Coding and Decoding for Multiuser Communication Systems

Shan LU

Kyoto, Japan

November, 2013

Abstract

Coding and decoding for multiuser communication systems are investigated. In this dissertation, we consider two channel models in multiuser communication systems: multiple-access adder channel (MAAC) and two-way relay channel (TWRC).

For MAAC, we propose a coding scheme of $(k+1)$-ary error-correcting signature codes. It is used to detect the status of users in MAAC, even in the presence of channel noise. The main coding scheme is presented that given a signature matrix A and a difference matrix $D = D^{+} - D^{-}$ a priori, we obtain a larger signature matrix by replacing each element in Hadamard matrix with A, or D^{+}, or D^{-} depending on the values of elements and their locations in Hadamard matrix. The set of rows of the proposed matrix gives an error-correcting signature code. Introducing the difference matrix makes it possible to construct the error-correcting signature code whose sum rate is increased with an increase in the order of Hadamard matrix. We give binary and non-binary signature codes. They are the best error-correcting signature codes for MAAC, in the sense that they have highest sum rates known.

For TWRC, we propose a low-complexity two-user turbo decoding scheme when turbo codes are applied in two users. Simplified sum trellis is provided for two-user iterative decoding at the relay to decrease the decoding complexity. It is obtained by removing one of the states in a pair of mutual symmetrical states from a sum trellis. For the Gaussian TWRC, decoding based on simplified sum trellis reduces the decoding complexity to half of that with the sum trellis, and does not degrade decoding performance since two output sequences from the pair of mutual symmetrical states are the same. For the fading TWRC, the transition probability density function from a state to next state in simplified sum trellis is approximately computed. The approximate decoding algorithm preserves low decoding complexity over the Gaussian TWRC, without much performance degradation.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter introduces two channel models in multiuser communications, multiple-access adder channel and two-way relay channel. Our contributions to multiple-access adder channel and two-way relay channel are briefly introduced.

1.1 Multiuser Communications

While traditional problems in communications concentrate on how one can efficiently transmit information between two users (known as point to point communication), in many practical situations information is communicated among several users over the common communication medium. Such multiuser communication systems exist in satellite networks, mobile communication networks, and wireless local area networks.

The field of multiuser communications started with Claude Shannon's paper [1] on two-way channels. Since then, research on multiuser communications has been an extremely active research area, and has seen a large number of fundamental contributions, covering, not only the two-way channel studied in [1], but also "many-to-one" multiple access channel, "one-to-many" broadcast channel, interference channel, relay channel, and any combinations of these, such as two-way relay channel.

In this dissertation, we focus on two channel models in multiuser communication systems: multiple-access adder channel and two-way relay channel.

1.1.1 Multiple-Access Adder Channel (MAAC)

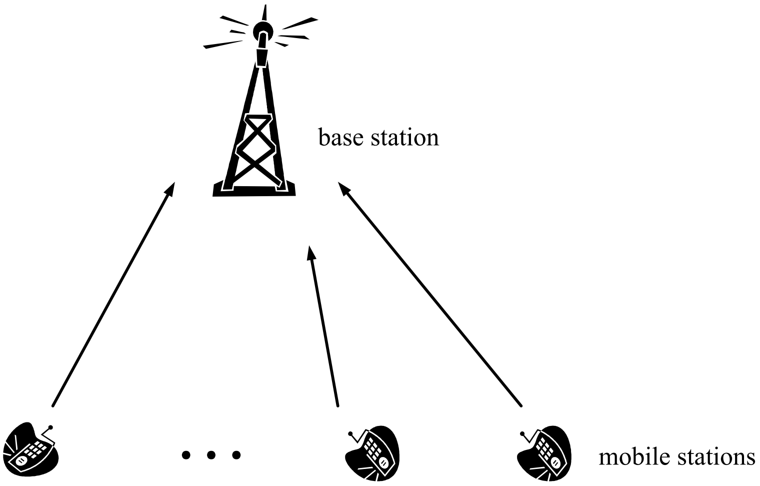


Figure 1.1: Multiple-access adder channel.

Suppose T mobile phone users are simultaneously transmitting signals to a common base station as in Fig. 1.1. This is known as the *multiple-access adder channel* (MAAC).

At the base station, the received signal is the superimposed signals from different users. To realize reliable communication, it is necessary for base station to recover T users' messages from the mixed signals, even in the presence of noise. Multiuser information theory [2] shows that multiuser coding can realize reliable communication for MAAC and has a higher total rate of transmission than traditional channel multiplexing techniques such as time-division.

The researches on multiuser coding for MAAC widely investigated in past three decades, can be divided into two cases. The first case is the multiuser codes under the assumption that all the users are active [6–11]. Another case is the multiuser codes supporting a varying number of users [12–14]. To support a varying number of users, all the users share a common all-zero codeword. When a user is idle or unactive, it is equivalent to transmitting the all-zero codeword.

Signature code is a special kind of multiuser codes supporting a varying number of users. Especially, for each user in the MAAC, two codewords are assigned, a common all-zero codeword and a user-specific non-zero codeword. When the user is active, its specific non-zero codeword is transmitted. The set of non-zero codewords is called as a signature code, if the sums of codewords from any sub-set of the set are distinct. The signature code can be used to identify the active users in MAAC. It is also can be used for fault diagnosis for multiprocessor systems, joint monitoring and routing in wireless sensor networks, location detection in hostile environments.

Signature code is related to the well-known problem of sum-distinct set and coin-weighting [15–17, 20] in additive number theory. These codes have no capability to correct errors, and only can be used for noiseless MAAC. Binary and non-binary error-correcting signature codes [25] [26] are constructed from Hadamard matrix. However, the sum rates of previous error-correcting codes are not increased with increase of code length. In this dissertation, we focus on the construction of error-correcting signature codes for MAAC with higher sum rate.

1.1.2 Two-Way Relay Channel (TWRC)

In this section, we consider another channel model in multiuser communication system: *two-way relay channel* (TWRC). Fig. 1.2 gives an example of TWRC, where two earth stations (users) exchange information via satellite (relay). There is no direct communication link between two earth stations.

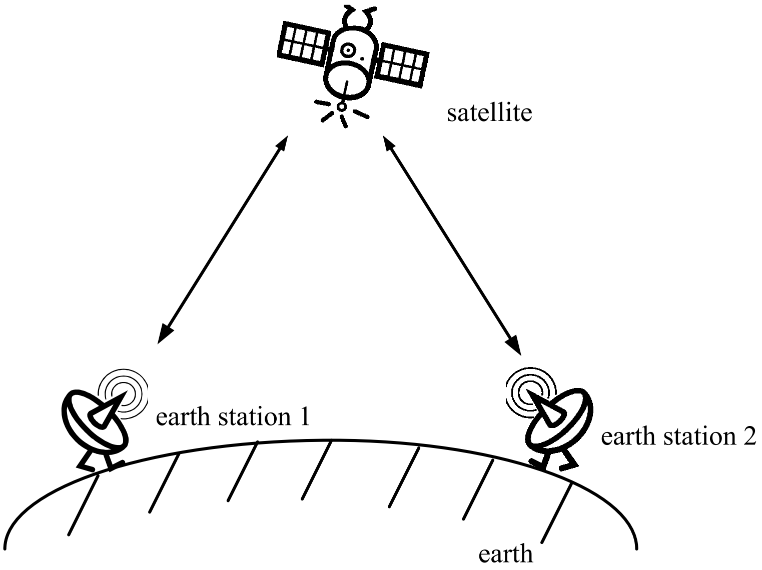


Figure 1.2: Two-way relay channel.

In TWRC, we investigate a two-time-slot transmission protocol to exchange packet once. In the first time slot, two users simultaneously transmit signals to relay, in the second time slot, the relay decodes the superimposed signal, and broadcasts an XORed message of two users' message. If the XORed message is successfully received for the two users, the users can decode the opposite user's message by an XOR operation of the local message and received XORed message. In this dissertation, we also focus on the decoding for TWRC.

1.2 Our Contributions

1.2.1 Contributions to Coding for MAAC

We now briefly introduce our contributions to coding for MAAC. We aim at the construction of error-correcting signature codes to detect the status of users for MAAC, even in the presence of channel noise.

The main coding scheme we present in Chapter 4 is as follows:

Given a signature matrix A and a difference matrix $D = D^{+} - D^{-}$ a priori, we obtain a larger signature matrix by replacing each element in Hadamard matrix with A, or D^{+}, or D^{-} depending on the values of elements and their locations in Hadamard matrix. The set of rows of proposed matrix gives an error-correcting signature code.

In this coding scheme, we extend the recursive coding scheme in Chapter 3 into a more general case as follows:

(1) the proposed signature code is $q\delta/2$-decodable signature code;

(2) signature matrix A is arbitrary, that is, code length, capability of error correction, number of users, binary or non-binary are arbitrary.

(3) when the number of rows of difference matrix D is larger than that of signature matrix A, the sum rate of the proposed signature code is increased with an increase in the order of Hadamard matrix.

We give binary and non-binary signature codes from the coding scheme. They are the best codes for MAAC,in the sense that they have highest sum rates known.

1.2.2 Contributions to Decoding for TWRC

In TWRC, we apply turbo codes for two users. For two-user turbo decoding at the relay, component decoder decodes the superimposed signal based on a sum trellis which has high decoding complexity. In this dissertation, we aim to decrease the complexity of two-user turbo decoding scheme. Simplified sum trellis are constructed to decrease the decoding complexity of two-user turbo decoding.

The main contributions to decoding for TWRC we present in Chapter 5 and 6 are as follows:

For Gaussian TWRC, simplified sum trellis is obtained by removing one of the states in a pair of mutual symmetrical states from a sum trellis. This removal reduces the decoding complexity to half of that with the sum trellis, and does not degrade decoding performance since two output sequences from the pair of mutual symmetrical states are the same.

For fading TWRC, the transition probability density function from a state to next state in simplified sum trellis is approximately computed. The approximate decoding algorithm preserves low decoding complexity over Gaussian TWRC, without much performance degradation.

The remaining part of this dissertation is organized as follows. In Chapter 2, we give preliminary of signature codes for MAAC. In Chapter 3, a recursive construction of error-correcting signature codes for MAAC is described. In Chapter 4, we give a family of error-correcting signature codes for MAAC with higher sum rate than that of codes in Chapter3. In Chapter 5, a low complexity two-user turbo decoding scheme for Gaussian TWRC is described. In Chapter 6, the low complexity two-user turbo decoding scheme is extended to fading TWRC. Finally, in Chapter 7, we close this dissertation by remarking some comments.

Chapter 2

Preliminary: Signature Codes for MAAC

This chapter describes the noiseless and noise MAAC, discusses how a signature code used for noiseless and noise MAAC to identify the status of users. Some notations and definitions are given.

2.1 Noiseless and Noise MAAC

A noiseless MAAC depicted in Fig. 2.1 is described as follows: with T users, the input alphabet to the noiseless MAAC is an integer set $\mathcal{K} \triangleq \{0, 1, 2, \ldots, k\}$, where k is a positive integer. Let $Z_i \in \mathcal{K}, i = 1, 2, \ldots, T$, be the channel inputs. Assume that Z_i maintain bit and

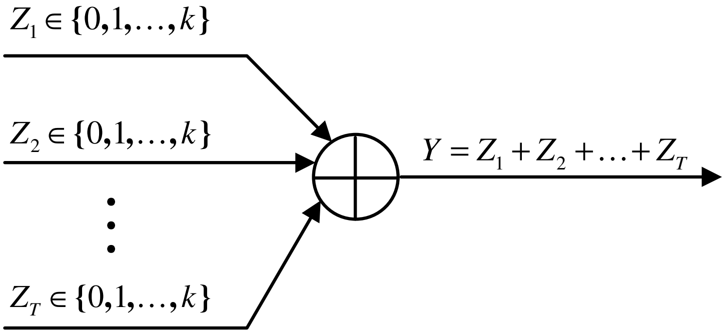
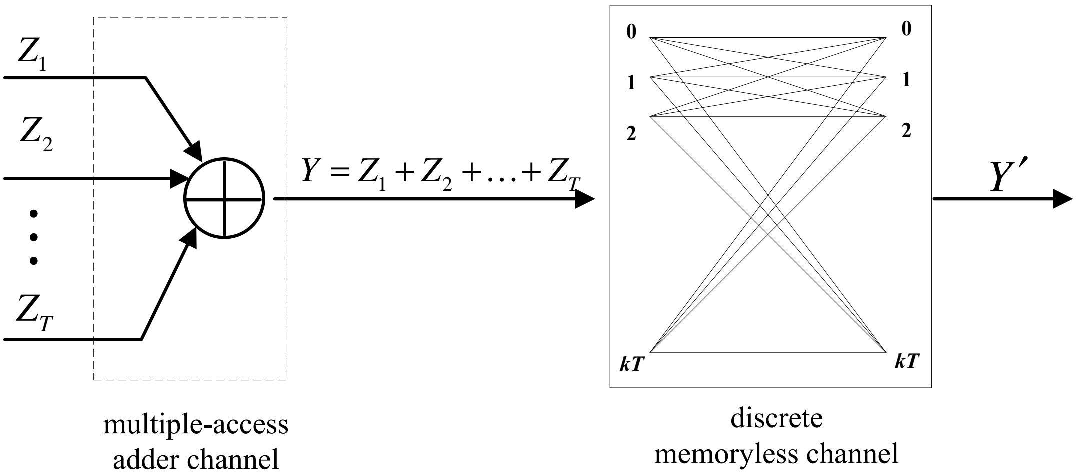

$Z_1 \in \{0,1,\ldots,k\}$
$Z_2 \in \{0,1,\ldots,k\}$
$Z_T \in \{0,1,\ldots,k\}$
$Y = Z_1 + Z_2 + \ldots + Z_T$

Figure 2.1: Noiseless multiple-access adder channel.

word synchronization. The channel output Y is given by

$$Y = Z_1 + Z_2 + \cdots + Z_T,$$

where "+" denotes the real-number addition. Clearly the output Y belongs to $\{0, 1, \ldots, kT\}$.


Z_1
Z_2
Z_T
$Y = Z_1 + Z_2 + \ldots + Z_T$
0
1
2
kT
0
1
2
kT
Y'
multiple-access adder channel
discrete memoryless channel

$Z_i \in \{0, 1, 2, \cdots, k\}, \quad (i = 1, 2, \cdots, T)$

Figure 2.2: Noisy multiple-access adder channel.

A MAAC disturbed by noise, often called a *noisy* MAAC [6] [11], is regarded as the noiseless MAAC cascaded with a discrete memoryless channel (Fig. 2.2). The discrete memoryless channel is $(kT+1)$-ary input and $(kT+1)$-ary output, and is completely described by transition probabilities for all the possible input-output pairs (i, j), $0 \leq i, j \leq kT$.

Multiuser coding is used for MAAC so that T users can communicate with a common receiver, even in the presence of noise. For noiseless MAAC, multiuser uniquely decodable codes are investigated in [6–10]. For noisy MAAC, multiuser error-correcting codes are studied in [6]. Non-binary multiuser error-correcting codes are further investigated in [11]. However, these codes [6–11] are under the assumption that all the users must be active. If the receiver do not know which users are idle in advance, then there exists a decoding ambiguity. Also, the sum rates of multiuser error-correcting codes in [6] [11] are not increased with an increase in the code length.

To support a varying number of users over the MAAC, all T users share a zero codeword $\mathbf{0}^n$ and no-transmission (idle) of the ith user corresponds to sending $\mathbf{0}^n$, where $\mathbf{0}^n$ is an all-zero row vector with length n. Multiuser codes in [13,14] with all the users sharing a common zero codeword $\mathbf{0}^n$ are studied to identify the active users for noisy MAAC. However, these codes guarantees unique identification of active users with constraint on the number of active users not exceeding m, where $m \ll T$.

2.2 Signature Codes for MAAC

In this section, we study another multiuser code to identify arbitrary number of active users for MAAC. Specifically, for the ith user, two codewords are assigned, a common all-zero

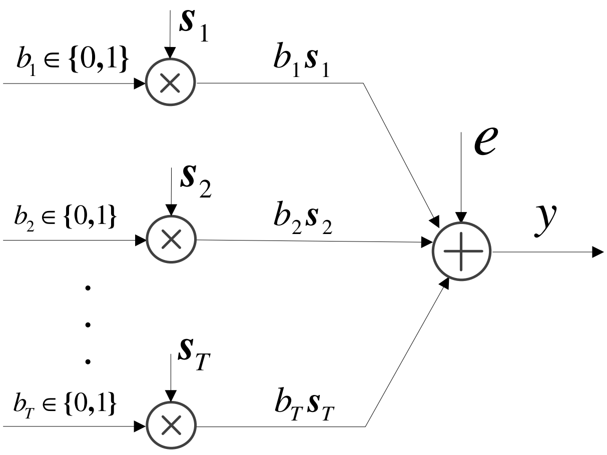


Figure 2.3: T-user transmission system with signature code over MAAC.

codeword $\mathbf{0}^n$ and a special non-zero codeword $\boldsymbol{s}_i$. The set of non-zero codewords $\mathcal{S} = \{\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_T\}$, $\boldsymbol{s}_i \in \mathcal{K}^n$ is called a *signature code*, where all the 2^T possible sums $\sum_{i=1}^{T} b_i\boldsymbol{s}_i$ are distinct, where $b_i \in \{0,1\}$.

Let $b_i \in \{0,1\}$ presents the status of ith user. The received vector is

$$\boldsymbol{y} = \sum_{i=1}^{T} b_i\boldsymbol{s}_i + \boldsymbol{e} = \boldsymbol{b}X + \boldsymbol{e}, \tag{2.1}$$

where $\boldsymbol{e}$ is error vector, $\boldsymbol{b} = [b_1, \ldots, b_T]$, and X is $T \times n$ signature matrix as

$$X = \begin{bmatrix} \boldsymbol{s}_1 \\ \boldsymbol{s}_2 \\ \vdots \\ \boldsymbol{s}_T \end{bmatrix}. \tag{2.2}$$

Definition 1 Let the *weight* of n-vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_n]$ is

$$w(\boldsymbol{y}) = \sum_{i=1}^{n} |y_i|, \tag{2.3}$$

where y_i is an integer. The *distance* between two vectors $\boldsymbol{y}$ and $\boldsymbol{y}'$ is defined by

$$d(\boldsymbol{y}, \boldsymbol{y}') = w(\boldsymbol{y} - \boldsymbol{y}'). \tag{2.4}$$

□

Definition 2 For any positive integer δ, for any non-zero T-vector $\boldsymbol{u} \in \{-1, 0, 1\}^T$, if

$$w(\boldsymbol{u}X) \geq \delta, \tag{2.5}$$

set $\mathcal{S}$ is a δ-decodable signature code, and matrix X is a δ-decodable signature matrix over $\mathcal{K}$. □

We denote δ-decodable $(k+1)$-ary signature code (matrix) with code length (the number of columns) n and cardinality (the number of rows) $T \triangleq |\mathcal{S}|$ by

$$(n, \delta, T)_k - \text{signature code (matrix)}.$$

The sum rate of the signature code is

$$R = T/n.$$

By Definition 2, the δ-decodable signature code implies that all possible 2^T sums of transmitted vectors satisfy

$$d\left(\sum_{i=1}^{T} b_i \boldsymbol{s}_i, \sum_{i=1}^{T} b_i' \boldsymbol{s}_i\right) \geq \delta$$

for any two distinct T-vectors $\boldsymbol{b} = [b_1, b_2, \ldots, b_T]$ and $\boldsymbol{b}' = [b_1', b_2', \ldots, b_T']$, where $b_i, b_i' \in \{0, 1\}$.

According to multiuser coding [6] [11], a δ-decodable signature code (matrix) for a noisy MAAC can correct errors if the weight of errors

$$w(\boldsymbol{e}) \leq \lfloor (\delta - 1)/2 \rfloor$$

where notation $\lfloor p \rfloor$ stands for the greatest integer less than or equal to p. The $(n, \delta = 1, T)_k$-code in [15–21] is said to be *uniquely decodable*, and is used for the noiseless MAAC.

To have a clear understanding of signature code for MAAC, two examples of binary $(k = 1)$ code are given below. The first code is for four users binary noiseless MAAC. The second code is for three users binary noisy MAAC, and is capable of correcting single error caused by channel noise.

Example 2.1 ([15]) The set

$$\mathcal{S} = \{101, 011, 110, 001\}$$

is a $(3, 1, 4)_1$-signature code. The uniquely decodability $(\delta = 1)$ of $\mathcal{S}$ can be observed from the fact that the 16 sums of codewords are distinct (see Table. 2.1). Thus, the decoder can uniquely determine the status b_i from the received vector $\boldsymbol{y}$.

This $(3, 1, 4)_1$-signature code can be used for a four-user binary noiseless MAAC. The rate of the signature code is

$$R = \frac{4}{3}.$$

□

Table 2.1: The decoding table for the $(3,1,4)_1$-signature code.

$\boldsymbol{b}$	$b_1\boldsymbol{s}_1$	$b_2\boldsymbol{s}_2$	$b_3\boldsymbol{s}_3$	$b_4\boldsymbol{s}_4$	$\sum_{i=1}^{4} b_i\boldsymbol{s}_i$
0000	000	000	000	000	000
0001	000	000	000	001	001
0010	000	000	110	000	110
0011	000	000	110	001	111
0100	000	011	000	000	011
0101	000	011	000	001	012
0110	000	011	110	000	121
0111	000	011	110	001	122
1000	101	000	000	000	101
1001	101	000	000	001	102
1010	101	000	110	000	211
1011	101	000	110	001	212
1100	101	011	000	000	112
1101	101	011	000	001	113
1110	101	011	110	000	222
1111	101	011	110	001	223

Table 2.2: The decoding table for the $(6,4,3)_1$-signature code.

$\boldsymbol{b}$	$b_1\boldsymbol{s}_1^\star$	$b_2\boldsymbol{s}_2^\star$	$b_1\boldsymbol{s}_3^\star$	$\sum_{i=1}^{3} b_i\boldsymbol{s}_i^\star$
000	000000	000000	000000	000000
001	000000	000000	110110	110110
010	000000	011011	000000	011011
011	000000	011011	110110	121121
100	101101	000000	000000	101101
101	101101	000000	110110	211211
110	101101	011011	000000	112112
111	101101	011011	110110	222222

Example 2.2 The set

$$\mathcal{S}^{\star} = \{101101, 011011, 110110\}$$

is a $(6,4,3)_1$-signature code. The rate of the signature code is

$$R = \frac{3}{6} = \frac{1}{2}.$$

The decoding table of $\mathcal{S}^{\star}$ is given in Table 2.2. It is verified that distance between every pair of sums of codewords in Table 2.2 is greater than or equal to 4, i.e., $\delta = 4$. Thus, the code is 4-decodable, and is used for three users binary noisy MAAC. This $(6,4,3)_1$-signature code is capable of correcting all error patterns of single error over a block of three digits. □

The signature codes are originally investigated as the problems of sum-distinct set and coin-weighting in additive number theory then applied to MAAC [15–21]. The binary signature code is equivalent to Cantor's sum-distinct set [15] and Lindström's coin weighing design [16]. Martirosyan [17] gives recursive construction of binary signature code with arbitrary code length. Mow [18] gives a generalized approach to construct binary signature code. The non-binary signature code was originally considered by Jevtić [19], and extended to arbitrary code length in [21]. The above signature codes are all recursively constructed. However, for each recursion, signature code preserves UD. Thus, these signature codes have no capability to correct errors, and only can be used for noiseless MAAC. Thus, a signature code designed to correct errors caused by channel noise and to identify status of users is required for noisy MAAC.

Binary error-correcting signature codes [25] [26] are constructed from Hadamard matrix, whose orthogonality provides the decodability of binary error-correcting signature code. However, the sum rates of previous error-correcting signature codes are not high. Also, for non-binary code, it is difficult to find such an orthogonal matrix.

In Chapter 3, we will give a recursive construction of $(k+1)$-ary error-correcting signature codes. In Chapter 4, we will give a generalized construction of $(k+1)$-ary error-correcting signature codes for MAAC. Binary and non-binary error-correcting signature codes have higher sum rate than that of binary codes in [25] [26] and non-binary code in Chapter 3.

2.3 Notations and Definitions

In this section, Kronecker product and Hadamard matrix are introduced which will be useful for constructing error-correcting signature code in the following chapters.

Kronecker product [43, p. 114] Let $A = [a_{ij}]$ be an $m \times m$ matrix and $B = [b_{ij}]$ is an $n \times n$ matrix over any field, the *Kronecker product* of A and B is the $mn \times mn$ matrix obtained from A by replacing every entry a_{ij} with matrix $a_{ij}B$. This product is written as $A \otimes B$. For example if

$$A = \begin{bmatrix} 1 & 0 \\ 1 & -1 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

we have

$$A \otimes B = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}.$$

It is proved that

$$(A \otimes B)(W \otimes X) = (AW) \otimes (BX). \tag{2.6}$$

Hadamard matrix [24, pp. 44-54,p. 422] A *Hadamard matrix* H_q of order q is a $q \times q$ matrix with elements either -1 or 1, and is defined by

$$H_q H_q^{\mathtt{T}} = qI \tag{2.7}$$

where "$\mathtt{T}$" implies the transposed matrix and I is the $q \times q$ identity matrix. In the other words, distinct rows of H_q are orthogonal, and the real inner product of any row with itself is q. Hadamard matrix H_q exists only for q being 1, 2, or a multiple of 4.

The (Sylvester-type) Hadamard matrix is recursively constructed in :

$$H_{2^j} = \begin{bmatrix} H_{2^{j-1}} & H_{2^{j-1}} \\ H_{2^{j-1}} & -H_{2^{j-1}} \end{bmatrix} \tag{2.8}$$

where $H_1 = [1]$. Obviously, in H_{2^j}, all the elements of the first row and the first column are $+1$'s.

For instance,

$$j = 2: \qquad H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}. \tag{2.9}$$

Moreover, it is easy to see that multiplying any row or column by -1 for H_q changes a Hadamard matrix into another. Thus, without loss of generality, we can assume that all the elements of the first row and the first column are $+1$'s. Such a Hadamard matrix is called *normalized*.

Let h_{ij} and h_{kj} be the elements in any two distinct rows $\boldsymbol{h}_i$ and $\boldsymbol{h}_k$ of H_q. The normalized Hadamard matrix has the following properties.

a) For $H_q(q \geq 2)$, the row $\boldsymbol{h}_i$ $(i > 1)$ has $q/2$ elements of $+1$'s and $q/2$ elements of -1's. This implies that, for a given i,

$$\sum_{\{j|h_{ij}=1\}} h_{ij} = \frac{q}{2}, \quad \sum_{\{j|h_{ij}=-1\}} h_{ij} = -\frac{q}{2}, \qquad i > 1. \tag{2.10}$$

b) For given k and i $(k \neq i \neq 1)$, among elements $h_{kj}(j = 1, 2, \ldots, q)$ whose column indices j satisfy $h_{ij} = 1$, there are $q/4$ elements of $+1$'s and $q/4$ elements of -1's. This implies that, for a given i and k,

$$\sum_{\{j|h_{ij}=1\}} h_{kj} = 0, \quad \sum_{\{j|h_{ij}=-1\}} h_{kj} = 0, \qquad i \neq k \neq 1. \tag{2.11}$$

Proof: Let

$$\sum_{\{j|h_{ij}=1,h_{kj}=1\}} h_{ij}h_{kj} = a; \quad \sum_{\{j|h_{ij}=1,h_{kj}=-1\}} h_{ij}h_{kj} = b;$$

$$\sum_{\{j|h_{ij}=-1,h_{kj}=1\}} h_{ij}h_{kj} = c; \quad \sum_{\{j|h_{ij}=-1,h_{kj}=-1\}} h_{ij}h_{kj} = d.$$

For the vector $\boldsymbol{h}_i$, we rewrite (2.10) as

$$a - b + c - d = 0. \tag{2.12}$$

For the vector $\boldsymbol{h}_k$, it also follows

$$a + b - c - d = 0. \tag{2.13}$$

Since the distinct rows in Hadamard matrix are orthogonal (2.7), we have that

$$a + b + c + d = 0. \tag{2.14}$$

Combining (2.12), (2.13), and (2.14), we have that

$$a + b = 0, \quad c + d = 0. \tag{2.15}$$

This complete the proof. □

Chapter 3

Recursive Construction of Error-Correcting Signature Codes for MAAC

Binary error-correcting signature codes are constructed in [25] [26] from Hadamard matrix, whose orthogonality provides the decodability of the codes. However, for non-binary code, it is difficult to find such an orthogonal matrix.

In this chapter, the first trial of $(k+1)$-ary error-correcting signature code for MAAC is given. The idea is from the recursive construction of $(k+1)$-ary UD signature code in [19]. By taking a submatrix of $(k+1)$-ary UD signature matrix in [19], the $(k+1)$-ary error-correcting signature code is recursively constructed from a trivial $(k+1)$-ary signature code.

The signature code's decoding procedure is also given, which consists of error correction and user identification.

3.1 $(k+1)$-Ary Signature Code

3.1.1 Encoder

We now examine the recursive construction of a $(k+1)$-ary signature matrix.

For any integer k, let $\ell = \lfloor \log_2 k \rfloor$ and $\boldsymbol{a} = [2^0, 2^1, \cdots, 2^{\ell-1}, k]$. Starting with $(\ell+1)\times 1$ matrix

$$X_1 = \boldsymbol{a}^{\mathtt{T}}, \tag{3.1}$$

jth matrix X_j for $j \geq 2$ is recursively produced by X_{j-1} as follows:

$$X_j = \begin{bmatrix} X_{j-1} & (\mathbf{0}^{T_{j-1}})^{\mathtt{T}} & X_{j-1} \\ \mathbf{0}^{n_{j-1}} & k & \boldsymbol{k}^{n_{j-1}} \\ X_{j-1} & (\boldsymbol{k}^{T_{j-1}})^{\mathtt{T}} & \bar{X}_{j-1} \end{bmatrix}. \tag{3.2}$$

The expressions in (3.2) are given as follows:

(a) $\boldsymbol{k}^p = k\mathbf{1}^p$, where $\mathbf{1}^p$ is the p-vector whose p components are all one.

(b) T_j and n_j are the numbers of the rows and the columns in matrix X_j.

(c) $\bar{X}_{j-1} = (\boldsymbol{k}^{T_{j-1}})^{\mathtt{T}}\mathbf{1}^{n_{j-1}} - X_{j-1}$.

From (3.2), note that the number of rows T_j and columns n_j obey recursion

$$\begin{aligned} T_j &= 2T_{j-1}+1, \quad T_1 = \ell+1 && (3.3) \\ n_j &= 2n_{j-1}+1, \quad n_1 = 1 && (3.4) \end{aligned}$$

for all $j \geq 2$. Therefore, it follows that

$$\begin{aligned} T_j &= 2^{j-1}\ell + 2^j - 1 && (3.5) \\ n_j &= 2^j - 1, \qquad j \geq 1. && (3.6) \end{aligned}$$

Let H'_j be the $(2^j-1)\times(2^j-1)$ sub-matrix of Hadamard matrix H_j of (2.8), obtained by deleting the first column and the first row. For example, from the H_4 of (2.9), we have that

$$H'_4 = \begin{bmatrix} -1 & 1 & -1 \\ 1 & -1 & -1 \\ -1 & -1 & 1 \end{bmatrix}. \tag{3.7}$$

The following lemma reveals that product $X_j H_j'^{\mathrm{T}}$ is a block diagonal matrix with block sub-matrices on the diagonal and all the zero matrices on the blocks off the diagonal.

Lemma 1 The product of X_j and $H_j'^{\mathrm{T}}$ is

$$\begin{aligned} X_j H_j'^{\mathrm{T}} &= -2^{j-1} \begin{bmatrix} \boldsymbol{a}^{\mathrm{T}} & O & \cdots & O \\ O & A & \cdots & O \\ \vdots & \vdots & \ddots & \vdots \\ O & O & \cdots & A \end{bmatrix} \\ &\triangleq -2^{j-1} \mathrm{diag}\{\boldsymbol{a}^{\mathrm{T}}, \underbrace{A, \ldots, A}_{2^{j-1}-1}\} \end{aligned} \tag{3.8}$$

where

$$A = \begin{bmatrix} k & 0 \\ (\boldsymbol{k}^{\ell+1})^{\mathrm{T}} - \boldsymbol{a}^{\mathrm{T}} & \boldsymbol{a}^{\mathrm{T}} \end{bmatrix} \tag{3.9}$$

and O is the rectangular zero matrices of suitable sizes. □

Proof: From X_j of (3.2), we introduce a $(T_j + 1) \times (n_j + 1)$ matrix

$$K_j \triangleq \begin{bmatrix} k & \boldsymbol{k}^{n_j} \\ (\boldsymbol{k}^{T_j})^{\mathrm{T}} & \bar{X}_j - X_j \end{bmatrix} = kJ - \begin{bmatrix} 0 & \mathbf{0}^{n_j} \\ (\mathbf{0}^{T_j})^{\mathrm{T}} & 2X_j \end{bmatrix} \tag{3.10}$$

where J denotes a $(T_j + 1) \times (n_j + 1)$ matrix and each of the elements is one.

To investigate $X_j H_j'^{\mathrm{T}}$, we first consider product $K_j H_j^{\mathrm{T}}$. Since the number of 1's and −1's are the same in every row of matrix H_j, except for the first row [24], it follows that

$$J H_j^{\mathrm{T}} = [(n_j + 1)(\mathbf{1}^{T_j+1})^{\mathrm{T}} \quad O].$$

Thus, the product of K_j and H_j^{T} is

$$K_j H_j^{\mathrm{T}} = \begin{bmatrix} (n_j + 1)k & \mathbf{0}^{n_j} \\ \boldsymbol{g}^{\mathrm{T}} & -2X_j H_j' \end{bmatrix} \tag{3.11}$$

where $\boldsymbol{g}^{\mathrm{T}} = (n_j+1)(\boldsymbol{k}^{T_j})^{\mathrm{T}} - 2X_j(\mathbf{1}^{T_j})^{\mathrm{T}}$.

On the other hand, from (3.2) and (3.10), we observe that

$$K_j = \begin{bmatrix} K_{j-1} & K_{j-1} \\ K_{j-1} & -K_{j-1} \end{bmatrix}. \tag{3.12}$$

Thus, it follows that

$$\begin{aligned} K_j H_j^{\mathrm{T}} &= 2 \begin{bmatrix} K_{j-1}H_{j-1}^{\mathrm{T}} & O \\ O & K_{j-1}H_{j-1}^{\mathrm{T}} \end{bmatrix} \\ &= \dots \\ &= 2^j \mathrm{diag}\{\underbrace{A, \dots, A}_{2^{j-1}}\} \qquad (3.13) \end{aligned}$$

where

$$A \triangleq \frac{1}{2} K_1 H_1^{\mathrm{T}} = \begin{bmatrix} k & 0 \\ (\boldsymbol{k}^{\ell+1})^{\mathrm{T}} - \boldsymbol{a}^{\mathrm{T}} & \boldsymbol{a}^{\mathrm{T}} \end{bmatrix}. \tag{3.14}$$

By comparing (3.11) and (3.13) and deleting the first row and the first column of $K_j H_j^{\mathrm{T}}$, we obtain (3.8). This proves the lemma. □

Recall that the set that is a collection of rows of matrix X_1 of (3.1), i.e., $\boldsymbol{a}^{\mathrm{T}}$, is a uniquely decodable ($\delta_1 = 1$) signature code [19], i.e., $w(\boldsymbol{u}\boldsymbol{a}^{\mathrm{T}}) \geq 1$ for any non-zero vector $\boldsymbol{u} \in \{-1, 0, 1\}^{\ell+1}$. We now show that matrix X_j of (3.2) gives a 2^{j-1}-decodable signature code.

Theorem 1 Set $\mathcal{S}_j$ that is a collection of rows of matrix X_j of (3.2), is a

$$(n_j = 2^j - 1,\ \delta_j = 2^{j-1},\ T_j = 2^{j-1}\ell + 2^j - 1)_k\text{-signature code}$$

with minimum distance 2^{j-1}. □

Proof: Each of the elements in X_j obviously belongs to $\mathcal{K}$, and T_j and n_j are given in (3.5) and (3.6).

Next we prove that $\mathcal{S}_j$ is 2^{j-1}-decodable, i.e.,

$$w(\boldsymbol{u}X_j) \geq 2^{j-1}$$

for $\boldsymbol{u} \neq \mathbf{0}^{T_j}$ and $\boldsymbol{u} \in \{-1,0,1\}^{T_j}$.

From (3.8) in Lemma 1, we have

$$\boldsymbol{u}X_j H_j'^{\mathrm{T}} = -\boldsymbol{u}2^{j-1}\mathrm{diag}\{\boldsymbol{a}^{\mathrm{T}}, \underbrace{A, \ldots, A}_{2^{j-1}-1}\}. \tag{3.15}$$

Let $\boldsymbol{u} = [\boldsymbol{u}_1, v_2, \boldsymbol{u}_3, \ldots, v_{(2^j-2)}, \boldsymbol{u}_{(2^j-1)}]$, where $\boldsymbol{u}_i \in \{-1,0,1\}^{\ell+1}, i = 1,3,\ldots,2^j-1$, and $v_i \in \{-1,0,1\}, i = 2,4,\ldots,2^j-2$. Let ϕ_i and h'_{ti} be the ith component in vectors $\boldsymbol{u}X_j$ and $\boldsymbol{h}'_t$ (the tth row of H'_j). The absolute value of the tth component in vector $\boldsymbol{u}X_jH_j'^{\mathrm{T}}$ is

$$|\sum_{i=1}^{n_j} \phi_i h'_{ti}| = \begin{cases} 2^{j-1}|\boldsymbol{u}_t\boldsymbol{a}^{\mathrm{T}}|, & \text{when } t \text{ is odd}, \\ 2^{j-1}|v_t k + \boldsymbol{u}_{t+1}((\boldsymbol{k}^{\ell+1})^{\mathrm{T}} - \boldsymbol{a}^{\mathrm{T}})|, & \text{else}. \end{cases} \tag{3.16}$$

Since $\boldsymbol{u} \neq \mathbf{0}^{T_j}$ by assumption, at least one non-zero element exists, such that for $t_0 \in \{1,2,\ldots,2^{j-1}\}$, either $\boldsymbol{u}_{t_0} \neq \mathbf{0}^{\ell+1}$ or $v_{t_0} \neq 0$. We consider two cases.

Case 1. $[\boldsymbol{u}_1, \boldsymbol{u}_3, \ldots, \boldsymbol{u}_{(2^j-1)}] \neq \mathbf{0}^{(l+1)2^{j-1}}$, i.e., t_0 is odd.

In this case, $|\sum_{i=1}^{n_j} \phi_i h'_{t_0 i}| = 2^{j-1}|\boldsymbol{u}_{t_0}\boldsymbol{a}^{\mathrm{T}}| \geq 2^{j-1}$.

Case 2. $[\boldsymbol{u}_1, \boldsymbol{u}_3, \ldots, \boldsymbol{u}_{(2^j-1)}] = \mathbf{0}^{(l+1)2^{j-1}}$, and $[v_2, v_4, \ldots, v_{(2^j-2)}] \neq \mathbf{0}^{2^{j-1}-1}$, i.e., t_0 is even.

In this case, $|\sum_{i=1}^{n_j} \phi_i h'_{t_0 i}| = 2^{j-1}|v_{t_0}k| \geq 2^{j-1}$.

Also, since all elements of H'_j are either -1 or $+1$, it follows that

$$\begin{aligned} w(\boldsymbol{u}X_j) &= \sum_{i=1}^{n_j} |\phi_i| = \sum_{i=1}^{n_j} |\phi_i h'_{t_0 i}| \\ &\geq |\sum_{i=1}^{n_j} \phi_i h'_{t_0 i}| \geq 2^{j-1}. \end{aligned} \tag{3.17}$$

Hence, set $\mathcal{S}_j$ is a 2^{j-1}-decodable signature code.

We still must show that the minimum distance of signature code $\mathcal{S}_j$ is 2^{j-1}. It is sufficient to show that there is a vector of $\boldsymbol{u} \in \{-1, 0, 1\}^{T_j}$ such that the equality in $w(\boldsymbol{u}X_j) \geq 2^{j-1}$ holds. Since the first row in X_j has 2^{j-1} weight, vector $\boldsymbol{u} = [1, 0, \ldots, \ldots, 0]$ with $w(\boldsymbol{u}) = 1$ has the desired property. This proves the theorem. □

We show an example to understand the construction procedure.

Example 3.1 Let $k = 2$. Obviously $\ell = \lfloor \log_2 2 \rfloor = 1$. The first matrix is $X_1 = [1\ 2]^{\mathsf{T}}$.

The second matrix is

$$X_2 = \begin{bmatrix} 1 & 0 & 1 \\ 2 & 0 & 2 \\ 0 & 2 & 2 \\ 1 & 2 & 1 \\ 2 & 2 & 0 \end{bmatrix}, \tag{3.18}$$

which gives $(3, 2, 5)_2$-signature code $\mathcal{S}_2 = \{101, 202, 022, 121, 220\}$.

The third matrix is

$$X_3 = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 2 & 0 & 2 & 0 & 2 & 0 & 2 \\ 0 & 2 & 2 & 0 & 0 & 2 & 2 \\ 1 & 2 & 1 & 0 & 1 & 2 & 1 \\ 2 & 2 & 0 & 0 & 2 & 2 & 0 \\ 0 & 0 & 0 & 2 & 2 & 2 & 2 \\ 1 & 0 & 1 & 2 & 1 & 2 & 1 \\ 2 & 0 & 2 & 2 & 0 & 2 & 0 \\ 0 & 2 & 2 & 2 & 2 & 0 & 0 \\ 1 & 2 & 1 & 2 & 1 & 0 & 1 \\ 2 & 2 & 0 & 2 & 0 & 0 & 2 \end{bmatrix},$$

which gives $(7, 4, 11)_2$-signature code

$$\begin{aligned} \mathcal{S}_3 = \{ \quad & 1010101, 2020202, 0220022, 1210121, 2200220, \\ & 0002222, 1012121, 2022020, 0222200, 1212101, 2202002\}. \end{aligned}$$

□

3.1.2 Decoding Rule

We now give the decoding rule of the $(k+1)$-ary signature code. Consider a T_j-user MAAC system with signature code $\mathcal{S}_j$ that is associated with signature matrix X_j of (3.2). Let the

ith user status be $b_i \in \{0,1\}$, $(i = 1, 2, \ldots, T_j)$. Assume that the channel is disturbed by the additive noise, and error vector $\boldsymbol{e}$ has weight $w(\boldsymbol{e}) \leq \lfloor (2^{j-1} - 1)/2 \rfloor$. Then the decoder receives vector

$$\boldsymbol{y} = \sum_{i=1}^{T_j} b_i \boldsymbol{s}_i + \boldsymbol{e} = \boldsymbol{b} X_j + \boldsymbol{e} \tag{3.19}$$

where $\boldsymbol{b} \triangleq [b_1, b_2, \ldots, b_{T_j}]$ and $\boldsymbol{s}_i \in \mathcal{S}_j$.

The decoding of the $(k+1)$-ary signature code includes two steps: error correction and user identification.

(*Error correction*) After multiplying $\boldsymbol{y}$ by matrix $H_j'^{\mathrm{T}}$, by Lemma 1 we have

$$\begin{aligned} \boldsymbol{y} H_j'^{\mathrm{T}} &= \boldsymbol{b} X_j H_j'^{\mathrm{T}} + \boldsymbol{e} H_j'^{\mathrm{T}} \\ &= -\boldsymbol{b} 2^{j-1} \mathrm{diag}\{\boldsymbol{a}^{\mathrm{T}}, A, \ldots, A\} + \boldsymbol{e} H_j'^{\mathrm{T}}. \end{aligned} \tag{3.20}$$

If $\boldsymbol{e}$ is a zero vector (noiseless, i.e., $w(\boldsymbol{e}) = 0$), then the elements of $\boldsymbol{y} H_j'^{\mathrm{T}}$ are multiples of 2^{j-1}, including zero.

If $\boldsymbol{e}$ has a weight, i.e., $w(\boldsymbol{e}) \neq 0$, then some elements of $\boldsymbol{y} H_j'^{\mathrm{T}}$ are not equal to the multiple of 2^{j-1} due to noise $\boldsymbol{e}$. Note that in (3.20), the weight of each element of $\boldsymbol{e} H_j'^{\mathrm{T}}$ is not greater than $\lfloor (2^{j-1} - 1)/2 \rfloor$, since $w(\boldsymbol{e}) \leq \lfloor (2^{j-1} - 1)/2 \rfloor$ by assumption and the element of $H_j'^{\mathrm{T}}$ is either 1 or -1. Thus, we correct these errors in $\boldsymbol{y} H_j'^{\mathrm{T}}$ simply by *replacing* these elements with the nearest multiple of 2^{j-1}. Specifically, let ζ_t be the tth component in vector

$\boldsymbol{y}H_j'^{\mathrm{T}}$, and

$$r_t \triangleq \zeta_t \text{ modulo } 2^{j-1}, \quad 0 \le r_t < 2^{j-1} \tag{3.21}$$

where residue r_t is non-negative. Then error correction is carried out as follows:

$$\begin{aligned} \zeta_t &\leftarrow \zeta_t, && \text{if } r_t = 0 \\ \zeta_t &\leftarrow \zeta_t - r_t, && \text{if } 0 < r_t \le \lfloor \tfrac{2^{j-1}-1}{2} \rfloor \\ \zeta_t &\leftarrow \zeta_t + 2^{j-1} - r_t, && \text{if } \lfloor \tfrac{2^{j-1}-1}{2} \rfloor < r_t < 2^{j-1}. \end{aligned} \tag{3.22}$$

(*User identification*) The status of the users is detected after the error correction of (3.22). The corrected version of $\boldsymbol{y}H_j'^{\mathrm{T}}$ is now reasonably represented by the same notation as $\boldsymbol{y}H_j'^{\mathrm{T}}$:

$$\begin{aligned} -\frac{1}{2^{j-1}}[\zeta_1, \zeta_2, \ldots, \zeta_{n_j}] &\triangleq -\frac{1}{2^{j-1}}\boldsymbol{y}H_j'^{\mathrm{T}} \\ &= \boldsymbol{b}\ \mathrm{diag}\{\boldsymbol{a}^{\mathrm{T}}, A, \ldots, A\}. \end{aligned} \tag{3.23}$$

To explain the decoding of (3.23), observe A of (3.9). Let $[\eta_1, \eta_2] \triangleq [c, \boldsymbol{v}]A$. Binary vector $\boldsymbol{v}$ can be decoded, since $X_1 = \boldsymbol{a}^{\mathrm{T}}$ is uniquely decodable [19]. After $\boldsymbol{v}$ is found, it follows that $c = (\eta_1 + \eta_2 - \boldsymbol{v}(\boldsymbol{k}^{\ell+1})^{\mathrm{T}})/k$. Based on this observation, from (3.23) we have the following detected status of users:

a) Binary vector $[b_{(\ell+2)i+1}, \ldots, b_{(\ell+2)i+\ell+1}]$ is decoded from the value of $-\zeta_{2i-1}/2^{j-1}$, $i = 0, 1, 2, \ldots, 2^{j-1}-1$, since $X_1 = \boldsymbol{a}^{\mathrm{T}}$ is uniquely decodable.

b) Bit $b_{(\ell+2)i}$ is detected:

$$b_{(\ell+2)i} = -(\zeta_{2i-1}/2^{j-1} + \zeta_{2i}/2^{j-1} + [b_{(\ell+2)i+1}, \ldots, b_{(\ell+2)i+\ell+1}][(\boldsymbol{k}^{\ell+1})^{\mathrm{T}}])/k \tag{3.24}$$

$$i = 1, 2, \ldots, 2^{j-1} - 1.$$

By the above decoding rule, the decoder can correct the transmission errors caused by the channel noise and uniquely resolve the received superimposed vector into the transmitted codewords that is the status of the corresponding users.

Note that the decoding operation only requires the multiplication of received signal $\boldsymbol{y}$ by binary matrix H'_j (see (3.20)), the modulo operation of the components of $\boldsymbol{y}H'_j$ (see (4.32)), and the integral division for user detection (see a), b) above).

Example 3.2 In this example, we apply the decoding rule to $(7,4,11)_2$-signature code $\mathcal{S}_3$ (Example 3.1) as follows:

Assume that the status of users is given by

$$[b_1, b_2, \ldots, b_{11}] = [0,1,1,0,0,1,1,1,0,0,1]$$

where users 2, 3, 6, 7, 8, and 11 are active. When the channel is disturbed by error vector $\boldsymbol{e} = [1,0,0,0,0,0,0]$, the received vector is

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{s}_2 + \boldsymbol{s}_3 + \boldsymbol{s}_6 + \boldsymbol{s}_7 + \boldsymbol{s}_8 + \boldsymbol{s}_{11} + \boldsymbol{e} \\ &= [8,4,7,8,5,8,9]. \end{aligned}$$

By multiplying vector $\boldsymbol{y}$ with matrix $H_3'^{\mathrm{T}}$, we get

$$\begin{aligned} \boldsymbol{y}H_3'^{\mathrm{T}} &= [8,4,7,8,5,8,9]H_3'^{\mathrm{T}} \\ &= [-9,-7,-1,-11,-13,1,-9]. \end{aligned} \tag{3.25}$$

By changing the elements in (3.25) into the nearest multiple of 4, we have the corrected version of $[-8,-8,0,-12,-12,0,-8]$.

Moreover, when divided by -4, the above vector becomes $[2, 2, 0, 3, 3, 0, 2]$. The elements in the vector with odd indices are mapped by a decoding table, since $X_1 = \boldsymbol{a}^{\mathrm{T}} = [1, 2]^{\mathrm{T}}$ is uniquely decodable. The table is $0 \rightarrow 00, 1 \rightarrow 10, 2 \rightarrow 01, 3 \rightarrow 11$. Therefore, we have

$$\begin{aligned} [b_1, b_2] &= [0, 1] \\ [b_4, b_5] &= [0, 0] \\ [b_7, b_8] &= [1, 1] \\ [b_{10}, b_{11}] &= [0, 1]. \end{aligned}$$

From (3.24), it follows that

$$\begin{aligned} b_3 &= (2 + 0 - [b_4, b_5][2, 2]^{\mathrm{T}})/2 = 1 \\ b_6 &= (3 + 3 - [b_7, b_8][2, 2]^{\mathrm{T}})/2 = 1 \\ b_9 &= (0 + 2 - [b_{10}, b_{11}][2, 2]^{\mathrm{T}})/2 = 0. \end{aligned}$$

Thus, decoding with error correction and user identification has been successfully completed.

□

3.2 Binary Signature Code

In this section, we propose a recursive construction of a binary signature code.

For $j \geq 0$, let X_j be the jth binary square matrix of order 2^j. The first matrix is

$$X_0 = [1]. \tag{3.26}$$

For each $j \geq 1$, X_j is constructed from X_{j-1} of order 2^{j-1} by recursion

$$X_j = \begin{bmatrix} X_{j-1} & X_{j-1} \\ X_{j-1} & \bar{X}_{j-1} \end{bmatrix}, \tag{3.27}$$

where matrix $\bar{X}_{j-1}$ indicates the complement of X_{j-1}, defined by

$$\bar{X}_{j-1} = J - X_{j-1}.$$

Here J is a $2^{j-1} \times 2^{j-1}$ square matrix, and each element is one.

Theorem 2 Set $\mathcal{S}_j$, composed of rows of matrix X_j of (3.27), is a

$$(n_j = 2^j, \delta_j = 2^{j-1}, T_j = 2^j)_1\text{-signature code}$$

with minimum distance 2^{j-1}. □

The proof of Theorem 2 resembles that of Theorem 1.

Note that the $(k+1)$-ary signature code in Theorem 1 can be deduced to a binary $(2^j - 1, 2^{j-1}, 2^j - 1)_1$-signature code when $k = 1$. Theorem 2 gives another binary $(2^j, 2^{j-1}, 2^j)_1$-signature code.

The decoding scheme of the binary signature code in Theorem 2 is omitted because it resembles that of non-binary code in Section 3.1.2.

3.3 Conclusion

We gave recursive construction of $(k+1)$-ary error-correcting signature code to identify users for MAAC, even in the presence of channel noise. Our recursion is originally from a trivial signature code. In the $(j-1)$th recursion, from a signature code with minimum distance of 2^{j-2}, we obtained a longer and larger signature code with minimum distance of 2^{j-1}. We also described the signature code's decoding procedure, which consists of error correction and user identification.

It is obviously that there are some constraints in the proposed code, such as capability of error correction, the code length and the number of users multiplying by the number of recursion, low sum rate since its particular recursive scheme. Nevertheless, the structure of coding matrix in this chapter leads us to obtain a more generalized coding scheme without these constraints.

Chapter 4

A Family of Error-Correcting Signature Codes for MAAC

In this chapter, we propose a generalized coding scheme of $(k+1)$-ary error-correcting signature codes for noisy MAAC. Given a signature matrix A and a difference matrix $D = D^+ - D^-$ *a priori*, we obtain a larger signature matrix by replacing each element in Hadamard matrix with A, or D^+, or D^- depending on the values of elements and their locations in Hadamard matrix. The set of rows of proposed matrix gives an error-correcting signature code. Introducing the difference matrix makes it possible to construct error-correcting signature code whose sum rate is increased with an increase in the order of Hadamard matrix. We give either binary or non-binary signature codes.

This coding scheme extends the recursive coding scheme into a more general case as follows:

(1) the proposed signature code is $q\delta/2$-decodable signature code;

(2) signature matrix A is arbitrary, that is, code length, capability of error correction, number of users, binary or non-binary are arbitrary.

(3) when the number of rows of difference matrix D is larger than that of signature matrix A, the sum rate of the proposed signature code is increased with an increase in the order of Hadamard matrix.

4.1 Main Theorem

In this section, we give a coding scheme of error-correcting signature code.

Before going on, some notations are prepared. Matrix D whose elements belong to $\mathcal{K}^{\pm} \triangleq \{0, \pm 1, \pm 2, \dots, \pm k\}$ is referred to as *difference matrix* which can be expressed by a difference form as $D = D^{+} - D^{-}$, where D^{+} and D^{-} are component matrices whose elements belong to $\mathcal{K}$. Let d_{ij}, d_{ij}^{+} and d_{ij}^{-} be the elements of matrix D, D^{+} and D^{-}, respectively. Given any difference matrix D, we can always set

$$\begin{aligned} d_{ij}^{+} &= d_{ij}, \quad d_{ij}^{-} = 0, \qquad \text{if } \ d_{ij} \geq 0 \\ d_{ij}^{+} &= 0, \qquad d_{ij}^{-} = |d_{ij}|, \ \ \text{if } \ d_{ij} < 0. \end{aligned} \tag{4.1}$$

Definition 3 For any positive integer δ, $T_d \times n$ matrix D is a *δ-decodable difference matrix* over $\mathcal{K}^{\pm}$ if it holds

$$w(\boldsymbol{u}D) \geq \delta \tag{4.2}$$

for any non-zero T-vector $\boldsymbol{u} \in \{-1, 0, 1\}^{T_d}$. □

We denote by $\langle n, \delta, T_d \rangle_k$ a δ-decodable $(2k+1)$-ary difference matrix with the number of columns n and the number of rows T_d. An $\langle n, 1, T_d \rangle_k$-difference matrix is said to be UD.

Then, we give a definition of matrix-selection product which is an operation to obtain a larger block matrix from smaller matrices.

Definition 4 Let A be a $T_a \times n$ signature matrix, D be a $T_d \times n$ difference matrix, and H_q be a $q \times q$ normalized Hadamard matrix ($q \geq 2$). We define matrix-selection product

$$\begin{aligned} H_q \bar{\otimes} [A|D] &\triangleq X_q \\ &= \begin{bmatrix} X_{11} & \dots & X_{1q} \\ \vdots & & \vdots \\ X_{q1} & \dots & X_{qq} \end{bmatrix} \end{aligned} \tag{4.3}$$

where

$$X_{ij} \triangleq h_{ij} \bar{\otimes} [A|D] = \begin{cases} A, & \text{if } i = 1 \\ D^{+}, & \text{if } i > 1, h_{ij} = 1 \\ D^{-}, & \text{if } i > 1, h_{ij} = -1. \end{cases} \tag{4.4}$$

□

Note that X_q is a $(T_a + (q-1)T_d) \times qn$ matrix.

Definition 5 Let $\Phi = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_q]$ be a qn-vector, where $\boldsymbol{\phi}_j$ is an n-vector. The block multiplication of qn-vector Φ and q-vector $\boldsymbol{h}_k$ is defined as

$$\Phi \boxtimes \boldsymbol{h}_k^{\mathtt{T}} \triangleq \sum_{j=1}^{q} \boldsymbol{\phi}_j h_{kj}. \tag{4.5}$$

□

Lemma 2 The block multiplication of $(T_a + (q-1)T_d) \times qn$ matrix X_q of (4.3) and $q \times q$ matrix $H_q^{\mathtt{T}}$ is a $(T_a + (q-1)T_d) \times qn$ matrix

$$X_q \boxtimes H_q^{\mathtt{T}} = \frac{q}{2} \begin{bmatrix} 2A & \mathrm{O}_a & \ldots & \mathrm{O}_a \\ (D^+ + D^-) & D & \vdots & \mathrm{O}_d \\ \vdots & \vdots & \ddots & \vdots \\ (D^+ + D^-) & \mathrm{O}_d & \vdots & D \end{bmatrix} \tag{4.6}$$

where O_a is a $T_a \times n$ zero matrix, and O_d is a $T_d \times n$ zero matrix.

Proof : The submatrix in $X_q \boxtimes H_q^{\mathtt{T}}$ is

$$\begin{aligned} M_{ik} &\triangleq (\boldsymbol{h}_i \bar{\otimes} [A|D]) \boxtimes \boldsymbol{h}_k^{\mathtt{T}} \\ &= [X_{i1}, X_{i2}, \ldots, X_{iq}] \boxtimes \boldsymbol{h}_k^{\mathtt{T}} \\ &= \sum_{j=1}^{q} X_{ij} h_{kj}, \quad 1 \leq i, k \leq q. \end{aligned} \tag{4.7}$$

For $i = 1$, it follows that

$$M_{1k} = \begin{cases} \sum\limits_{j=1}^{q} A h_{1j} = qA, & k = 1, i = 1 \\ \sum\limits_{j=1}^{q} A h_{kj} = \mathrm{O}_a, & k = 2, \ldots, q, i = 1, \end{cases} \tag{4.8}$$

since $X_{1j} = A$.

For a given i, $1 < i \leq q$, we have that

$$\begin{aligned} M_{ik} &= \sum_{j=1}^{q} h_{ij} \bar{\otimes} [A|D] h_{kj} \\ &= \sum_{j=1}^{q} h_{kj} D^{\text{sign}(h_{ij})} \\ &= D^{+} \cdot \sum_{\{j|h_{ij}=1\}} h_{kj} + D^{-} \cdot \sum_{\{j|h_{ij}=-1\}} h_{kj} \\ &= \begin{cases} (q/2)(D^{+} + D^{-}), & k = 1, 1 < i \leq q \\ (q/2)D, \ k = i, 1 < k \leq q, 1 < i \leq q \\ \text{O}_d, & k \neq i, 1 < k \leq q, 1 < i \leq q \end{cases} \end{aligned} \tag{4.9}$$

where sign$(\cdot)$ is a function that extracts the sign of an integer number. The second equality in (4.9) is due to that

$$\begin{aligned} M_{i,k=i} &= D^{+} \cdot \sum_{\{j|h_{ij}=1\}} h_{k=i,j} + D^{-} \cdot \sum_{\{j|h_{ij}=-1\}} h_{k=i,j} \\ &= \frac{q}{2} D^{+} - \frac{q}{2} D^{-} = \frac{q}{2} D. \end{aligned} \tag{4.10}$$

The third equality in (4.9) is set up by (2.11). This proves the lemma. □

The following theorem gives a $q\delta/2$-decodable signature code from matrix X_q.

Theorem 3 For any two non-zero vectors $\boldsymbol{u}_a \in \{-1, 0, 1\}^{T_a}$ and $\boldsymbol{u}_d \in \{-1, 0, 1\}^{T_d}$, if matrices A and D satisfy that

$$w(\boldsymbol{u}_a A) \geq \delta_a \tag{4.11}$$

$$w(\boldsymbol{u}_d D) \geq \delta_d \tag{4.12}$$

set $\mathcal{S}_q$ that is a collection of rows of matrix X_q of (4.3) is a

$$(qn, q\delta/2, T_a + (q-1)T_d)_k\text{-signature code,}$$

where $\delta = \min\{2\delta_a, \delta_d\}$. □

Proof : Let $T = T_a + (q-1)T_d$, we prove that set $\mathcal{S}_q$ is $q\delta/2$-decodable, i.e.

$$w(\boldsymbol{u}X_q) \geq q\delta/2 \tag{4.13}$$

for $\boldsymbol{u} \in \{-1,0,1\}^T$ and $\boldsymbol{u} \neq \boldsymbol{0}^T$.

Let $\boldsymbol{u}X_q = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \boldsymbol{\phi}_q]$, where $\boldsymbol{\phi}_i$ is an n-vector. From (4.5), we have that

$$\begin{aligned}\boldsymbol{u}X_q \boxtimes H_q^{\mathtt{T}} &= [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \boldsymbol{\phi}_q] \boxtimes H_q^{\mathtt{T}} \\ &= [\sum_{j=1}^{q} h_{1j}\boldsymbol{\phi}_j, \ldots, \sum_{j=1}^{q} h_{ij}\boldsymbol{\phi}_j, \ldots, \sum_{j=1}^{q} h_{qj}\boldsymbol{\phi}_j].\end{aligned} \tag{4.14}$$

Let $\boldsymbol{u} = [\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_q]$, where $\boldsymbol{u}_1 \in \{-1,0,1\}^{T_a}$ and $\boldsymbol{u}_r \in \{-1,0,1\}^{T_d}$, $r = 2, \ldots, q$. Multiplying (4.6) by $\boldsymbol{u}$, and comparing with (4.14), we have that

$$\begin{aligned} & w(\sum_{j=1}^{q} h_{ij}\boldsymbol{\phi}_j) \\ = & \begin{cases} \frac{q}{2} w(2\boldsymbol{u}_1 A + (\sum\limits_{r=2}^{q} \boldsymbol{u}_r)(D^+ + D^-)), & i = 1 \\ \frac{q}{2} w(\boldsymbol{u}_i D), & i > 1. \end{cases}\end{aligned} \tag{4.15}$$

Since $\boldsymbol{u} \neq \boldsymbol{0}^T$ by assumption, there exists at least one i_0, so that $\boldsymbol{u}_{i_0=1} \neq \boldsymbol{0}^{T_a}$ or $\boldsymbol{u}_{i_0} \neq \boldsymbol{0}^{T_d}$ for $i_0 \in \{2, \ldots, q\}$.

Case 1. $\boldsymbol{u}_{i_0=1} \neq \mathbf{0}^{T_a}, [\boldsymbol{u}_2, \ldots, \boldsymbol{u}_q] = \mathbf{0}^{T-T_a}$. In this case, from (4.11), we have that

$$w(\sum_{j=1}^{q} h_{1j}\boldsymbol{\phi}_j) = \frac{q}{2} w(2\boldsymbol{u}_1 A) \geq \frac{q}{2} 2\delta_a \geq \frac{q}{2}\delta \tag{4.16}$$

Case 2. $\boldsymbol{u}_{i_0} \neq \mathbf{0}^{T_d}$, $1 < i_0 \leq q$. In this case, from (4.12), we have that

$$w(\sum_{j=1}^{q} h_{i_0 j}\boldsymbol{\phi}_j) = \frac{q}{2} w(\boldsymbol{u}_{i_0} D) \geq \frac{q}{2}\delta_d \geq \frac{q}{2}\delta, \quad 1 < i_0 \leq q. \tag{4.17}$$

Since all the elements of H_q are either -1 or 1, it follows that

$$\begin{aligned} w(\boldsymbol{u}X_q) &= \sum_{j=1}^{q} w(\boldsymbol{\phi}_j) = \sum_{j=1}^{q} w(h_{i_0 j}\boldsymbol{\phi}_j) \\ &\geq w(\sum_{j=1}^{q} h_{i_0 j}\boldsymbol{\phi}_j) \geq \frac{q}{2}\delta. \end{aligned} \tag{4.18}$$

This proves the theorem. □

Remark 1 The coding scheme in Theorem 3 is a generalization of the recursive coding scheme in Chapter 3 where the recursive matrix can be rewritten as matrix $H_q \bar{\otimes} [A|D]$ with the first row and first column removed, where

$$A = D^+ = \begin{bmatrix} 0 & 0 \\ (\mathbf{0}^{\ell+1})^{\mathsf{T}} & \boldsymbol{a}^{\mathsf{T}} \end{bmatrix} \text{ and } D^- = \begin{bmatrix} k & k \\ (\boldsymbol{k}^{\ell+1})^{\mathsf{T}} & (\boldsymbol{k}^{\ell+1})^{\mathsf{T}} - \boldsymbol{a}^{\mathsf{T}} \end{bmatrix}. \tag{4.19}$$

From Theorem 1, the sum rate of the code in Chapter 3 is

$$\frac{2^{j-1}\ell + 2^j - 1}{2^j - 1} = 1 + \frac{2^{j-1}\ell}{2^j - 1} \leq (1 + \ell)$$

where $(1+\ell)$ is the sum rate of the original signature matrix $\boldsymbol{a}^{\mathsf{T}}$. It is obviously that the sum rate is decreased with an increase of code length $(2^j - 1)$.

Let $A' = (D')^+ = D^-$ and $(D')^- = A$, the matrix $H_q\bar{\otimes}[A'|D']$ can also give an error-correcting signature code. Since $T_{a'} = T_{d'}$, the sum rate of the code retains with an increase of the code length.

In the next remark, we demonstrate that our coding scheme in Theorem 3 can obtain codes whose sum rate are increased with the increase of the code length.

Remark 2 To illustrate the point of our work, let us examine further the coding scheme of (4.3). Let H_q be a Sylvester-type Hadamard matrix. Then we have $H_q = H_{q/2}\otimes H_2$. With q be replaced with $2q$, (4.3) becomes

$$\begin{aligned} X_{2q} &= H_{2q}\bar{\otimes}[A|D] \\ &= (H_q \otimes H_2)\bar{\otimes}[A|D] \\ &= H_q\bar{\otimes}[X_2|D_2] \end{aligned} \tag{4.20}$$

where $X_2 = H_2\bar{\otimes}[A|D]$ is a $(2n, \delta, T_a + T_d)_k$-signature matrix, and $D_2 = H_2 \otimes D$. It is shown in (4.20) that, when a $(2n, \delta, T_a + T_d)_k$-signature code $\mathcal{S}_2$ is given *a priori*, a $(q \cdot 2n, q \cdot \delta, T_a + (2q-1)T_d)_k$-signature code $\mathcal{S}_{2q}$ is generated by Hadamard matrix H_q. The sum rate of $\mathcal{S}_{2q}$ is

$$\begin{aligned} R(\mathcal{S}_{2q}) &= \frac{T_a + (2q-1)T_d}{q(2n)} \\ &= \frac{T_a + T_d}{2n} + \frac{(1-1/q)(T_d - T_a)}{2n} \\ &= R(\mathcal{S}_2) + \frac{(1-1/q)(T_d - T_a)}{2n}. \end{aligned} \tag{4.21}$$

Turning now to (4.20), a simple implementation of X_{2q} may be given by $H_{2q}\bar{\otimes}[A|A]$.

In this special case, signature matrix $H_{2q}\bar{\otimes}[A|A]$ is formed by replacing elements $+1$ with A and -1 with $O_a(D^-)$ in Hadamard matrix (see (4.4)) without any distinction between element $+1$'s in the first row and the remaining rows. For example of $k = 1$, signature matrix $H_{2q}\bar{\otimes}[A|A]$ reduces to the binary error-correcting signature matrices in [25] [26]. This results in that the code parameters of $(q2n, q\delta, q2T_a)_k$-signature matrix $H_{2q}\bar{\otimes}[A|A]$ is exactly q-times of $(2n, \delta, 2T_a)_k$-signature matrix $H_2\bar{\otimes}[A|A]$ without any increase of sum rate. The same phenomenon can be observed in the conventional error-correcting coding for MAAC [6] [11] [12].

The coding scheme in this paper seems similar to the implementation above and the conventional works [6] [11] [12], in the sense that code length and error-correction capability is improved by a multiple of q. However, we here try to support *as many users as possible* by employing the difference matrix $D = D^+ - D^-$, instead of A, in the rows except the first row. As a result, if $T_d > T_a$, it is possible to construct code $\mathcal{S}_{2q}$ whose sum rate of (4.21) is increased with an increase in the value of q. This will be discussed in the following section.

Although δ_a, δ_d in Theorem 3 are any positive integer, hereafter we consider only the case of $\delta_a = 1$, $\delta_d = 1$, i.e. $\delta = \min\{2\delta_a, \delta_d\} = 1$. □

4.2 $(k + 1)$-Ary Signature Codes

In this section, we give a UD ($\delta_a = 1$) signature matrix A and a UD ($\delta_d = 1$) difference matrix D. From Theorem 3, two $(k + 1)$-ary $q/2$-decodable signature codes are obtained.

4.2.1 Signature Code with Code Length $2^j q$

For any integer k, let $\ell = \lfloor \log_2 k \rfloor$, and $1 \times (\ell+1)$ matrix $\boldsymbol{a} = [2^0, 2^1, \cdots, 2^{\ell-1}, k]$. We note that matrix $\boldsymbol{a}^{\mathrm{T}}$ is UD signature matrix [19], i.e. $w(\boldsymbol{u}\boldsymbol{a}^{\mathrm{T}}) \geq 1$ with $\boldsymbol{u} \neq \mathbf{0}^{\ell+1}, \boldsymbol{u} \in \{-1,0,1\}^{\ell+1}$. From $\boldsymbol{a}^{\mathrm{T}}$, UD difference matrix and UD signature matrix are given following.

Lemma 3 Let $D^{(0)} = \boldsymbol{a}^{\mathrm{T}}$, for any non-negtive integer j, matrix

$$D^{(j)} = \begin{bmatrix} D^{(j-1)} & D^{(j-1)} \\ D^{(j-1)} & -D^{(j-1)} \\ I^{(j-1)} & O^{(j-1)} \end{bmatrix} \tag{4.22}$$

is a $\langle 2^j, \delta_d^j = 1, T_d^j = j2^{j-1} + (\ell+1)2^j \rangle_k$-difference matrix, where $I^{(j)}$ is $2^j \times 2^j$ identity matrix and $O^{(j)}$ is $2^j \times 2^j$ all-zero matrix. □

The proof of Lemma 3 is omitted since it resembles the proof in [6]. Note that when $k = 1$, (4.22) reduces to the difference matrix constructed by Cantor [15] and Chang [6].

Construction I Let $X_2^{(0)} = \boldsymbol{a}^{\mathrm{T}}$. From UD difference matrix $D^{(j)}$ of (4.22), by Theorem 3, set $\mathcal{S}_q^{(j)}$ that is a collection of rows of matrix

$$X_q^{(j)} = H_q \bar{\otimes} [X_2^{(j-1)} | D^{(j-1)}], \quad j \geq 1 \tag{4.23}$$

is a $(2^{j-1}q, q/2, q(j-1)2^{j-2} + q(\ell+1)2^{j-1} - 2^{j-1} + 1)_k$- signature code. □

Note that set $\mathcal{S}_2^{(j)}$ that is a collection of rows of matrix

$$X_{q=2}^{(j)} = H_2 \bar{\otimes} [X_2^{(j-1)} | D^{(j-1)}]$$

Table 4.1: Error-correcting signature codes ($k = 2$) by Construction I (Subscript k omitted).

j	$D^{(j-1)}$	$\mathcal{S}_2^{(j-1)}$	$\mathcal{S}_q^j$				
	$\langle 2^{j-1}, 1,$	$(2^{j-1}, 1,$	$(2^{j-1}q, q/2, q(j-1)2^{j-2} - 2^{j-1} + q2^j + 1)$				
	$(j-1)2^{j-2} + 2^j\rangle$	$(j-3)2^{j-2} + 2^j + 1)$	$q = 2$	$q = 4$	$q = 8$	$q = 16$	$q = 32$
1	$\langle 1, 1, 2\rangle$	(1,1,2)	(2,1,4)	(4,2,8)	(8,4,16)	(16,8,32)	(32,16,64)
2	$\langle 2, 1, 5\rangle$	(2,1,4)	(4,1,9)	(8,2,19)	(16,4,39)	(32,8,79)	(64,16,159)
3	$\langle 4, 1, 12\rangle$	(4,1,9)	(8,1,21)	(16,2,45)	(32,4,93)	(64,8,189)	(128,16,381)
4	$\langle 8, 1, 28\rangle$	(8,1,21)	(16,1,49)	(32,2,105)	(64, 4, 217)	(128,8,441)	(256,16,889)

is a $(2^{j-1}, 1, T_a^j = (j-1)2^{j-1} + (\ell+1)2^j - 2^{j-1} + 1)_k$-signature matrix. We have that

$$T_d^j - T_a^j = 2^j - 1.$$

Table 1 gives signature codes with $k = 2$ and $j \leq 4$ by Construction I. We give an example of $(8, 4, 16)_2$-signature code in Table 1 in detail.

Example 4.1 Let $k = 2$, $\ell = 1$, and $\boldsymbol{a} = [1, 2]$. From (4.22) and (4.23), we have

$$D^{(0)} = X_2^{(0)} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}. \tag{4.24}$$

From Construction I, the set $\mathcal{S}_8^{(1)}$ that is a collection of rows of matrix $X_8^{(1)} = H_8 \bar{\otimes} [X_2^{(0)} | D^{(0)}]$ (see in pp. 47) is a $(8, 4, 16)_2$-signature code. □

4.2.2 Signature Code with Code Length $2^j q - 1$

In this section, we give a $q/2$-decodable $(k+1)$-ary signature code with code length $(2^j q - 1)$.

Construction II Let $\tilde{X}_2^{(1)} = [(\mathbf{0}^{(\ell+1)})^{\mathrm{T}}, \boldsymbol{a}^{\mathrm{T}}]$. Set $\tilde{\mathcal{S}}_q^{(j)}$ is a collection of rows of a matrix that formed by removing the first all-zero column of

$$\tilde{X}_q^{(j)} = H_q \bar{\otimes} [\tilde{X}_2^{(j-1)} | (-D^{(j-1)})], \quad j \geq 2. \tag{4.25}$$

By Theorem 3, $\tilde{\mathcal{S}}_q^{(j)}$ is a

$$(2^{j-1}q - 1, q/2, q(j-1)2^{j-2} + q(\ell+1)2^{j-1} - 2^{j-1} - \ell)_k\text{-signature code.} \qquad \square$$

$$X_8^{(1)} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 2 & 0 & 2 & 0 & 2 & 0 & 2 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 2 & 2 & 0 & 0 & 2 & 2 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 2 & 0 & 0 & 2 & 2 & 0 & 0 & 2 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 2 & 0 & 2 & 0 & 0 & 2 & 0 & 2 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 2 & 2 & 0 & 0 & 0 & 0 & 2 & 2 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 2 & 0 & 0 & 2 & 0 & 2 & 2 & 0 \end{bmatrix}$$

Note that when $k = 1$, $\tilde{\mathcal{S}}_{q=2}^{(j)}$ reduces to the UD binary signature code in [15], and when $k \geq 1$, $\tilde{\mathcal{S}}_{q=2}^{(j)}$ reduces to the UD signature code in [19].

4.3 Binary Signature Codes

From Construction I and II, when $k = 1$, binary signature codes are provided with code length $2^j q$ and $2^j q - 1$. In this section, we give the construction of binary signature code with code length extending to qn or $qn - 1$, where n is a positive integer.

Before the construction, we recall the UD signature code [17] and UD difference matrix [7] with arbitrary length n.

An arbitrary positive integer n is represented as the binary form

$$n = \sum_{j=0}^{r} n_j 2^j \tag{4.26}$$

where $r = \lfloor \log_2 n \rfloor$, and $n_j \in \{0, 1\}$, $j = 0, 1, \ldots, r$.

From [17], for arbitrary code length n, $(n, \delta_a = 1, T_a(n))_1$-signature matrix A^n has the number of users

$$T_a(n) = \sum_{j=1}^{r} n_j [j 2^{j-1} + \sum_{i=0}^{j-1} n_i 2^i + 1] + n_0. \tag{4.27}$$

From [7], for arbitrary code length n, $\langle n, \delta_d = 1, T_d(n) \rangle_1$-difference matrix D^n over $\{-1, 0, 1\}$ has the number of users

$$T_d(n) = r 2^r + n - \sum_{j=0}^{r-1} |n_j - 1| (j + 2) 2^{j-1}. \tag{4.28}$$

4.3.1 Binary Signature Code with Code Length qn

Construction III For any positive integer n, from $T_a(n) \times n$ UD binary signature matrix A^n in [17], and $T_d(n) \times n$ UD difference matrix D^n in [7], by Theorem 3, set of $\mathcal{S}_q^n$ that is a collection of rows of matrix

$$X_q^n = H_q \bar{\otimes} [A^n | D^n]$$

is a $(qn, q/2, T_a(n) + (q-1)T_d(n))_1$-signature code. □

Note that when $n = 2^j$, $\mathcal{S}_q^n$ is the signature code $\mathcal{S}_q^{(j)}$ in Construction I with $k = 1$.

4.3.2 Binary Signature Code with Code Length $qn - 1$

Construction IV For any positive integer n, let $\tilde{A}^n = [(\mathbf{0}^{T_a(n-1)})^{\intercal}, A^{n-1}]$. Set $\tilde{\mathcal{S}}_q^n$ is a collection of rows of a matrix that formed by removing the first all-zero column of

$$\tilde{X}_q^{(j)} = H_q \bar{\otimes} [\tilde{X}_2^{(j-1)} | (-D^{(j-1)})], \quad j \geq 2. \tag{4.29}$$

By Theorem 3, set $\tilde{\mathcal{S}}_q^n$ is a $(qn - 1, q/2, T_a(n-1) + (q-1)T_d(n))_1$-signature code. □

Note that when $n = 2^j - 1$, $\tilde{\mathcal{S}}_q^n$ is signature code $\tilde{S}_q^{(j)}$ in Construction II with $k = 1$.

4.4 Decoding Rule

The construction in Theorem 3 reduces the decoding problem for error-correcting codes to the δ_a-decodable signature matrix and δ_d-decodable difference matrix. We now give the decoding rule of the $(k+1)$-ary error-correcting signature code.

Consider a T-user MAAC system with signature code $\mathcal{S}_q$ that is associated with signature matrix X_q. Assume that the channel is disturbed by the additive noise, and error vector $\boldsymbol{e}$ has weight $w(\boldsymbol{e}) \leq \lfloor (q/2-1)/2 \rfloor$. From (2.1), the decoder receives vector

$$\boldsymbol{y} = \boldsymbol{b}X_q + \boldsymbol{e}. \tag{4.30}$$

The decoding of the $(k+1)$-ary signature code includes two steps: error correction and user identification.

(*Error correction*) After block multiplying $\boldsymbol{y}$ by matrix $H_q^{\mathtt{T}}$, by Lemma 2 we have

$$\begin{aligned}
\boldsymbol{y} \boxtimes H_q^{\mathtt{T}} &= \boldsymbol{b}X_q \boxtimes H_q^{\mathtt{T}} + \boldsymbol{e} \boxtimes H_q^{\mathtt{T}} \\
&= \frac{q\boldsymbol{b}}{2} \begin{bmatrix} 2A & \mathrm{O}_a & \ldots & \mathrm{O}_a \\ (D^+ + D^-) & D & \vdots & \mathrm{O}_d \\ \vdots & \vdots & \ddots & \vdots \\ (D^+ + D^-) & \mathrm{O}_d & \vdots & D \end{bmatrix} \\
&\quad + \boldsymbol{e} \boxtimes H_q^{\mathtt{T}}.
\end{aligned} \tag{4.31}$$

If $\boldsymbol{e}$ is a zero vector (noiseless, i.e., $w(\boldsymbol{e}) = 0$), then the elements of $\boldsymbol{y} \boxtimes H_q^{\mathtt{T}}$ are multiples of $q/2$, including zero.

If $\boldsymbol{e}$ has a weight, i.e., $w(\boldsymbol{e}) \neq 0$, then some elements of $\boldsymbol{y} \boxtimes H_q^{\mathtt{T}}$ are not equal to the multiple of $q/2$ due to noise $\boldsymbol{e}$. Note that in (4.31), the weight of each element of $\boldsymbol{e} \boxtimes H_q^{\mathtt{T}}$ is not greater than $\lfloor (q/2-1)/2 \rfloor$, since $w(\boldsymbol{e}) \leq \lfloor (q/2-1)/2 \rfloor$ by assumption and the element of $H_q^{\mathtt{T}}$ is either 1 or -1. Thus, we correct these errors in $\boldsymbol{y} \boxtimes H_q^{\mathtt{T}}$ simply by *replacing* these elements with the nearest multiple of $q/2$. Specifically, let ζ_t be the t-th component in vector $\boldsymbol{y} \boxtimes H_q^{\mathtt{T}}$, and

$$r_t \triangleq \zeta_t \text{ modulo } q/2, \quad 0 \leq r_t < q/2 \tag{4.32}$$

where residue r_t is non-negative. Then error correction is carried out as follows:

$$\begin{array}{lcll} \zeta_t & \leftarrow & \zeta_t, & \text{if } r_t = 0 \\ \zeta_t & \leftarrow & \zeta_t - r_t, & \text{if } 0 < r_t \leq \lfloor \frac{q/2-1}{2} \rfloor \\ \zeta_t & \leftarrow & \zeta_t + 2^{j-1} - r_t, & \text{if } \lfloor \frac{q/2-1}{2} \rfloor < r_t < q/2. \end{array} \tag{4.33}$$

(*User identification*) The status of the users is detected after the error correction of (4.33). The corrected version of $\boldsymbol{y} \boxtimes H_q^{\mathtt{T}}$ is now reasonably represented by the same notation as $\boldsymbol{y} \boxtimes H_q^{\mathtt{T}}$:

$$\begin{aligned} \frac{2}{q}[\zeta_1, \zeta_2, \ldots, \zeta_{qn}] &\triangleq \frac{2}{q}\boldsymbol{y} \boxtimes H_q^{\mathtt{T}} \\ &= \boldsymbol{b} \begin{bmatrix} 2A & \mathrm{O}_a & \ldots & \mathrm{O}_a \\ (D^+ + D^-) & D & \vdots & \mathrm{O}_d \\ \vdots & \vdots & \ddots & \vdots \\ (D^+ + D^-) & \mathrm{O}_d & \vdots & D \end{bmatrix}. \end{aligned} \tag{4.34}$$

Since A is δ_a-decodable signature matrix and D is δ_d-decodable difference matrix, decoding table can be obtained. Then, we have the following detected status of users:

a) Binary vector $[b_{T_a+iT_d+1},\ldots,b_{T_a+(i+1)T_d}]$ is decoded from the value of $\frac{2}{q}[\zeta_{(i+1)n},\ldots,\zeta_{(i+2)n}]$, $i=0,1,\ldots,(q-2)$, by the decoding table of D.

b) Let

$$\begin{aligned}\boldsymbol{y}_a= &\frac{1}{q}[\zeta_1,\ldots,\zeta_n]\\ &-\frac{1}{2}\sum_{i=0}^{q-2}[b_{T_a+iT_d+1},\ldots,b_{T_a+(i+1)T_d}](D^{+}+D^{-})),\end{aligned} \tag{4.35}$$

binary vector $[b_1,\ldots,b_{T_a}]$ is decoded from $\boldsymbol{y}_a$ by the decoding table of A.

By the above decoding rule, the decoder can correct the transmission errors caused by the channel noise and uniquely resolve the received superimposed vector into the transmitted codewords that is the status of the corresponding users.

Note that the decoding operation only requires the block multiplication of received signal $\boldsymbol{y}$ by binary matrix H_q^{T} (see (4.31)), the modulo operation of the components of $\boldsymbol{y}\boxtimes H_q^{\mathrm{T}}$ (see (4.32)), and the mapping operation based on the decoding table for user detection (see a), b) above).

Example 4.2 In this example, we apply the decoding rule to $(8,4,16)_2$-signature code $\mathcal{S}_8^1$ (Example 4.1) as follows:

Assume that the status of users is given by

$$[b_1,b_2,\ldots,b_{11}]=[0,1,1,0,0,1,1,1,0,0,1,0,1,1,0,0]$$

where users 2, 3, 6, 7, 8, 11, 13, and 14 are active. When the channel is disturbed by error vector $\boldsymbol{e} = [1, 0, 0, 0, 0, 0, 0, 0]$, the received vector is

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{s}_2 + \boldsymbol{s}_3 + \boldsymbol{s}_6 + \boldsymbol{s}_7 + \boldsymbol{s}_8 + \boldsymbol{s}_{11} + \boldsymbol{s}_{13} + \boldsymbol{s}_{14} + \boldsymbol{e} \\ &= [13, 7, 4, 5, 8, 5, 6, 9]. \end{aligned}$$

(*Error correction*) By block multiplying vector $\boldsymbol{y}$ with matrix $H_8^{\mathtt{T}}$, we get

$$\begin{aligned} \boldsymbol{y} \boxtimes H_8^{\mathtt{T}} &= [13, 7, 4, 5, 8, 5, 6, 9] H_8^{\mathtt{T}} \\ &= [57, 5, 9, 13, 1, 5, 13, 1]. \qquad (4.36) \end{aligned}$$

By changing the elements in (4.36) into the nearest multiple of 4, we have the corrected version of $[56, 4, 8, 12, 0, 4, 12, 0]$.

(*User identification*) Moreover, when divided by 4, the above vector becomes $[14, 1, 2, 3, 0, 1, 3, 0]$. The elements $[b_3, \ldots, b_{16}]$ are mapped by a decoding table, since $D_0 = \boldsymbol{a}^{\mathtt{T}} = [1, 2]^{\mathtt{T}}$ is uniquely decodable. The table is $0 \rightarrow 00, 1 \rightarrow 10, 2 \rightarrow 01, 3 \rightarrow 11$. Therefore, we have

$$\begin{aligned} [b_3, b_4] &= [1, 0], \\ [b_5, b_6] &= [0, 1], \\ [b_7, b_8] &= [1, 1], \\ [b_9, b_{10}] &= [0, 0], \\ [b_{11}, b_{12}] &= [1, 0], \\ [b_{13}, b_{14}] &= [1, 1], \\ [b_{15}, b_{16}] &= [0, 0]. \end{aligned}$$

Since the elements in D_0 are all positive, it follows that $D_0^- = \mathbf{0}$ and $D^+ + D^- = D$. Thus, from (4.35), we have that

$$\begin{aligned} \boldsymbol{y}_a &= \frac{1}{8}\zeta_1 - \frac{1}{2}\sum_{i=0}^{6}[b_{3+2i}, b_{4+2i}](D^+ + D^-) \\ &= \frac{56}{8}\zeta_1 - \frac{10}{2} = 2. \end{aligned}$$

Since $X_2^0 = D_0$, we decode $[b_1, b_2]$ the decoding table of D_0 and have

$$[b_1, b_2] = [0, 1].$$

Thus, decoding with error correction and user identification has been successfully completed. □

4.5 Conclusion

In this chapter, we proposed a coding scheme of $(k+1)$-ary error-correcting signature codes for noisy MAAC. Given a signature matrix A and a difference matrix $D = D^+ - D^-$ *a priori*, we obtain a larger signature matrix by replacing each element in Hadamard matrix with A, or D^+, or D^- depending on the values of elements and their locations in Hadamard matrix. The set of rows of proposed matrix gives an error-correcting signature code. Introducing the difference matrix makes it possible to construct error-correcting signature code whose sum rate is increased with an increase in the order of Hadamard matrix. We gave either binary or non-binary signature codes with higher sum rates than previous codes.

Chapter 5

Two-User Turbo Decoding with Simplified Sum Trellis for Gaussian TWRC

5.1 Introduction

In the previous chapters, we discuss signature codes for MAAC, in the later chapters, we consider the decoding problems for TWRC.

Relay networks, as an efficient strategy to improve cell-edge user performance, have attracted significant attention. TWRC as shown in 5.1 is a fundamental network structure of much interest to the wireless communications research community. Physical-layer network

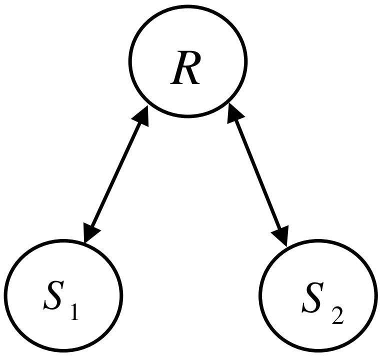


Figure 5.1: Two-way relay channel.

coding (PNC) [28], which is a network coding method [29] applying for a physical layer, exchanges messages between two users with the help of a relay. The message exchange consists of two phases: first, two users simultaneously transmit signals to a relay node; second, the relay detects the superimposed signal, and broadcasts an XORed message of two users' messages. Each user can decode the opposite user's message by an XOR operation of the local message and received XORed message.

Theoretical analysis shows that the PNC scheme improves the throughput performance of a wireless network [30], and achieves a higher maximum sum-rate than that of the four-time-slot transmission scheme due to two-time-slot transmission [31–33]. For practical application of PNC at the relay, several transmission protocols, such as amplification forward [34] [35] and denoise-and-forward protocols [36], have been proposed at the relay. The maximum a posteriori estimation and minimum mean square error (MMSE) estimation, etc., are investigated to estimate the XORed message [37]. In the TWRC, modulation with a 5-ary constellation is also investigated [38]. In the above schemes, however, channel coding is not taken into account.

In combination with channel coding, PNC can achieve more reliable communication. A joint channel decoding network coding scheme is presented in [39]. The relay decodes the superimposed signal to the soft information, i.e. *a posteriori probability* (APP), of the arithmetic sum of the two users' messages, and transforms it to the XORed codeword. In [39], when the repeat-accumulate code is employed, a decoder at the relay is provided by extending the belief propagation algorithm for the traditional point-to-point channel to the TWRC. On the other hand, the Viterbi algorithm for the TWRC is investigated when convolutional code is used [40]. The two-user trellis is given by combining the trellises of two convolutional codes of two users, where the state set is a Kronecker product of two state sets of convolutional codes. Furthermore, a reduced-state trellis is given based on a state set formed by an XOR operation of two state sets of convolutional codes [40–42]. However, the decoding with the reduced-state trellis is to directly obtain the XORed message without distinguishing the three values of the arithmetic sum, which degrades the decoding performance and is not suitable for iterative decoding.

In this chapter, we combine binary turbo coding with PNC in Gaussian TWRC. We propose a two-user turbo decoding scheme with a simplified sum trellis. For two-user iterative decoding at the relay, the component decoder with its simplified sum trellis decodes the superimposed signal to the arithmetic sum of two users' messages. The simplified sum trellis is obtained by removing one of the states in a pair of mutual symmetrical states from a sum trellis. This removal reduces the decoding complexity to half of that with the sum trellis, and does not degrade decoding performance over AWGN channel since two output sequences from the pair of mutual symmetrical states are the same.

5.2 System Model

5.2.1 Encoder

Consider a wireless TWRC, where two users S_1 and S_2 communicate with each other through a single relay R (Fig. 5.2). It is assumed that all nodes are half duplex, i.e. a node cannot receive and transmit simultaneously. The channel from user S_1 (S_2) to relay R is additional white Gaussian noise (AWGN) channel.

We employ two of the same binary turbo encoders in TWRC. Let $\boldsymbol{u}_i = (u_{i,1}, ..., u_{i,K})$, $u_{i,k} \in \{0, 1\}$, be the message of user $S_i, i = 1, 2$, and $\boldsymbol{c}_i = (c_{i,1}, ..., c_{i,N}), c_{i,n} \in \{0, 1\}$, be the transmitted codeword, whose parity part is the output of the component encoders (recursive systematic convolutional (RSC) encoders: RSCs $\mathcal{P}$ and $\mathcal{Q}$). For transmission, we will favor $c_{i,k} \in \{+1, -1\}$ over $c_{i,k} \in \{0, 1\}$ under the mapping $\{0 \leftrightarrow +1, 1 \leftrightarrow -1\}$.

Consider a two-phase transmission protocol: at the first phase, two users S_1 and S_2 transmit $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ simultaneously to the relay R (Fig. 5.2). Thus, the received signal at relay R has a superposition form

$$\boldsymbol{y}_R = \boldsymbol{c}_1 + \boldsymbol{c}_2 + \boldsymbol{z} \tag{5.1}$$

where $\boldsymbol{z} = (z_1, ..., z_N)$ is noise whose element z_n is a zero-mean Gaussian noise with variance

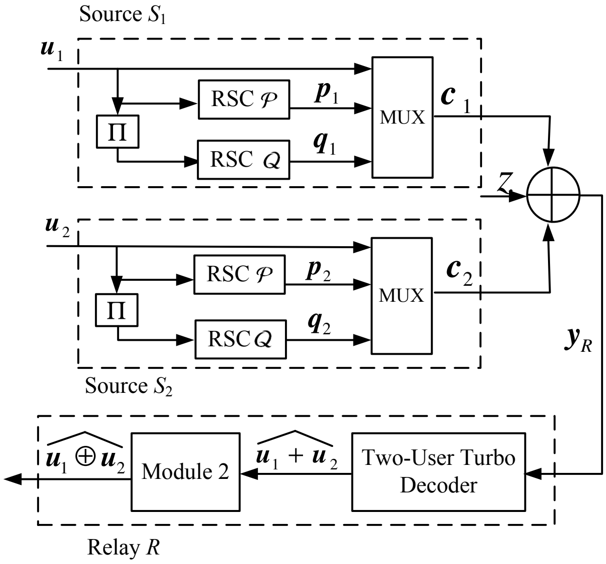


Figure 5.2: System model of turbo codes in two-way relay channel.

σ^2.

At the relay, a two-user turbo decoding estimates APPs $\mathrm{P}((u_{1,k} + u_{2,k}) = v \mid \boldsymbol{y}_R)$ of the arithmetic sum of two users' messages, where $v = 0, 1, 2$, and gives the estimation of $(\widehat{\boldsymbol{u}_1 + \boldsymbol{u}_2})$. The network-coded information bit, i.e. the estimation of XOR of $u_{1,k}$ and $u_{2,k}$, is

$$(\widehat{u_{1,k} \oplus u_{2,k}}) = \begin{cases} 0, & \text{if } \mathrm{P}((u_{1,k} + u_{2,k}) = 0|\boldsymbol{y}_R) \\ & \quad +\mathrm{P}((u_{1,k} + u_{2,k}) = 2|\boldsymbol{y}_R) \\ & \quad \geq \mathrm{P}((u_{1,k} + u_{2,k}) = 1|\boldsymbol{y}_R) \\ 1, & \qquad\qquad \text{otherwise.} \end{cases} \tag{5.2}$$

At the second phase, the relay broadcasts $(\widehat{\boldsymbol{u}_1 \oplus \boldsymbol{u}_2})$ to the two users. Each user can decode the opposite user's message by XOR operation of the local message and received network-coded message. Here, we focus on the decoding at the relay, and ignore the second

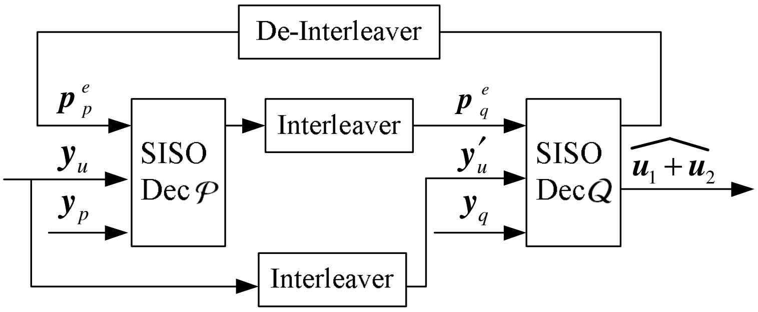


Figure 5.3: Iterative decoder in two-way relay channel.

phase.

5.2.2 Iterative Decoding

To estimate APPs $\mathrm{P}((u_{1,k}+u_{2,k}) \mid \boldsymbol{y}_R)$, the relay performs an iterative decoding based on two component decoders, typically BCJR decoders. As shown in Fig. 5.3, the iterative decoder has the same structure as a traditional iterative decoder except that the information carried out in this iterative decoder is non-binary, i.e. three possible values: $0, 1, 2$. Accordingly, component decoder $\mathcal{P}$ estimates APPs $\mathrm{P}((u_{1,k}+u_{2,k})|\boldsymbol{y}_u, \boldsymbol{y}_p, P_{p,k})$, and component decoder $\mathcal{Q}$ estimates APPs $\mathrm{P}((u_{1,k}+u_{2,k})|\boldsymbol{y}_u', \boldsymbol{y}_q, P_{q,k})$. Here y_u is the received superimposed systematic vector, $\boldsymbol{y}_u'$ is the interleaved version of the superimposed systematic vector, and $\boldsymbol{y}_p$ ($\boldsymbol{y}_q$) is the received superimposed vector of the two parity-vectors from RSCs $\mathcal{P}$ ($\mathcal{Q}$) of users 1 and 2. $P_{p,k} = \mathrm{P}((u_{1,k}+u_{2,k}) = v)$, $v = 0, 1, 2$, is the priori information of decoder $\mathcal{P}$, which is the extrinsic information received from decoder $\mathcal{Q}$.

By BCJR algorithm with trellis, we have [43]

$$\mathrm{P}((u_{1,k}+u_{2,k})=v|\boldsymbol{y}_u,\boldsymbol{y}_p,P_{p,k}) = \sum_{(u_{1,k}+u_{2,k})=v} p((\boldsymbol{a},\boldsymbol{b})\rightarrow(\tilde{\boldsymbol{a}},\tilde{\boldsymbol{b}}),\boldsymbol{y}_u,\boldsymbol{y}_p) \qquad (5.3)$$

where $(\boldsymbol{a},\boldsymbol{b})$ and $(\tilde{\boldsymbol{a}},\tilde{\boldsymbol{b}})$ are states in the trellis at times k and $k+1$, respectively (see below). Function $p((\boldsymbol{a},\boldsymbol{b})\rightarrow(\tilde{\boldsymbol{a}},\tilde{\boldsymbol{b}}),(\boldsymbol{y}_u,\boldsymbol{y}_p))$ is the probability density function over the edge $e((\boldsymbol{a},\boldsymbol{b}),(\tilde{\boldsymbol{a}},\tilde{\boldsymbol{b}}))$ in the trellis from $(\boldsymbol{a},\boldsymbol{b})$ to $(\tilde{\boldsymbol{a}},\tilde{\boldsymbol{b}})$. The trellis used in the iterative decoding will be given in the next section.

5.3 Simplified Sum Trellis for Gaussian TWRC

In this section, we propose the construction of simplified sum trellis at the relay. The simplified sum trellis is obtained by removing one of the states in pair of mutual symmetrical states from a sum trellis.

5.3.1 Sum Trellis

Before proceeding, we describe a sum trellis [40] of component decoder $\mathcal{P}$ in Fig. 5.3 in this section. A similar approach would give the trellis of $\mathcal{Q}$. A simplified version of the trellis, which is obtained from the sum trellis, will be given in Sect. 5.3.2. Note that the sum trellis in [40] is for two-user convolutional decoding with the Viterbi algorithm.

First, we describe trellis T of RSC $\mathcal{P}$. Let the state set of trellis T be $\mathbb{A}=\{\boldsymbol{a}_l|l=1,2,...,2^L\}$, $\boldsymbol{a}_l\in\{0,1\}^L$, where L is the number of the memory cells of encoder $\mathcal{P}$. For

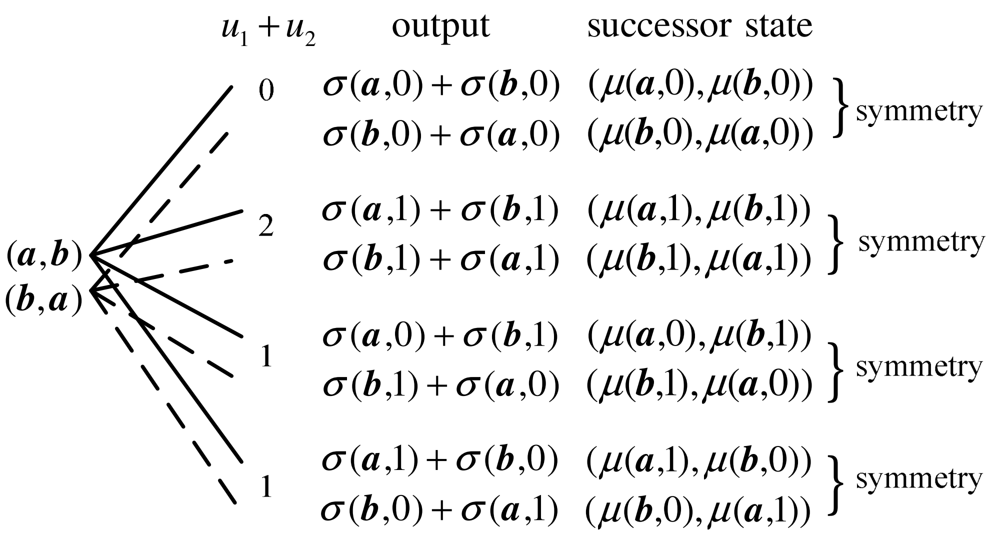

$u_1 + u_2$
output
successor state
0
$\sigma(\boldsymbol{a},0)+\sigma(\boldsymbol{b},0)$
$(\mu(\boldsymbol{a},0),\mu(\boldsymbol{b},0))$
$\sigma(\boldsymbol{b},0)+\sigma(\boldsymbol{a},0)$
$(\mu(\boldsymbol{b},0),\mu(\boldsymbol{a},0))$
symmetry
2
$\sigma(\boldsymbol{a},1)+\sigma(\boldsymbol{b},1)$
$(\mu(\boldsymbol{a},1),\mu(\boldsymbol{b},1))$
$\sigma(\boldsymbol{b},1)+\sigma(\boldsymbol{a},1)$
$(\mu(\boldsymbol{b},1),\mu(\boldsymbol{a},1))$
symmetry
$(\boldsymbol{a},\boldsymbol{b})$
$(\boldsymbol{b},\boldsymbol{a})$
1
$\sigma(\boldsymbol{a},0)+\sigma(\boldsymbol{b},1)$
$(\mu(\boldsymbol{a},0),\mu(\boldsymbol{b},1))$
$\sigma(\boldsymbol{b},1)+\sigma(\boldsymbol{a},0)$
$(\mu(\boldsymbol{b},1),\mu(\boldsymbol{a},0))$
symmetry
1
$\sigma(\boldsymbol{a},1)+\sigma(\boldsymbol{b},0)$
$(\mu(\boldsymbol{a},1),\mu(\boldsymbol{b},0))$
$\sigma(\boldsymbol{b},0)+\sigma(\boldsymbol{a},1)$
$(\mu(\boldsymbol{b},0),\mu(\boldsymbol{a},1))$
symmetry

Figure 5.4: The relation between successor states (outputs) from mutual symmetrical states $(\boldsymbol{a},\boldsymbol{b})$ and $(\boldsymbol{b},\boldsymbol{a})$, $(\boldsymbol{b}\neq\boldsymbol{a})$.

input bit $u\in\{0,1\}$ at state $\boldsymbol{a}\in\mathbb{A}$ at time k, the output is $\tilde{u}=\sigma(\boldsymbol{a},u)$, and the successor state at time $k+1$ is $\tilde{\boldsymbol{a}}=\mu(\boldsymbol{a},u)$, where σ and μ are the edge function and state function, respectively, based on the encoder. Connecting the state $\boldsymbol{a}$ at time k and the state $\tilde{\boldsymbol{a}}$ at time $k+1$, we can obtain an edge $e(\boldsymbol{a},\tilde{\boldsymbol{a}})$ in the trellis.

The trellis, denoted by two-user *sum trellis* T_{sum}, used in decoder $\mathcal{P}$ in Fig. 5.3 is a joint description of RSCs $\mathcal{P}$ of two users, since the input (output) information of component decoder $\mathcal{P}$ correspond to the arithmetic sum of output (input) messages of RSCs $\mathcal{P}$ of two users. Accordingly, the state set of two-user sum trellis T_{sum} is $\mathbb{A}_{sum}=\{(\boldsymbol{a}_l,\boldsymbol{b}_n)|l,n=1,2,...,2^L\}$, where $\boldsymbol{a}_l\in\mathbb{A}$ is from the state set of user 1, and $\boldsymbol{b}_n\in\mathbb{A}$ is from the state set of user 2. For state $(\boldsymbol{a},\ \boldsymbol{b})$ with input $(u_1+u_2)\in\{0,1,2\}$, the output is $(\sigma(\boldsymbol{a},u_1)+\sigma(\boldsymbol{b},u_2))\in\{0,1,2\}$, and the successor state at time $k+1$ is $(\mu(\boldsymbol{a},u_1),\mu(\boldsymbol{b},u_2))$ (Fig. 5.4).

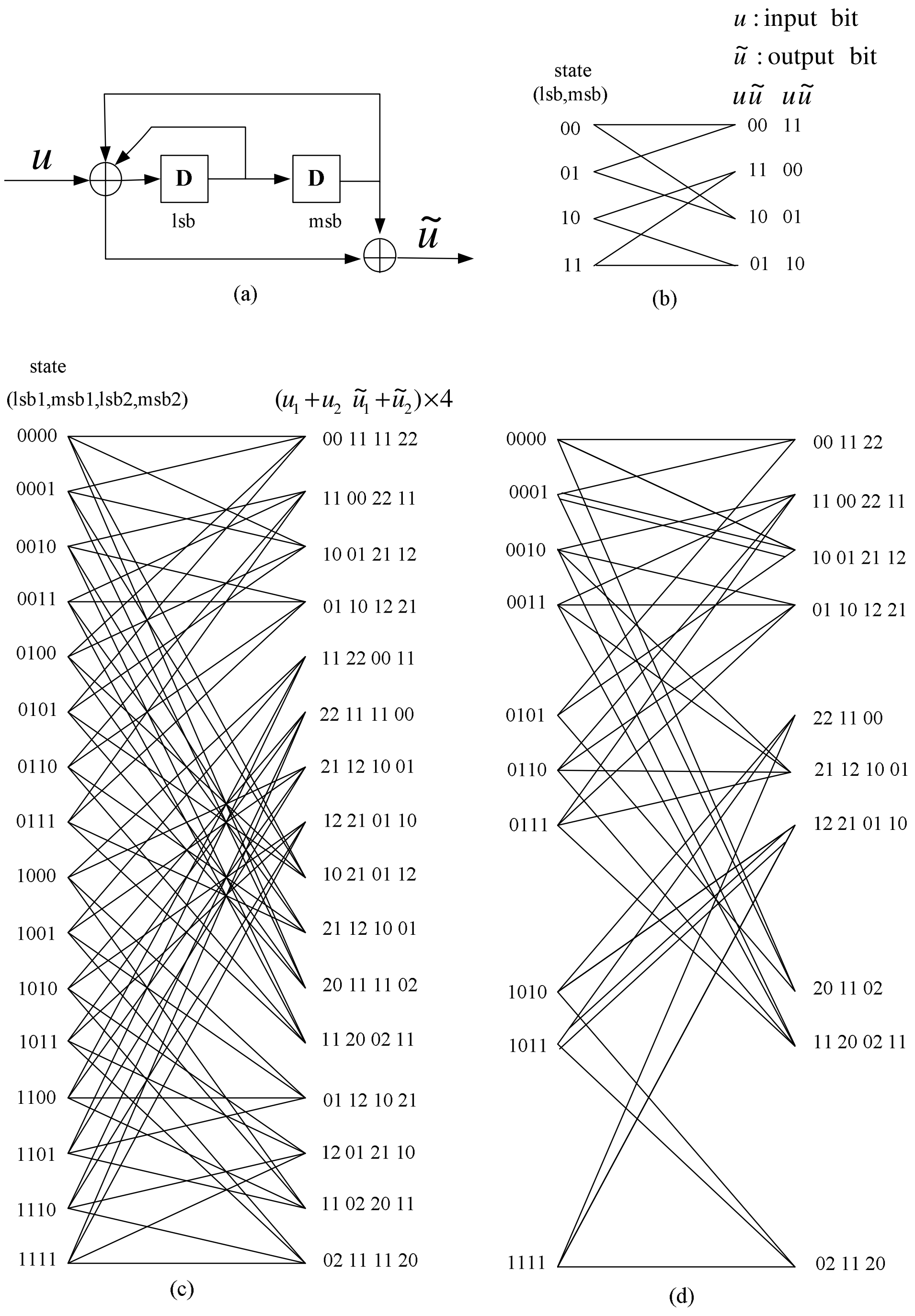


Figure 5.5: An example of forming a simplified sum trellis Γ_{ssm}.

(a). An RSC encoder. (b). The trellis of the RSC encoder.

(c). The sum trellis T_{sum}. (d). The simplified sum trellis Γ_{ssm}.

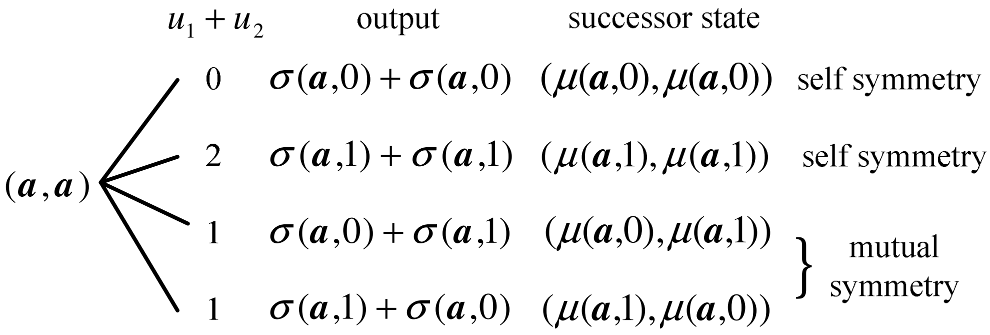

$u_1 + u_2$
output
successor state
$(\boldsymbol{a},\boldsymbol{a})$
0
$\sigma(\boldsymbol{a},0)+\sigma(\boldsymbol{a},0)$
$(\mu(\boldsymbol{a},0),\mu(\boldsymbol{a},0))$
self symmetry
2
$\sigma(\boldsymbol{a},1)+\sigma(\boldsymbol{a},1)$
$(\mu(\boldsymbol{a},1),\mu(\boldsymbol{a},1))$
self symmetry
1
$\sigma(\boldsymbol{a},0)+\sigma(\boldsymbol{a},1)$
$(\mu(\boldsymbol{a},0),\mu(\boldsymbol{a},1))$
1
$\sigma(\boldsymbol{a},1)+\sigma(\boldsymbol{a},0)$
$(\mu(\boldsymbol{a},1),\mu(\boldsymbol{a},0))$
mutual symmetry

Figure 5.6: The relation between successor states (outputs) from self symmetrical states $(\boldsymbol{a}, \boldsymbol{a})$.

Note that, for each state, there are four edges corresponding to four possible inputs because of $u_1, u_2 \in \{0, 1\}$. Throughout all the states and inputs, two-user sum trellis T_{sum} is obtained with the number of M^2 states and the number of $4M^2$ edges, where $M = 2^L$.

We give an example of a sum trellis. An RSC encoder with $L = 2$ memory cells is given in Fig. 5.5(a) with its trellis in Fig. 5.5(b). Consider two states, for instance, 00 and 10, of user 1 and 2, respectively. For the state 00 in user 1, one of two successor states is 00 with input $u_1 = 0$ and output $\tilde{u}_1 = 0$. Similarly, for the state 10 in user 2, one of two successor states is 01 with input $u_2 = 1$ and output $\tilde{u}_2 = 0$. Note that in Fig. 5.5(b), the two pairs of input u and output $\tilde{u}$ of a state are labeled at the right side of the trellis at the same line level of the state. The first and second input-output pairs correspond to the first and second edges originated from the state.

To obtain the sum trellis of Fig. 5.5(c), we combine two state sets (Fig. 5.5(b)) of the two users to produce the state set of the sum trellis. For instance, we combine the state 00 in user 1 and the state 10 in user 2 to produce the state 0010. Their successor states in the sum trellis are also a combination of successor states of users 1 and 2. For instance, one of the four successor states is 0001 with input $u_1 + u_2 = 1$ and output $\tilde{u}_1 + \tilde{u}_2 = 0$ (see the first edge from the state 0010). For the state 0010, the four input-output pairs of 10 01 21 12, corresponding to the first to forth edges, are labeled at the right side at the same line level of state 0010 in the sum trellis.

5.3.2 Simplified Sum Trellis

We are ready to give a simplified sum trellis, which is equivalent to the sum trellis above. Removing partial states from the sum trellis reduces the decoding complexity without any degradation of decoding performance.

In sum trellis T_{sum}, two states $(\boldsymbol{a}, \boldsymbol{b})$ and $(\boldsymbol{a}', \boldsymbol{b}')$, where $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{a}', \boldsymbol{b}' \in \mathbb{A}$, are *symmetrical*, if $\boldsymbol{a} = \boldsymbol{b}'$ and $\boldsymbol{b} = \boldsymbol{a}'$. There exist two types of symmetrical states. We consider the states $(\boldsymbol{a}, \boldsymbol{b})$ and $(\boldsymbol{b}, \boldsymbol{a})$ to be *mutual symmetrical* if $\boldsymbol{a} \neq \boldsymbol{b}$. When $\boldsymbol{a} = \boldsymbol{b}$, the state is called *self symmetrical*. For state $(\boldsymbol{a}, \boldsymbol{b})$, $(\boldsymbol{a} \neq \boldsymbol{b})$, there always exists a mutual symmetrical state $(\boldsymbol{b}, \boldsymbol{a})$ in T_{sum}, since the two turbo encoders are the same for the two users by assumption. This means that the mutual symmetrical states are in-pair in the trellis. Note that the self symmetrical state is itself, and is not in-pair.

In the following, we show that removing one of the states in a pair of mutual symmetrical states from the sum trellis produces an equivalent and simplified trellis. Let's consider only the situation of $\boldsymbol{a} \neq \boldsymbol{b}$. Look at the relation between successor states (or outputs) from mutual symmetrical states $(\boldsymbol{a}, \boldsymbol{b})$ and $(\boldsymbol{b}, \boldsymbol{a})$ (Fig. 5.4). It is obvious from Fig. 5.4 that for $u_1 + u_2 = 0$ or 2, their successor states are (mutual or self-) symmetrical, and their outputs are the same. For $u_1 + u_2 = 1$, successor state $(\mu(\boldsymbol{a}, u_1 = 0), \mu(\boldsymbol{b}, u_2 = 1))$ from state $(\boldsymbol{a}, \boldsymbol{b})$ is symmetrical to successor state $(\mu(\boldsymbol{b}, u_1 = 1), \mu(\boldsymbol{a}, u_2 = 0))$ from state $(\boldsymbol{b}, \boldsymbol{a})$. The output of $(\boldsymbol{a}, \boldsymbol{b})$ is equal to that of $(\boldsymbol{b}, \boldsymbol{a})$, i.e. $\sigma(\boldsymbol{a}, u_1 = 0) + \sigma(\boldsymbol{b}, u_2 = 1) = \sigma(\boldsymbol{b}, u_1 = 1) + \sigma(\boldsymbol{a}, u_2 = 0)$. The remaining two successor states are also symmetrical.

This observation tells us that at time k for every edge from state $(\boldsymbol{a}, \boldsymbol{b})$, there exists an edge that has the same output from its mutual symmetrical state $(\boldsymbol{b}, \boldsymbol{a})$. Moreover, the corresponding successor states at time $k + 1$ are also self or mutual symmetrical. When the successor state is self symmetrical, the two successor states are in fact a single state. When the successor state is mutual symmetrical, as we stated above, the two outputs from the two mutual symmetrical states are also the same. As a result, for input $(k + 1)$-th to $(K + 1)$-th bits in a given message sequence, the output sequence from $(\boldsymbol{a}, \boldsymbol{b})$ at time k is identical to that from $(\boldsymbol{b}, \boldsymbol{a})$ at time k. It is known [44] that two trellises are equivalent if they generate the same code, which is the set of all possible output sequences. This means that we can remove one of the states in a pair of mutual symmetrical states at time k from the sum-trellis T_{sum}. This is also true for all the times, i.e. $k = 1, 2, \ldots, K$.

After removing the partial states discussed above, we turn to discussing the edges. We show that the number of edges originated from self symmetrical state becomes three, and the number of edges originated from mutual symmetrical state is still four. As we stated, any state $(\boldsymbol{a}, \boldsymbol{b})$ at time k in the sum trellis has four edges corresponding to four possible inputs $u_1 + u_2$, $u_1, u_2 \in \{0, 1\}$. Since one of the states in a pair of symmetrical states is removed, the edges originated from the removed states at time k are deleted. When a successor state at time $k+1$ is removed, the corresponding edge from the state $(\boldsymbol{a}, \boldsymbol{b})$ is moved to connect to the mutual symmetrical state of the successor state at time $K + 1$. We discuss the following two cases.

When $\boldsymbol{a} = \boldsymbol{b}$, i.e. the case of the self symmetrical state $(\boldsymbol{a}, \boldsymbol{a})$, among its four successor states (Fig. 5.6), there exist two states at time $k + 1$ that are always self symmetrical. The others are mutual symmetrical. Since one of the states in a pair of mutual symmetrical states at time $k + 1$ is removed, the edge is moved to connect to the remaining one. Since the two edges have the same inputs and outputs, the two edges are merged to a single edge. This means that the number of edges from the self symmetrical state is three.

When $\boldsymbol{a} \neq \boldsymbol{b}$, i.e. the case of the mutual symmetrical state $(\boldsymbol{a}, \boldsymbol{b})$, symmetrical state does not exist among its four successor states $(\mu(\boldsymbol{b}, u_1), \mu(\boldsymbol{a}, u_2))$, $u_1, u_2 \in \{0, 1\}$. Therefore, the number of edges from mutual symmetrical states is still four. Note that there probably exist two different edges $e((\boldsymbol{a}, \boldsymbol{b}), (\tilde{\boldsymbol{a}}, \tilde{\boldsymbol{b}}))$ and $e'((\boldsymbol{a}, \boldsymbol{b}), (\tilde{\boldsymbol{a}}, \tilde{\boldsymbol{b}}))$, $(\boldsymbol{a} \neq \boldsymbol{b})$, which originate from the same state and connect to the same successor state, since their inputs (outputs) are different.

In summary of the above discussion, we give the steps to simplify the sum trellis. First, remove one of the states in a pair of symmetrical states from the sum trellis. Second, remove the four edges originated from each of the removed states at time k. Third, move

Table 5.1: The number of states and edges in T_{sum} and Γ_{ssm}.

	T_{sum}		Γ_{ssm}	
M	state	edge	state	edge
	M^2	$4M^2$	$M(M+1)/2$	$2M(M-1)+3M$
2	4	16	3	10
4	16	64	10	36
8	64	256	36	136
16	246	1024	136	528

the edges, connected to the removed state at time $k+1$, to its mutual state at time $k+1$. Fourth, merge the two edges between a self symmetrical state at time k and a same state at time $k+1$. We finally deduce the sum trellis to a simplified sum trellis Γ_{ssm}.

Table 5.1 gives the numbers of states and edges in the simplified sum trellis. In the sum trellis, the number of the self symmetrical state $(\boldsymbol{a}, \boldsymbol{a})$, $\boldsymbol{a} \in \mathbb{A}$, is $M = |\mathbb{A}|$. The rest are the number of $M^2 - M$ mutual symmetrical states. By removing half of the mutual symmetrical states, the total number of states in the simplified sum trellis becomes $M(M+1)/2$. The number of edges in the trellis is $((M^2 - M)/2) * 4 + M * 3$. We see that the number of states and edges in the simplified sum trellis Γ_{ssm} is nearly half in sum trellis T_{sum}. This means that nearly half of the computational complexity can be decreased when decoding at the relay is with Γ_{ssm}. In addition, four examples of state numbers and edge numbers in T_{sum}

and Γ_{ssm} are given for $M = 2$ to 16 in Table 5.1.

Figure 5.5(d) is an example of a simplified sum trellis obtained from the sum trellis of Fig. 5.5(c). In the sum trellis of Fig. 5.5(c), there are four self symmetrical states and six pairs of mutual symmetrical states. Removing one of the states in a pair of symmetrical states reduces the number of states to 10 (see the left side of the trellis in Fig. 5.5(d)). Edges originated from the removed states at the left side are removed. For the removed states at the right side, the edges connected to the state are moved to connect to its symmetrical state. For instance, the edges $e(0000, 1000)$, $e(0001, 1000)$, $e(0101, 1000)$ are moved to connect to state 0010, denoted by $e'(0000, 0010)$, $e'(0001, 0010)$, $e'(0101, 0010)$. For the self symmetrical state, its two edges are merged. For instance, for state 0000, the two edges $e(0000, 0010)$ and $e'(0000, 0010)$ are merged to a single edge $e(0000, 0010)$. By the way, there are two edges between states 0001 and 0010, and two edges between states 1011 and 0111.

Remark: The simplified sum trellis is different from the reduced-state trellis in [40]. The reduced-state trellis is given based on a simple state combination, i.e. forming a state set by an XOR operation of two state sets of two users' codes [40–42]. Also, the input (output) is the XOR of the two inputs (outputs) of two users. This means that the decoding with the reduced-state trellis is to directly obtain the XORed messages without distinguishing three values of the arithmetic sum. Therefore, the trellis degrades the decoding performance, and is not suitable for iterative decoding. In our work on the simplified sum trellis, although we remove one of the states in a pair of mutual symmetrical states, the simplified one is equivalent to the sum trellis without any degradation of the decoding performance, and is suitable for iterative decoding.

5.4 Simulation

In this section, we give the performance evaluation of the proposed two-user turbo decoding scheme with the simplified sum trellis. We only focus on the evaluation of bit error rate (BER) of decoded bit $u_1 \oplus u_2$ at the relay. In the simulation, SNR per user is $1/\sigma^2$ (the total transmit power of the two users is 2, and the average power of each one is 1). The message length is 4096 bits.

Before proceeding, let's first evaluate the BER performance of the two-user decoding with convolutional code (i.e. the component code of the turbo code) over AWGN channel. The polynomial of the convolutional code used at sources S_1 and S_2 is $(g_0, g_1) = (13, 15)_8$, where g_0 and g_1 are the feedback polynomial and the forward polynomial in octal form [43], respectively. In the decoder at relay R, the decoding is based on the BCJR algorithm with the simplified sum trellis, and but without any iterations. In Fig. 5.7, we see that the BER with the simplified sum trellis is the same as that with the sum trellis. We verify that the simplification of sum trellis does not degrade any decoding performance of BER. Moreover, for comparison we give the BER with the reduced-state trellis in [40]. We observe that the proposed scheme outperforms that with reduced-state trellis, e.g. by about 0.8 dB gain at the BER of 10^{-5}. The degradation of reduced-state trellis is due to that the decoding with the trellis is to obtain the XORed messages without distinguishing the three values of the

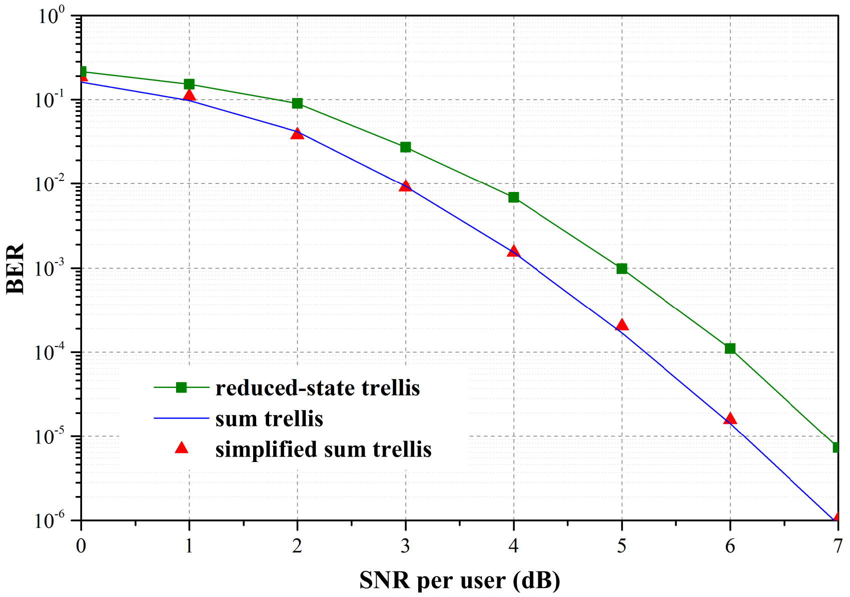


Figure 5.7: BERs of decoded bit $u_1 \oplus u_2$ with the reduced-state trellis [40], sum trellis, and simplified one at the relay with convolutional code (AWGN).

arithmetic sum.

Let us now return to the two-user turbo decoding. In Fig. 5.8, we give a BER performance of the two-user turbo decoding scheme with the simplified sum trellis when TWRC is AWGN channel. The polynomial of component codes of the turbo code is also $(g_0, g_1) = (13, 15)_8$. We see that the BER of the proposed scheme is the same as that of the decoding scheme with the sum trellis, although the decoding complexity of the proposed scheme is nearly half of the conventional one as we stated in Sect. 5.3.2. In addition, we give BER of a two-user decoding with MMSE estimation [37] for comparison. We observe that the proposed scheme outperforms the MMSE scheme by about 0.6 dB at BER of 10^{-5}. Note that in the MMSE scheme, the relay estimates the superimposed signal to sum $\boldsymbol{c}_1 + \boldsymbol{c}_2$

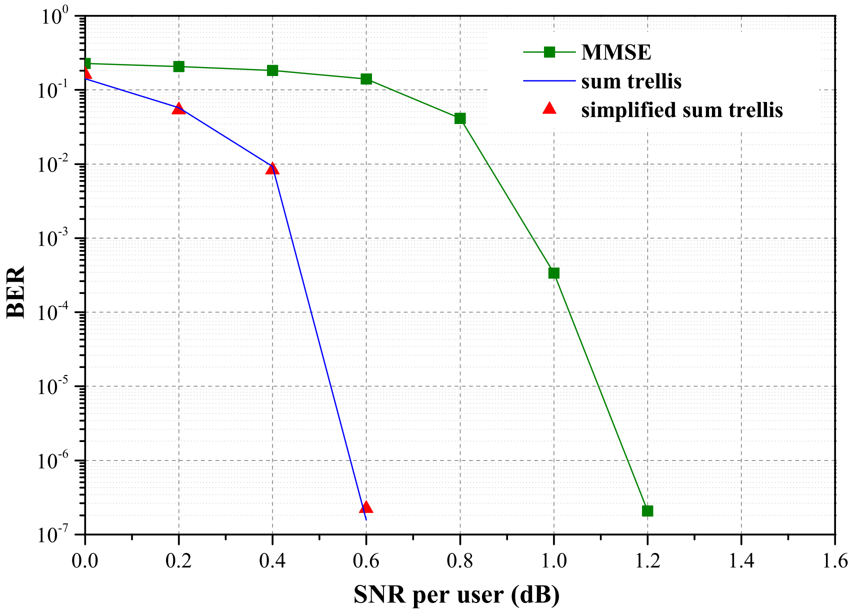


Figure 5.8: BERs of decoded bit $u_1 \oplus u_2$ with the sum trellis and simplified one and with MMSE [37] at the relay with turbo code (AWGN).

of two codewords, and decodes it to network-coded message $\boldsymbol{u}_1 \oplus \boldsymbol{u}_2$.

5.5 Conclusion

We combined binary turbo coding to PNC in the TWRC. We proposed a two-user turbo decoding scheme with a simplified sum trellis. For two-user iterative decoding at the relay, the component decoder decodes the superimposed signal to the arithmetic sum of two users' messages. For the two-user decoding, we employed a simplified sum trellis, which is obtained by removing one of the states in a pair of mutual symmetrical states from a sum trellis. This removal reduces the decoding complexity to half of that with the sum trellis, and does not degrade decoding performance over AWGN channel since two output sequences from the pair of mutual symmetrical states are the same.

Chapter 6

Two-User Turbo Decoding with Simplified Sum Trellis for Fading TWRC

6.1 Introduction

In the Gaussian TWRC channel model, the received signal is assumed to be affected only by a constant attenuation and a constant delay. In wireless communications, digital transmission often needs a more elaborate model, since it may be necessary to account for propagation vagaries, referred to as " fading, " which affect the signal strength. It may either be due to multipath propagation, referred to as multipath induced fading, or due to shadowing from obstacles affecting the wave propagation, sometimes referred to as shadow fading. The fading may vary with time, geographical position or radio frequency, and is often modeled as a random process. A fading channel is a communication channel comprising fading. In this chapter, we consider the two-user turbo decoding for fading TWRC.

Now we give the model of fading TWRC. Consider the fading TWRC, where h_1 (h_2) is the channel coefficient between relay R and user S_1 (S_2). The received signal at the relay is

$$y_R = h_1 c_1 + h_2 c_2 + z \tag{6.1}$$

where $c_1, c_2 \in \{+1, -1\}$ are transmitted signal which are either the systematical bits u_1, u_2 or the parity bits $\tilde{u}_1, \tilde{u}_2$ (see Fig. 5.5).

When the channel coefficients h_1 and h_2 are known, for the Gaussian noise z, the channel transition probability density function at relay R is given by

$$p(y_R|(c_1, c_2), h_1, h_2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{(y_R - (h_1 c_1 + h_2 c_2))^2}{2\sigma^2} \right\}. \tag{6.2}$$

We observe that

$$h_1 c_1 + h_2 c_2 = \begin{cases} h_1 - h_2, & \text{if} \quad c_1 + c_2 = 0^+ \\ -h_1 + h_2, & \text{if} \quad c_1 + c_2 = 0^-. \end{cases} \tag{6.3}$$

Generally, $h_1 - h_2 \neq -h_1 + h_2$, if $h_1 \neq h_2$. Thus, in (6.3) we distinguish these two cases: $c_1 = 1, c_2 = -1$ (i.e. $c_1 + c_2 = 0^+$) and $c_1 = -1, c_2 = 1$ (i.e. $c_1 + c_2 = 0^-$).

Recall that the sum trellis and the simplified one in Section 5.3 are for the Gaussian TWRC ($h_1 = h_2 = 1$), where $h_1c_1+h_2c_2$ is always equal to 0 in both cases above. Since in the two trellises constructed for Gaussian TWRC, these two cases are not distinguished. They can not be directly applied to fading TWRC. A modification of these trellises are required.

In this chapter, we propose a two-user turbo decoding algorithm for fading TWRC. A modified simplified sum trellis for fading TWRC is obtained distinguishing the cases of $c_1 = 1, c_2 = -1$ and $c_1 = -1, c_2 = 1$. The transition probability density function from a state to next state in simplified sum trellis is approximately computed. The approximate decoding algorithm preserves low complexity over Gaussian TWRC, without much performance degradation.

6.2 Simplified Sum Trellis for Fading TWRC

Let the edge in trellises, connecting the state $(\boldsymbol{a}, \boldsymbol{b})$ at time k and the state $(\tilde{\boldsymbol{a}}, \tilde{\boldsymbol{b}})$ at time $k+1$, be with input $u_1 + u_2$ and output $\tilde{u_1} + \tilde{u_2}$. For simplicity, we refer the edge whose input is $u_1 + u_2 = 1$ or output is $\tilde{u_1} + \tilde{u_2} = 1$ to 1-in-edge. Since c_1, c_2 are transmitted signal which is associated with either the systematical bits u_1, u_2 or the parity bits $\tilde{u}_1, \tilde{u}_2$, the cases of $c_1 + c_2 = 0^+$ (0^-) means input $u_1 + u_2 = 1^+$ (1^-) or output $\tilde{u_1} + \tilde{u_2} = 1^+$ (1^-). Similarly, we refer the edge whose input is $u_1 + u_2 = 1^+$ (1^-) or output is $\tilde{u_1} + \tilde{u_2} = 1^+$ (1^-) to 1^+ (1^-)-in-edge.

To apply the sum trellis in Section 5.3 to fading TWRC, a modification is to simply distinguish 1-in-edge into 1^+-in-edge or 1^--in-edge. Moreover, similar to the simplification on the sum trellis in Section 5.3, we have a simplification for fading as follows: First, remove one of the states in a pair of symmetrical states from the sum trellis. Second, for the edges originated from each of the removed states at time k, move the 1-in-edges to the symmetrical state at time k, and remove the remaining edges. Lastly, move the edges, connected to the removed state at time $k+1$, to its mutual state at time $k+1$. As a result, we deduce the sum trellis for fading to a modified sum trellis.

Note that for fading TWRC the 1-in-edge is moved to its symmetrical state, while for Gaussian TWRC the edges originated from the removed states are all removed. If there exists a 1-in-edge from a state at k to a state at $k+1$, there definitely exists a parallel 1-in-edge between these two states. In fact, one of these parallel is 1^+-in-edge, and the other is 1^--in-edge, since input $u_1 + u_2$ (or output $\tilde{u}_1 + \tilde{u}_1$) of these two edges are same.

Decoding with the modified sum trellis does not degrade any performance, but still has a high decoding complexity since the number of edges in the trellis has not been deduced much. Next, for the purpose of low complexity, we describe a two-user approximate decoding algorithm with the modified sum trellis.

Generally, the algorithm is similar to one stated in Sect. 5.3 with the exception of computing the transition probability density function of the two parallel edges. Let the 1^+-in-edge be labeled by $((u_1, u_2), (\tilde{u}_1, \tilde{u}_2))$. The 1^--in-edge is then labeled by $((u_2, u_1), (\tilde{u}_2, \tilde{u}_1))$. Over a block fading channel with independent noise, for the 1^+-in-edge, the channel transition probability density function of two symbols, i.e. $u_1h_1 + u_2h_2$ and $\tilde{u}_1h_1 + \tilde{u}_2h_2$ within a block, is the product of ones of two single-symbols (see (6.2)). That is

$$p(y_u, y_p|(u_1, u_2), (\tilde{u}_1, \tilde{u}_2), h_1, h_2) = p(y_u|(u_1, u_2), h_1, h_2)p(y_p|(\tilde{u}_1, \tilde{u}_2), h_1, h_2). \tag{6.4}$$

Also, for the 1^--in-edge, we have

$$p(y_u, y_p|(u_2, u_1), (\tilde{u}_2, \tilde{u}_1), h_1, h_2) = p(y_u|(u_2, u_1), h_1, h_2)p(y_p|(\tilde{u}_2, \tilde{u}_1), h_1, h_2). \tag{6.5}$$

We compute the transition probability density function from state $(\boldsymbol{a}, \boldsymbol{b})$ to the state $(\tilde{\boldsymbol{a}}, \tilde{\boldsymbol{b}})$ in BCJR algorithm by choosing the higher one between the parallel edges, i.e.

$$\begin{aligned} & p(y_u, y_p|(\boldsymbol{a}, \boldsymbol{b}) \to (\tilde{\boldsymbol{a}}, \tilde{\boldsymbol{b}}), h_1, h_2) \\ = \ & \max\{p(y_u, y_p|(u_1, u_2), (\tilde{u}_1, \tilde{u}_2), h_1, h_2), \ p(y_u, y_p|(u_2, u_1), (\tilde{u}_2, \tilde{u}_1), h_1, h_2)\}. \end{aligned} \tag{6.6}$$

Due to this approximation, decoding with the modified sum trellis results in a little degradation of performance (see Figs. 6.1 and 6.2). Fortunately, the decoding complexity with the reduced sum trellis is almost the same as that with the simplified sum trellis in Section 5.3.2 since the two parallel edges in the modified sum trellis are viewed as a single edge in decoding procedure. To summarize, the approximate decoding algorithm for fading TWRC can be seen as a decoding with the simplified sum trellis in Section 5.3.2 with the transition probability density function of 1-in-edge being approximated by (6.6).

6.3 Simulation

In this section, we give BER performance of the two-user approximate decoding algorithm with the simplified sum trellis over block Rayleigh fading TWRC, where with a block, the

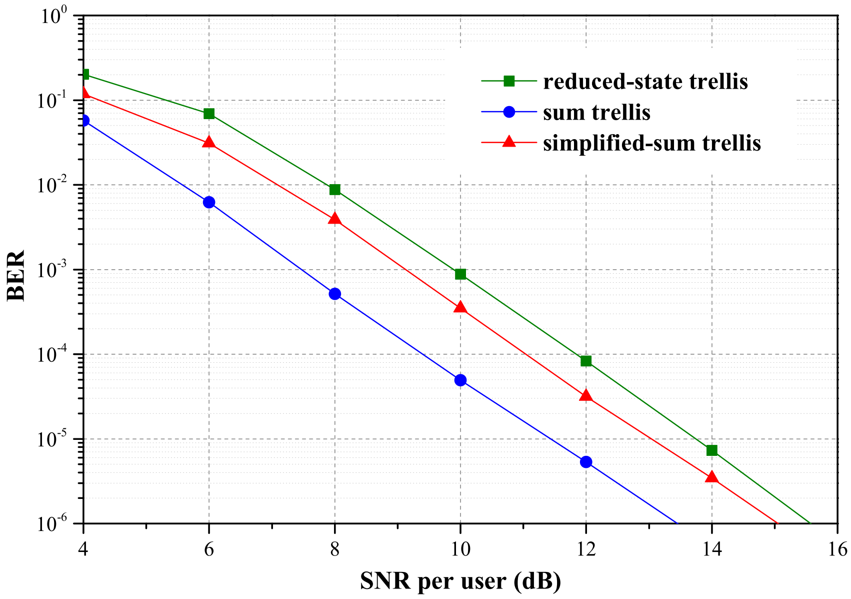


Figure 6.1: BERs of decoded bit $u_1 \oplus u_2$ with the reduced-state trellis [40], sum trellis, and simplified one at the relay with convolutional code (Rayleigh fading TWRC).

channel coefficients are constant.

In our simulation, we assume that fading coefficients are perfect known at the relay. The block length is set to 2. In Fig. 6.1, we give the BERs of convolutional code's BCJR decoding, with the same code polynomial as in Fig. 5.7. Indeed, our decoding scheme still outperforms the conventional one, even over fading channel. Figure 6.2 shows the BERs of two-user turbo decoding. In both Figs. 6.1 and 6.2, we see that BERs with the simplified sum trellis degrade, compared with those with the sum trellis for fading. Note that the decoding with the simplified sum trellis preserves a low complexity, as we illustrated in Table 5.1.

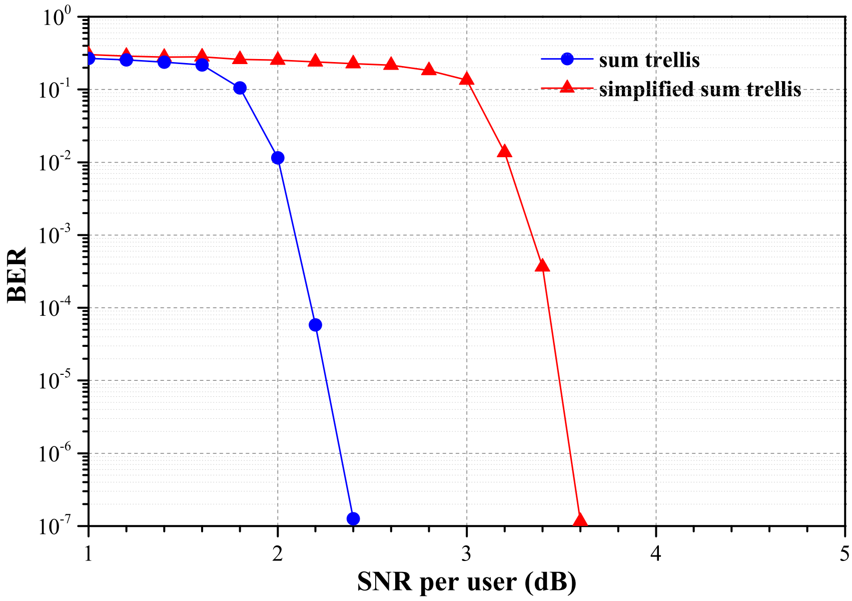


Figure 6.2: BERs of decoded bit $u_1 \oplus u_2$ with the sum trellis and simplified one at the relay (Rayleigh fading TWRC).

6.4 Conclusion

In this chapter, we proposed a two-user turbo decoding algorithm for fading TWRC. The simplified sum trellis for fading TWRC distinguish the cases of $c_1 = 1, c_2 = -1$ and $c_1 = -1, c_2 = 1$. The transition probability density function from a state to next state in simplified sum trellis is approximately computed. The approximate decoding algorithm preserves low complexity over Gaussian TWRC, without much performance degradation.

Chapter 7

Concluding Remarks

Coding and decoding for multiuser communication systems are investigated. In this dissertation, we considered two channel models: multiple-access adder channel (MAAC) and two-way relay channel (TWRC).

For MAAC, we proposed a coding scheme of $(k+1)$-ary error-correcting signature codes for noisy MAAC. The main coding scheme is presented that given a signature matrix A and a difference matrix $D = D^{+} - D^{-}$ a priori, we obtained a larger signature matrix by replacing each element in Hadamard matrix with A, or D^{+}, or D^{-} depending on the values of elements and their locations in Hadamard matrix. The set of rows of proposed matrix gave an error-correcting signature code. Introducing the difference matrix makes it possible to construct error-correcting signature code whose sum rate is increased with an increase in the order of Hadamard matrix. We gave binary and non-binary signature codes. They are the best codes for MAAC, in the sense that they have highest sum rates known.

For TWRC, we proposed a low-complexity two-user turbo decoding scheme when turbo codes are applied in two users. Simplified sum trellis is provided for two-user iterative decoding at the relay to decrease the decoding complexity. It is obtained by removing one of the states in a pair of mutual symmetrical states from a sum trellis. For Gaussian TWRC, decoding based on simplified sum trellis reduces the decoding complexity to half of that with the sum trellis, and does not degrade decoding performance since two output sequences from the pair of mutual symmetrical states are the same. For fading TWRC, the transition probability density function from a state to next state in simplified sum trellis is approximately computed. The approximate decoding algorithm preserves low decoding complexity over Gaussian TWRC, without much performance degradation.

There are a lot of open problems on error-correcting signature coding for noisy MAAC that seem to deserve further investigation. The most interesting one is the following problem: For given k, n and δ, what is the bound of maximum number of tolerate users for MAAC, and how to construct a signature code to approach the bound of maximum number.

Acknowledgments

I would like to express the deepest appreciation to my advisors, Prof. Jun Cheng and Prof. Yoichiro Watanabe, for their continual and excellent guidance, caring, patience, and providing me with an excellent atmosphere for doing research. I would never have been able to finish my dissertation without their guidance.

I also want to thank Prof. Ying Li, who is my advisor when I was a graduate student. Thanks for her drawing me to the interesting field of error-correcting codes.

I would like to thank G. Song, W. Hou, M. He, S. Liu, H. Han. A number of valuable discussions with them in information theory seminar are also acknowledged. I also wishes to express thanks to Prof. P. Li of City University of Hong Kong and Prof. H. Yagi from University of Electro-Communications for their useful discussions in this work.

The scholarships supplied by Ministry of Education, Culture, Sports, Science and Technology (MEXT) and the Iwatani Naoji Foundation are the financial support to me during this work.

Finally, I would like to thank my parents and my husband. They were always sup-

porting me and encouraging me with their best wishes. I also want to thank my son, his born and laughing give me endless energy to complete my work.

Bibliography

[1] C. E. Shannon, "Two-way communication channels," in Proc. *Berkeley Symp. Mathematical statistics and probability*, J. Newman, Ed. Berkeley, CA: Univ. Califonia Press, vol. 1, pp. 611-644, 1961.

[2] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Second Edition, New York: Wiley, 2006.

[3] T. Bohman, "A sum packing problem of Erdős and the Conway–Guy sequence," in Proc. *Amer. Math. Soc.*, vol.124, no.12, pp.3627–3636, Dec. 1996.

[4] T. Kasami and S. Lin, "Coding for a multiple-access channel," *IEEE Trans. Inform. Theory*, vol. IT-22, no. 2, pp. 129–137, Mar. 1976.

[5] E. J. Weldon, Jr., "Coding for a multiple access channel," *Inform. Contr.,* vol. 36, no. 3, pp. 256–274, 1978.

[6] S. C. Chang and E. J. Weldon, Jr., "Coding for T-user multiple-access chanels," *IEEE Trans. Inform. Theory*, vol. IT-25, no. 6, pp. 684–691, Nov. 1979.

[7] S. C. Chang, "Further results on coding for T-user multiple-access channels," *IEEE Trans. Inform. Theory*, vol. IT-30, no. 2, pp. 411–415, Mar. 1984.

[8] T. J. Ferguson, "Generalized T-user codes for multiple-access channels," *IEEE Trans. Inform. Theory*, vol. IT-28, no. 5, pp. 775–778, Sept. 1979.

[9] B. L. Hughes and A. B. Cooper, III, "Nearly optimal multiuser codes for the binary adder channels," *IEEE Trans. Inform. Theory*, vol. IT-42, no. 2, pp. 387–398, Mar. 1996.

[10] G. H. Khachatrian and S. S. Martirossian, III, "Code construction for the T-user noiseless adder channel," *IEEE Trans. Inform. Theory*, vol. IT-44, no. 5, pp. 1953–1957, Sept. 1998.

[11] J. Cheng and Y. Watanabe, "Multi-user k-ary code for noisy multiple-access adder channel," *IEEE Trans. Inform. Theory*, vol. IT-47, no. 6, pp. 2603-2607, Sept. 2001.

[12] J. Cheng and Y. Watanabe, "Spreading set with error correction for multiple-access adder channel," *IEEE Trans. Inform. Theory*, vol. IT-52, no. 12, pp. 5524–5529, Dec. 2006.

[13] T. Ericson and L. Györfi, "Superimposed codes in R^n," *IEEE Trans. Inform. Theory*, vol. IT-34, No. 4, pp. 877–880, July 1988.

[14] P. Z. Fan, M. Darnell, and B. Honary, "Superimposed codes for the multiaccess binary adder channel," *IEEE Trans. Inform. Theory*, vol. IT-41, No. 4, pp. 1178–1182, July 1995.

[15] D. G. Cantor and W. H. Mills, "Determining a subset from certain combinatorial properties," *Can. J. Math.,* vol. 18, pp. 42-48, 1966.

[16] B. Lindström, "Determining subsets by unramified experiments," in *A Survey of statistical Design and Linear Models*, ed. J. N. Srivastava, North-Holland, New York, 1975.

[17] S. S. Martirosyan and G. G. Khachatryan, "Construction of signature codes and the coin weighing problem," *Probl. Inform. Transm.*, vol. IT-25, pp. 334-335, Oct.-Dec. 1989.

[18] W. H. Mow, "Recursive constructions of detecting matrices for multiuser coding: a unifying approach," *IEEE Trans. Inform. Theory*, vol. IT-55, no. 1, pp. 93-98, Jan. 2009.

[19] D. Jevtić, "On families of sets of integral vectors whose representatives form sum-distinct sets," *SIAM J. Discrete Math.*, vol. 8, no. 4, pp. 652-660, Nov. 1995.

[20] D. Jevtić, "Disjoint uniquely decodable codebooks for noiseless synchronized multiple-access adder channels generated by integer sets," *IEEE Trans. Inform. Theory*, vol. IT-38, No. 3, pp. 1142–1146, May 1992.

[21] J. Cheng and Y. Watanabe, "T-user code with arbitrary code length for multiple-access adder channel," *IEICE Trans. Fundamentals*, vol. E82-A, no. 10, pp. 2011–2016, Oct. 1999.

[22] D. Jevtić, "On sum distinct sets of integral vectors," *ARS Combinatoria*, vol. 45, pp. 87-95, April 1997.

[23] L. Györfi and B. Laczay, "Signature coding and information transfer for the multiple access adder channel," in Proc. *IEEE Information Theory Workshop* (*ITW2004*), pp. 242-246, San Antonio, Texas, USA, 2004.

[24] F. J. MacWilliams and N. J. Sloane, *The Theory of Error-Correcting Codes*, Amsterdam, The Netherlands: North-Holland, 1977.

[25] J. Cheng, K. Kamoi, and Y. Watanabe, “Error-correcting signature code for multiple-access adder channel,” in Proc. *IEEE Int. Symposium on Information Theory* (*ISIT2005*), pp. 2036-2039, Sept. 4–9, 2005, Adelaide, Australia.

[26] J. Cheng, K. Kamoi, and Y. Watanabe, “User identification by signature code for noisy multiple-access adder channel,” in Proc. *IEEE Int. Symposium on Information Theory* (*ISIT2006*), pp. 1974-1977, July 9–14, 2006, Seattle, USA.

[27] J. Cheng, K. Kamoi, and Y. Watanabe, “Error-correcting non-binary signature code for multiple-access adder channel,” in Proc. *IEEE Int. Symposium on Information Theory* (*ISIT2007*), pp. 2571-2574, June 24–29, 2007, Nice, France.

[28] S. Zhang, S. C. Liew, and P. P. Lam, “Hot topic: physical layer network coding,” in Proc. *12th Annual International Conf. on Mobile Computing and Networking* (*MOBICOM’06*), pp.358-365, Los Angeles, Calif, USA, Sept. 2006.

[29] Y. Wu, P. A. Chou, and S. Y. Kung, “Information exchange in wireless networks with network coding and physical-layer broadcast,” Proc. *39th Annual Conf. on Information Sciences and Systems* (*CISS*), pp.1-6, The Johns Hopkins University, March 2005.

[30] K. Lu, S. Fu, Y. Qian, and H. Chen, “On capacity of random wireless networks with physical-layer network coding,” *IEEE Journal on Selected Areas in Communications*, vol.27, no.5, pp.763-772, June 2009.

[31] Y. Han, S. H. Ting, C. K. Ho, and W. H. Chin, "High rate two-way amplify-and-forward half-duplex relaying with OSTBC," in Proc. *IEEE International Vehicular Technology Conf.* (*VTC*), pp.2426-2430, Marina Bay, Singapore, May 2008.

[32] B. Rankov and A. Wittneben, "Spectral efficient protocols for halfduplex fading relay channels," *IEEE Journal on Selected Areas in Communications*, vol.25, no.2, pp.379-389, Feb. 2007.

[33] R. Louie, Y. Li, and B. Vucetic, "Practical physical layer network coding for two-way relay channels: performance analysis and comparison," *IEEE Trans. Wireless Communications*, vol.9, no.2, pp.764-777, Feb. 2010.

[34] P. Popovski and H. Yomo, "Wireless network coding by amplify-and-forward for bi-directional traffic flows," *IEEE Communication Letter*, vol.11, no.1, pp.16-18, Jan. 2007.

[35] P. Popovski and H. Yomo, "Physical network coding in two-way wireless relay channels," in Proc. *IEEE International Conf. on Communication* (*ICC*), pp.707-711, Glasgow, Scotland, June 2007.

[36] E. Peh, Y. Liang, and Y. Guan, "Power control for physical layer network coding in fading environments," in Proc. *IEEE Personal, Indoor and Mobile Radio Communication* (*PIMRC*), pp.1-5, Cannes, France, Sept. 2008.

[37] S. Zhang, S. C. Liew, and L. Lu, "Physical layer network coding schemes over finite and infinite fields," in Proc. *IEEE Global Telecommunications Conf.* (*GLOBECOM'08*), pp.3784-3789, New Orleans, USA, Nov.-Dec. 2008.

[38] T. Koike-Akino, P. Popovski, and V. Tarokh, “Optimized constellations for two-way wireless relaying with physical network coding,” *IEEE Journal on Selected Areas in Communications*, vol.27, no.5, pp.773-787, June 2009.

[39] S. Zhang and S. C. Liew, “Channel coding and decoding in a relay system operated with physical-layer network coding,” *IEEE Journal on Selected Areas in Communications*, vol.7, no.5, pp.88-796, June 2009.

[40] D. To and J. Choi, “Convolutional codes in two-way relay networks with physical-layer network coding,” *IEEE Trans. Wireless Communications*, vol.9, no.9, pp.2724-2729, Sept. 2010.

[41] J. Harshan and B. Sundar Rajan, “Finite signal-set capacity of two-user Gaussian multiple access channel,” in Proc. *IEEE Int. Symposium on Information Theory* (*ISIT*), pp.1203-1207, Toronto, Canada, July 2008.

[42] J. Harshan and B. Sundar Rajan, “Coding for two-user SISO and MIMO multiple access channels,” arXiv:0901.0168v3 [cs.IT], accessed Feb. 2009.

[43] S. Lin and D. J. Costello, *Error Control Coding: Fundamentals and Applications*, 2nd ed., Prentice-Hall, New Jersey, 2004.

[44] C. B. Schlegel and L. C. Perez, *Trellis and Turbo Coding*, Wiley-IEEE Press, New Jersey, 2004.

[45] L. C. Perez, J. Seghers, and D. J. Costello, “A distance spectrum interpretation of turbo codes,” *IEEE Trans. Inform. Theory*, vol. IT-42, no. 6, pp. 1698-1709, Nov. 1996.

Peer-Reviewed Publications

1. S. Lu, Y. Li, J. Cheng, and Y. Watanabe, "Two-user turbo decoding with simplified sum trellis in two-way relay channel," *IEICE Trans. Commun.*, vol.E96-B, vo.1, pp.73-80, Jan. 2013.

2. S. Lu, J. Cheng, and Y. Watanabe, "Recursive construction of $(k+1)$-ary error-correcting signature code for multiple-access adder channel," *IEICE Trans. Fundamentals*, vol.E96-A, No.12, 2013. (in press)

3. S. Lu, W. Hou, and J. Cheng, "A family of $(k+1)$-ary signature codes for noisy multiple-access adder channel," submitted to *IEEE Trans. Inform. Theory.*

4. S. Lu, Y. Li, and J. Cheng, "Low-complexity turbo decoding scheme for two-way relay network," in Proc. *IEEE Int. Conf. on Wireless Communications and Signal Processing* (*WCSP*), pp.1-5, Suzhou, China, Oct. 2010.

5. M. He, S. Lu, G. Song, J. Cheng, and Y. Watanabe, "Channel capacity with superposed M-PAM modulation," in Proc. *IEEE 4th Int. Conf. on Intellignet Computation*

Technology and Automation (*ICICTA*), pp.475-478, Shenzhen, China, Mar. 2011.

6. S. Lu, J. Cheng, and Y. Watanabe, "Decoding for non-binary signature code," in Proc. *IEICE Int. Symposium on Information Theory and Its Applications* (*ISITA*), pp.382-386, Hawaii, USA, Oct. 2012.

7. S. Lu, J. Cheng, W. Hou, and Y. Watanabe, "Generalized construction of signature code for T-user multiple-access adder channel," in Proc. *IEEE Int. Symposium on Information Theory* (*ISIT*), pp.1655-1659, Istanbul, Turkey, July, 2013.

8. S. Lu, W. Hou, and J. Cheng, "Construction of error-correcting signature code on Hadamard matrix," in Proc. *IICREST third Int. Symposium Radio System and Space Plasma* (*ISRSSP*), pp.57-63, Sofia, Bulgaria, Aug. 2013.

Technical Publications

1. S. Lu and Y. Li, "Research on joint turbo decoding and network coding in two-way relay channel," in Proc. *CIE (Chinese Institute of Electronics) 16th Symposium on Information Theory* (*CSIT*), pp.624-630, Beijing, China, Oct. 2009.

2. S. Lu, J. Cheng, and Y. Watanabe, "Two-user decoding for two-way relay network," in Proc. *the 7th Joint Symposium between Doshisha University and Chonnam National University*, pp.132-127, Gwangju, Korea, Aug. 2010.

3. S. Lu, J. Cheng, and Y. Li, "Low complexity decoding scheme for two-way relay network," in Proc. *SITA the 33rd Symposium on Information Theory and its Application* (*SITA*), pp.806-810, Nagano, Japan, Nov. 2010.

4. S. Lu, K. Iwata, J. Cheng, and Y. Watanabe, "Viterbi decoding based on simplified XOR trellis in two-way relay network," in Proc. *IEICE General Conference*, B-5-50, pp.436, Tokyo, Japan, Feb. 2011.

5. S. Fukagawa, S. Lu, J. Cheng, and Y. Watanabe, "Coded cooperative communication

system with encrypted SCCC code," in Proc. *IEICE Society Conference*, A-6-7, pp.158, Hokkaido, Japan, Sept. 2011. (in Japanese)

6. W. Hou, S. Lu, J. Cheng, and Y. Watanabe, "Preprocess scheme of error?correction code for physical layer security," in Proc. *IEICE General Conf.*, B-5-5, pp.404, Okayama, Japan, Mar. 2012.

7. S. Lu, J. Cheng, and Y. Watanabe, "Decoding scheme for non-binary signature code, *IEICE Technical Report*, vol.112, no.124, IT2012-30, pp.125-129, July 2012.

8. S. Lu, J. Cheng, and Y. Watanabe, "Signature code for multiple access adder channel," *IEICE Technical Report*, vol.112, no.215, IT2012-39, pp.47-52, Sept. 2012.

9. K. Iwata, S. Lu, J. Cheng, and Y. Watanabe, "Low-complexity decoding of trellis coded modulation over two-way relay channel," in Proc. *IEICE General Conference*, B-8-22, pp. 257, Gifu, Japan, Mar. 2013. (in Japanese)

10. S. Lu, W. Hou, and J. Cheng, "Coding scheme of error-correcting signature codes from Hadamard matrix," in Proc. *IEICE the 36th Symposium on Information Theory and Its Application* (*SITA*), Ito, Shizuoka, Japan, Nov. 2013.

11. W. Hou, S. Lu, and J. Cheng, "Rate compatibility of spatially coupled LDPC codes via repeat-accumulation extension," in Proc. *IEICE the 36th Symposium on Information Theory and Its Application* (*SITA*), Ito, Shizuoka, Japan, Nov. 2013.

12. S. Lu, Wei Hou, and Jun Cheng, "A family of error-correcting signature code," in

Proc. *51th Joint Symposium on Research Centers, Science and Engineering Research Institute*, Doshisha University, Kyoto, Japan, Dec. 2013.

Patent

1. S. Lu, Y. Li, and Y. Sun, “Network coding based on turbo decoding,” China Patent CN101674091, Sept. 2012.