡도를 보이며, 몇몇의 순환 부호 및 소실 채널에서 최대 우도 (maximal likelihood, ML)과 같은 수준의 성능임을 보인다. MDS 부호의 경우, 확장된 패리티검사 행렬 및 작은 크기의 행렬의 역연산을 활용하였을 경우의 성능을 분석한다.

두 번째로, 분산 저장 시스템을 위한 순환 부호 및 기존의 부분 접속 복구 부호(LRC)를 이용한 이진 혹은 삼진 부분 접속 복구 부호 설계법을 제안하였다. 초기

연구로서, 순환 부호 및 연접을 활용한 이진 및 삼진 LRC 설계 기법이 연구되었다. 최소 해밍 거리가 4,5, 혹은 6인 제안된 이진 LRC 중 일부는 상한과 비교해 보았을 때 최적 설계임을 증명하였다. 또한, 비슷한 방법을 적용하여 좋은 파라미터의 삼진 LRC를 설계할 수 있었다. 그 외에 기존의 LRC를 활용하여 큰 해밍 거리의 새로운 LRC를 설계하는 방법을 제안하였다. 제안된 LRC는 분리된 복구 군 조건에서 최적이거나 최적에 가까운 값을 보였다.

마지막으로, GRP LDPC 부호는 Nakagami-m 블록 페이딩 및 블록 간섭이 있는 두 상태의 이진 대칭 채널을 기반으로 한다. 이러한 채널 환경에서 GRP LDPC 부호는 하나의 블록 간섭이 발생했을 경우, 이론적 성능에 가까운 좋은 성능을 보여준다. 이러한 이론 값은 채널 문턱값이나 채널 outage 확률을 통해 검증할 수 있다. 제안된 설계에서는, 변형된 PEXIT 알고리즘을 활용하여 기초 행렬을 설계한다. 또한 AJ-PR LDPC 부호는 주파수 도약 환경에서 발생하는 추적 재밍이 있는 환경을 기반으로 한다. 채널 환경으로 MFSK 변복조 방식의 레일리 블록 페이딩 및 무작위한 지속 시간이 있는 재밍 환경을 가정한다. 이러한 재밍 환경으로 최적화하기 위해, 부분 균일 구조 및 해당되는 밀도 진화 (density evolution, DE) 기법이 활용된다. 여러 시뮬레이션 결과는 추적 재밍이 존재하는 환경에서 제안된 부호가 802.16e에 사용되었던 LDPC 부호보다 성능이 우수함을 보여준다.

주요어: 자기 동형 군 부호기, BCH 부호, 블록 페이딩, 블록 간섭, 순환 부호, 분산 저장 시스템, 오류 정정 부호, 소실 채널, 주파수 도약 대역 확산, 추적 노이즈 재밍, 반복 소실 부호기, LDPC 부호, 부분 접속 복구 부호, MDS 부호, 군통신, perfect 부호, 프로토그래프 LDPC 부호, PEXIT, 프로토그래프 근 부호, 정지 중복성

학번: 2013-20778