

國立臺灣大學電機資訊學院電信工程研究所

碩士論文

Graduate Institute of Communication Engineering
College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

分增益多重接取系統與隨機存取系統

Gain-Division Multiple Access and Random Access

楊裕欽

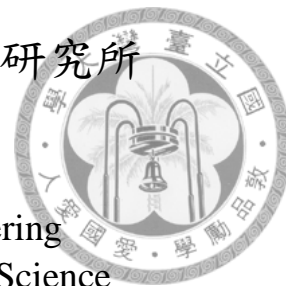
Yu-Qin Yang

指導教授：林茂昭博士

Advisor: Mao-Chao Lin, Ph.D.

中華民國109 年7 月

July, 2020



國立臺灣大學碩士學位論文

口試委員會審定書

分增益多重接取系統與隨機存取系統

Gain-Division Multiple Access and Random Access

本論文係楊裕欽君（學號 R07942121）在國立臺灣大學電信工程學研究所完成之碩士學位論文，於民國 109 年 7 月 16 日承下列考試委員審查通過及口試及格，特此證明

口試委員：

林其明

（簽名）

（指導教授）

李育清

趙叔平

呂世諱

蘇賜麟

所 長

蘇火保

（簽名）



致謝

首先要感謝我的指導教授林茂昭教授，在研究部分給我們很大的空間來發揮自己的想像力，時而給我們指點迷津，感謝李世楷教授時而指點我研究的盲點，讓我不會陷入錯誤迴圈，感謝李晃昌教授適時的給予一些幫助。再來也要感謝學長們，感謝會名跟亭宣學長的搞笑為實驗室帶來歡樂，讓實驗室和樂融融，感謝永超跟家俊學長的技術指導讓我受益良多，讓我節省了很多研究時間。接著感謝我的同屆同學，感謝籌尹跟我一起頑皮、感謝得生帶來話題、感謝蕙如給我問問題，大家也都不吝嗇給我一些研究上的想法，讓我覺得很感動。再來謝謝我的學弟妹們，柏翰、政晏、律琦，雖然碩二比較少去實驗室，沒有很常看到你們，但是有空可以再辦實驗室活動來聚一聚，最後再次感謝 503 實驗室的大家。



中文摘要

分增益多重存取技術是一種最近被提出的多重存取技術，它是利用各個使用者獨立的通道係數來達成多重存取，因此多個使用者可以使用同樣的通道資源同時進行傳送。本篇論文對分增益多重存取技術作進一步研究並研究將其應用於隨機存取系統。

我們提出了分增益多重存取技術的疊加信號與 log-likelihood ratio 的較通用的表示形式，讓其在實作上更為簡單方便。此外，針對使用極化碼的分增益多重存取，我們也提出了聯合通道解碼方式，讓分增益多重存取技術有更好的效果。

針對先前文獻中基於分群演算法通道盲測之技術，我們也提出了改進方式以取得更接近克拉馬-羅下限的性能。我們也提出了適用於各種星座圖調變以及認意使用者數目來求取通道係數的通用法則。

此外，我們將分增益多重存取技術應用在隨機存取的通道上，利用扎德奧夫-朱序列來進行同步分析和輔助通道估測。性能遠優於不考慮多重存取設計的一般隨機存取。

關鍵字：多重存取、衰弱通道、通道估計、盲蔽估測、分群演算法、極化碼、聯合通道解碼、隨機存取、扎德奧夫-朱序列、同步。



Abstract

Gain division multiple access (GDMA) is a recently proposed multiple access technique that allows multiple users to share the same resource by exploiting distinct channel coefficients associated with distinct users. In this thesis, we conduct an extended study of GDMA and also its applications to random access.

We propose a general formula for expressing the superimposed signal of all users and the log-likelihood ratio of each user bit. As to the polar coded GDMA, we propose a joint channel decoder that achieved better BLER performance as compared to the decoding using separate decoders.

In the literature, blind estimation schemes based on the clustering algorithm for the GDMA system were proposed. In this thesis, we proposed improved methods as that the channel estimation can be closer to the Cramer Rao bound. In the literature, blind estimation schemes based on the clustering algorithm allow only a small number of users and modulation with small constellation sizes. We propose a method that can be applied to a random user number and a random constellation.

In the random access techniques, uncoordinated users transmit packets randomly. We introduced a random access system using GDMA. Zadoff-Chu sequence is used to assist the acquisition of synchronization and channel estimation. The resultant throughput is superior to the conventional random access system, which fails upon collision without using multiple access.

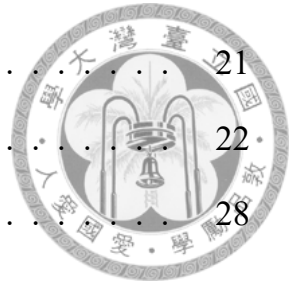
Key words: Multiple access, fading channel, channel estimation, blind estimation, clustering algorithm, Polar codes, Joint channel decoder, Random access, Zadoff-Chu sequence, Synchronization.



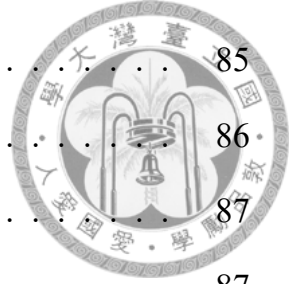
Contents

| | |
|--|----------|
| 口試委員會審定書 | i |
| 致謝 | ii |
| 中文摘要 | iii |
| Abstract | iv |
| Contents | vi |
| List of Figures | ix |
| List of Tables | xii |
| 1 Introduction | 1 |
| 2 Gain-Division Multiple Access | 5 |
| 2.1 Detection Principle | 5 |
| 2.2 Implementation Method | 11 |
| 2.2.1 BPSK-modulated GDMA Implementation | 11 |
| 2.2.2 QPSK-modulated GDMA Implementation | 14 |
| 2.2.3 PAM-modulated GDMA Implementation | 17 |
| 2.3 Joint channel decoding for the GDMA system | 21 |

| | | |
|----------|--|-----------|
| 2.3.1 | Introduction of Joint Channel Decoding | 21 |
| 2.3.2 | LDPC Coded GDMA system | 22 |
| 2.3.3 | Polar Coded GDMA system | 28 |
| 3 | Cluster-Based Channel Estimation | 43 |
| 3.1 | Review of Clustering Algorithm | 45 |
| 3.1.1 | K-Means Algorithm | 45 |
| 3.1.2 | The LBG Algorithm | 46 |
| 3.1.3 | K-Means++ Algorithm | 47 |
| 3.1.4 | Modified K-Means++ Algorithm | 48 |
| 3.1.5 | Gaussian Mixture Model | 49 |
| 3.2 | Cluster-based Estimation Analysis and Design | 51 |
| 3.2.1 | Clustering Sample Size Analysis | 52 |
| 3.2.2 | Proposed Clustering Algorithm | 56 |
| 3.2.3 | Derivation of Channel Coefficients | 62 |
| 3.3 | Resolving Phase Ambiguity | 68 |
| 3.3.1 | Differential Encoding | 69 |
| 3.3.2 | Noncoherent Block-Coded Modulation | 69 |
| 3.3.3 | Section Noncoherent Block-Coded Modulation | 71 |
| 4 | A Random Access Scheme based on Gain Division Multiple Access | 76 |
| 4.1 | Introduction of RACH and Grant | 77 |
| 4.2 | Introduction of Random Access System | 79 |
| 4.3 | System Model | 83 |
| 4.3.1 | OFDM-GDMA system | 83 |
| 4.3.2 | Doppler-effect and unknown timing drift | 83 |



| | | |
|----------|---|------------|
| 4.3.3 | Slotted RA OFDM-GDMA system | 85 |
| 4.3.4 | Unslotted RA OFDM-GDMA system | 86 |
| 4.4 | Preamble Synchronization | 87 |
| 4.4.1 | Preamble | 87 |
| 4.4.2 | Synchronization | 89 |
| 4.5 | Hybrid Channel Estimation | 91 |
| 4.6 | Simulation | 92 |
| 5 | Conclusion and Future Works | 98 |
| | Bibliography | 100 |



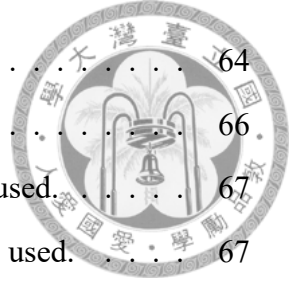


List of Figures

| | | |
|------|---|----|
| 2.1 | Transmitters of U users occupying the same resource. | 6 |
| 2.2 | GDMA detector for BPSK and $U = 2$ | 7 |
| 2.3 | An example of superimposed levels in the case of $m = 1$ and $U = 2$ | 8 |
| 2.4 | An example of superimposed levels in the case of $m = 1$ and $U = 3$ | 10 |
| 2.5 | BER performances of uncoded GDMA BPSK system over block Rayleigh fading channels. | 13 |
| 2.6 | QPSK modulation gray mapping. | 16 |
| 2.7 | BER performances of uncoded GDMA QPSK system over block Rayleigh fading channels. | 16 |
| 2.8 | 4-PAM modulation gray mapping. | 18 |
| 2.9 | 8-PAM modulation gray mapping. | 18 |
| 2.10 | BER performances of uncoded GDMA 4-PAM system over block Rayleigh fading channels. | 19 |
| 2.12 | Multi-level receiver in GDMA system for recovering the messages of U users. | 19 |
| 2.11 | BER performances of uncoded GDMA 8-PAM system over block Rayleigh fading channels. | 20 |
| 2.13 | Separated channel decoding in the case of $m = 1$ and $U = 2$ | 21 |
| 2.14 | Joint channel decoding in the case of $m = 1$ and $U = 2$ | 22 |
| 2.15 | The message passing algorithm along the edges in factor graph. | 24 |
| 2.16 | BLER performances of (1008, 504) LDPC-coded GDMA systems over quasi-static Rayleigh flat-fading channels. | 27 |

| | | |
|------|--|----|
| 2.17 | BLER performances of (1008, 504) LDPC-coded GDMA OFDM systems over quasi-static Rayleigh flat-fading channels with 16-subcarrier. | 27 |
| 2.18 | The graph factor for polar code, $N = 8, n = 3$ | 29 |
| 2.19 | The message passing function in factor graph. | 30 |
| 2.20 | The flow chart of polar code G-SPA decoder. | 33 |
| 2.21 | BLER performances of (256, 128) Polar-coded GDMA systems over quasi-static Rayleigh flat-fading channels. | 35 |
| 2.22 | BLER performances of (1024, 512) Polar-coded GDMA OFDM systems over quasi-static Rayleigh flat-fading channels with 16-subcarrier. | 35 |
| 2.23 | The factor graph of J-SCL upper layer (even branch) message passing. . . | 37 |
| 2.24 | The factor graph of J-SCL lower layer (odd branch) message passing. . . | 38 |
| 2.25 | BLER performances of (256, 128) Polar-coded GDMA systems over quasi-static Rayleigh flat-fading channels. | 40 |
| 2.26 | BLER performances of (1024, 512) Polar-coded GDMA OFDM systems over quasi-static Rayleigh flat-fading channels with 16-subcarrier. | 41 |
| 2.27 | BLER performances of Polar-coded GDMA OFDM systems over quasi-static Rayleigh flat-fading channels with 16-subcarrier and interleaver. . . | 41 |
| 3.1 | Superimposed levels and the corresponding received signals in the case of $m = 1$ and $U = 2$ | 43 |
| 3.2 | Splitting from q to get $q + \epsilon$ and $q - \epsilon$ with a perturbation vector ϵ | 47 |
| 3.3 | Grouping result of hard clustering by applying nearest neighbor rule. . . . | 49 |
| 3.4 | MSE of estimated channel coefficient over AWGN channels. | 53 |
| 3.5 | The probability of worst case with BPSK transmission for different user U . . | 55 |
| 3.6 | MSE of estimated channel coefficient with pair samples over AWGN channels. | 56 |
| 3.7 | The flow chart of proposed cluster-based channel estimation algorithm. . . | 57 |
| 3.8 | An example of cluster check. | 59 |
| 3.9 | The iteration number of estimated channel gain at $N_t = 128$ | 60 |
| 3.10 | An example of structure check. | 61 |

| | | |
|------|---|----|
| 3.11 | The steps to compose superimposed levels. | 64 |
| 3.12 | The steps of derivation of channel coefficients. | 66 |
| 3.13 | The MSE of estimated channel gain at $N_t = 128$. GMM is used. | 67 |
| 3.14 | The MSE of estimated channel gain at $N_t = 1000$. GMM is used. | 67 |
| 3.15 | Modulation scheme for DE-BPSK transmission. | 69 |
| 3.16 | BLER performances for a (128,64) Polar-coded GDMA system over quasi-static Rayleigh flat-fading single carrier channels. NBC and GMM are both used. | 70 |
| 3.17 | BLER performances of Polar-coded GDMA OFDM systems over quasi-static Rayleigh flat-fading channels with 16-subcarrier. | 75 |
| 4.1 | Two type of RACH process. | 79 |
| 4.2 | (a) An unslotted ALOHA system. (b) A slotted ALOHA system. | 79 |
| 4.3 | (a) The vulnerable period for unslotted system. (b) The vulnerable period for slotted system. | 80 |
| 4.4 | The throughput of ALOHA system. | 82 |
| 4.5 | BLER performances for a (1008,504) LDPC-coded OFDM-GDMA system where $U = 2$ users over frequency-selective Rayleigh fading channels. | 84 |
| 4.6 | (a) case 1 (b) case 2 | 86 |
| 4.7 | Synchronization and channel estimation process | 89 |
| 4.8 | MSE comparison of first and second channel estimation | 93 |
| 4.9 | ALOHA system simulation | 95 |
| 4.10 | No user collision case | 96 |
| 4.11 | Two user collision case | 96 |
| 4.12 | The APP detector variance without considering MUI noise term in Poisson distribution collision case | 97 |
| 4.13 | The APP detector variance without considering MUI noise term in Poisson distribution collision case | 97 |





List of Tables

- 2.1 Mapping rules for superimposed signal in the case of $m = 1$ and $U = 2$. . 8
- 2.2 Mapping rules for superimposed signal in the case of $m = 1$ and $U = 3$. . 10
- 3.1 The probability that at least one superimposed level is empty for $L = 4$. . 55



Chapter 1

Introduction

Multiple access techniques are critical topics in cellular mobile communications. In most of the multiple access systems, users are allowed to transmit signals simultaneously through different types of user-specific signatures. However, the number of users that can be served in a multiple access system is usually limited due to the constraint of user-specific signatures in the allocated resources. Compared to orthogonal multiple access schemes, non-orthogonal multiple access (NOMA) [1] that allows the overlapping of multiple users over a radio resource can make the resource allocation more flexible. In [1], a survey paper, briefly classified NOMA into different types such as the scrambling based Power Domain (PD) -NOMA, the spreading based SCMA, the interleaving based IDMA. In [2], a novel multiple access techniques called gain-division multiple access (GDMA) was investigated. We can define GDMA as a gain-based NOMA system.

Different from PD-NOMA that multiple users are superposed according to the specific power domain, the gain-division multiple access (GDMA) utilizes the different combinations of amplitudes and phase shifts in channel coefficients caused

by the independent fading channels. Hence, U users are allowed to share the same transmit resource with multi-level detection in the GDMA system. The detection scheme of GDMA originated from the concept of physical layer network coding (PLNC), which coordinates transmissions among nodes, and packet collisions in traditional wireless networks can be eliminated [3].

In forward error coding GDMA system, there are two types of detection principles: separate channel decoding (SCD) and joint channel decoding (JCD). GDMA-SCD is a straightforward method that separates individual information from superimposed signal information, and there is no exchange of information between collided users. GDMA-JCD is a more sophisticated method that employs all information of superimposed signals to decode. In [4], a joint channel decoding and physical-layer network coding (G-JCNC) [5] was applied for the GDMA LDPC coded system. However, there is no effective JCD for polar code. We introduced JCD decoders for a polar coded system, which can effectively improve the error performances.

Since the multiuser detection technique (MUD) in the GDMA system allows U users to share the same resource, each of the received symbols consisting of the signals transmitted from U users can be seen as a superimposed signal with 2^{mU} levels where m is the modulation order. A blind channel estimation scheme [4] was proposed for achieving spectral efficiency since no additional pilot signal is needed for channel estimation. Firstly, the system classifies the received symbols into 2^{mU} groups by applying a clustering algorithms [6] [7] [8], where the symbols

corresponding to the same superimposed level are grouped together, the estimates of the levels can be attained through the average of the symbols in each group. Secondly, the system utilizes the geometrical configuration of clustering estimated 2^{mU} levels to derive an estimated U channel gains. Thirdly, a phase ambiguity algorithm is used to remove the phase ambiguity.

In this thesis, we proposed a general adaptive clustering algorithm that can reduce the complexity and achieve Mean Square Error (MSE) for channel estimation closer to the Cramer Rao (CR) bound as compared to the algorithm used in [4]. In [4], the algorithm for the first and second stages can only be applied for a small amount of users U and for modulation with small order m . With this observation, we also propose an algorithm for deriving channel gains by recursive operations. This algorithm can be applied to any number U of users and any constellation size m .

Recently, random access (RA) protocols have acquired a lot of attention, not only from satellite communication but also from researchers active in fields such as Internet of Things (IoT) and machine-to-machine communications. Random access is a multiuser communication system in which users transmit packets over a shared channel, which is random. When there is more than one packets in the time duration, these packets collide, which results in transmission failure. Based on the conventional RA scheme (ALOHA system), several random access protocols were developed. Most of them used more channel resources to avoid collisions. To tackle the condition for which there are many device connections and few channel

resources, we introduce a novel RA scheme which resolves the collisions without additional channel resources by applying the GDMA concept. The scenarios of slotted, un-slotted, grant-based, and grant-free for the RA-GMDA system are investigated. The preamble signals are used for synchronization and channel estimation, where the preamble based channel estimation can be integrated into the clustering based channel estimation.

The organization of this thesis is as follows. In Chapter 2, basic concepts of the GDMA technique are reviewed ,and the system models are also illustrated. The modified cluster-based channel estimation for the GDMA system is proposed in Chapter 3 including the clustering algorithms and the schemes to resolve the phase ambiguity problem. Moreover, the GDMA based random access scheme is proposed in Chapter 4. Concluding remarks and future works are presented in Chapter 5.



Chapter 2

Gain-Division Multiple Access

In this chapter, we consider the scenario for uplink transmissions over independent fading channels, and the gain-division multiple access (GDMA) with a multiuser detection technique is reviewed for further discussions.

2.1 Detection Principle

A communication system in which U users are sharing the same transmit resource is considered, and the symbol timings of transmissions are assumed to be perfectly aligned. The block diagram of transmitters is shown in Fig. 2.1. The message bit sequence $\mathbf{d}_p = [d_u(1), d_u(2), \dots, d_u(k)]$ of user- u is encoded into the codeword $\mathbf{c}_u = [c_u(1), c_u(2), \dots, c_u(N_c)]$ by an (N_c, k) forward error-correction (FEC) code with code rate k/N_c , and the codeword \mathbf{c}_p is then passed to the modulator with order m to get the sequence of modulated symbols $\mathbf{x}_p = [x_u(1), x_u(2), \dots, x_u(N_c/m)]$. The signals of U users are transmitted to the receiver through the independent Rayleigh flat-fading channels, and each of the received symbols in sequence $\mathbf{r} =$

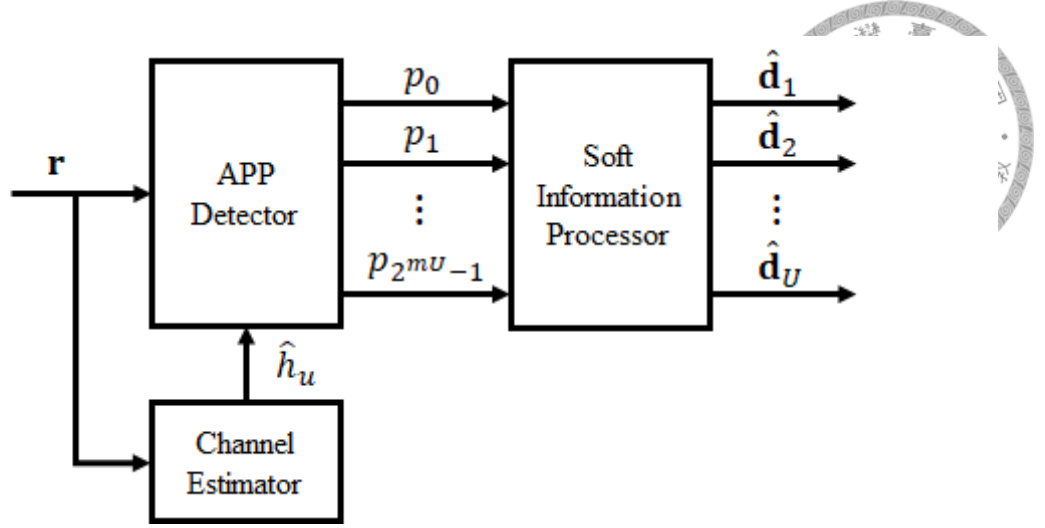


Figure 2.1: Transmitters of U users occupying the same resource.

$[r(1), r(2), \dots, r(N_c/m)]$ can be represented as

$$\begin{aligned}
 r(n) &= \sum_{u=1}^U h_u x_u(n) + w(n) \\
 &= s(n) + w(n),
 \end{aligned} \tag{2.1}$$

where h_u is the channel coefficient between user- u and the receiver, $w(n)$ is the complex additive white Gaussian noise (AWGN) and $s(n)$ is the superimposed symbol consisting of the signals transmitted from U users. There are 2^{mU} possible levels in each s , where we drop the symbol index n for brevity, and the levels are distinguishable due to the effect of independent fading channels.

The channel coefficient h_u for a Rician distributed channel can be represented as

$$h_u = \frac{1}{\sqrt{K+1}} \cdot a + \frac{\sqrt{K}}{\sqrt{K+1}} \cdot b \cdot i, \tag{2.2}$$

where a and b are normal distribution with variance 1, a is the real part coeffi-

cient and b is the imaginary part. h_u is the Rayleigh distribution for $K = 0$, and Rician distribution for $K \neq 0$.

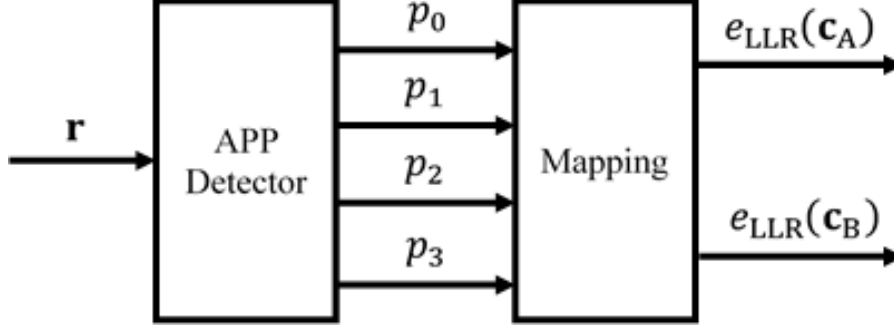


Figure 2.2: GDMA detector for BPSK and $U = 2$.

For the GDMA detector, we divided it into two steps to illustrate. Firstly, we obtained a 2^{mU} dimension probability vector of *a posteriori* probabilities (APPs) for all the superimposed signals. Secondly, Map the 2^{mU} dimension probability vector into U Log-likelihood Ratio(LLR). Fig. 2.2 is the block diagram of the GDMA detector in the case of $U = 2$ and BPSK modulation ($m = 1$), and the users are denoted as user-A and user-B. The mapping rules for code bits of two users and superimposed levels are tabulated in Table 2.1, and the possible levels of s are denoted as $S[l]$ with $l \in \{0, 1, 2, 3\}$. In Fig. 2.3, a graphical example of the superimposed levels and the corresponding binary mappings (c_A, c_B) is provided when h_A and h_B are the channel coefficients of user-A and user-B respectively. Note that an accurate estimation of channel coefficients for all users is necessary to recognize the superimposed signal levels properly. In this chapter, we assume that the channel state information (CSI) can be perfectly recovered at the receiver. Thus we can derive all the possible levels of received symbol s for further detection.

The *a posteriori* probabilities (APPs) that s is the l -th level with $l \in \{0, 1, 2, 3\}$

Table 2.1: Mapping rules for superimposed signal in the case of $m = 1$ and $U = 2$.

| l | c_A | c_B | x_A | x_B | $S[l]$ |
|-----|-------|-------|-------|-------|--------------|
| 0 | 0 | 0 | 1 | 1 | $h_A + h_B$ |
| 1 | 0 | 1 | 1 | -1 | $h_A - h_B$ |
| 2 | 1 | 0 | -1 | 1 | $-h_A + h_B$ |
| 3 | 1 | 1 | -1 | -1 | $-h_A - h_B$ |

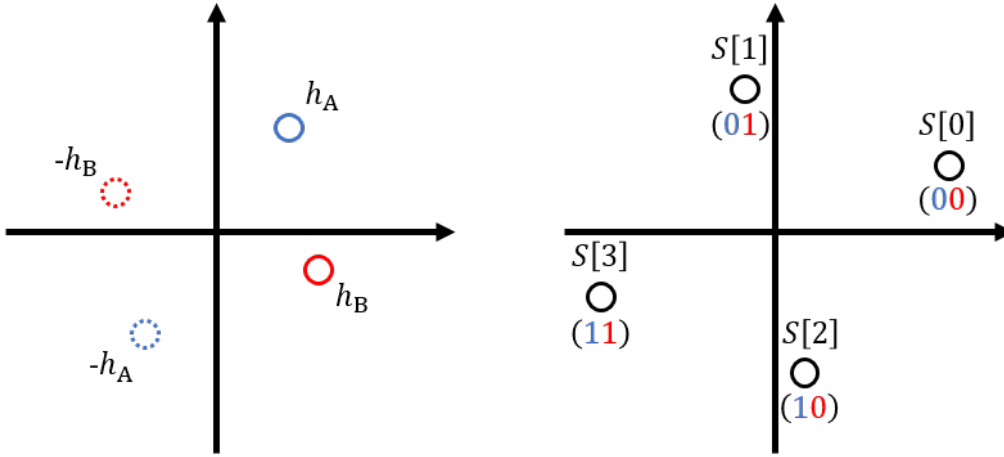


Figure 2.3: An example of superimposed levels in the case of $m = 1$ and $U = 2$.

given the received symbol r can be calculated by

$$\begin{aligned}
 p_l &= \Pr \{s = S[l] | r\} \\
 &= \Pr \{r | s = S[l]\} \cdot \frac{\Pr \{s = S[l]\}}{\Pr \{r\}},
 \end{aligned} \tag{2.3}$$

and

$$\Pr \{r | s = S[l]\} = \frac{1}{2\pi\sigma_w^2} \exp \left(-\frac{|r - S[l]|^2}{2\sigma_w^2} \right), \tag{2.4}$$

where $\Pr \{s = S[l]\}$ is the *a priori* probabilities of s and σ_w^2 is the variance of AWGN. According to the concept of physical layer network coding (PLNC), the

APPs of code bits given the received symbol r can be obtained as

$$\begin{aligned}\Pr\{c_A = 0|r\} &= \Pr\{s = S[0]|r\} + \Pr\{s = S[1]|r\} = p_0 + p_1, \\ \Pr\{c_A = 1|r\} &= \Pr\{s = S[2]|r\} + \Pr\{s = S[3]|r\} = p_2 + p_3,\end{aligned}\tag{2.5}$$

and

$$\begin{aligned}\Pr\{c_B = 0|r\} &= \Pr\{s = S[0]|r\} + \Pr\{s = S[2]|r\} = p_0 + p_2, \\ \Pr\{c_B = 1|r\} &= \Pr\{s = S[1]|r\} + \Pr\{s = S[3]|r\} = p_1 + p_3.\end{aligned}\tag{2.6}$$

The APPs of code bits can be derived for multiple users with the observation of received superimposed symbols as long as the CSI is available at the receiver. Once the APPs of code bits are obtained, the corresponding log-likelihood ratios (LLRs) of code bits are given by

$$\begin{aligned}e_{\text{LLR}}(c_A) &= \ln \frac{\Pr\{c_A = 0|r\}}{\Pr\{c_A = 1|r\}} = \ln \frac{p_0 + p_1}{p_2 + p_3}, \\ e_{\text{LLR}}(c_B) &= \ln \frac{\Pr\{c_B = 0|r\}}{\Pr\{c_B = 1|r\}} = \ln \frac{p_0 + p_2}{p_1 + p_3}.\end{aligned}\tag{2.7}$$

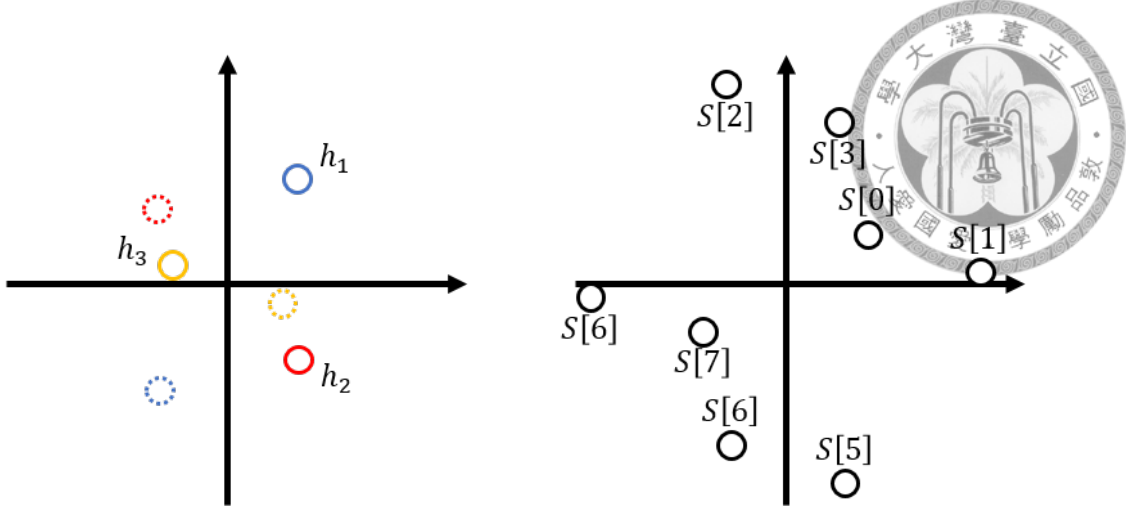


Figure 2.4: An example of superimposed levels in the case of $m = 1$ and $U = 3$.

Table 2.2: Mapping rules for superimposed signal in the case of $m = 1$ and $U = 3$.

| l | c_1 | c_2 | c_3 | x_1 | x_2 | x_3 | $S[l]$ |
|-----|-------|-------|-------|-------|-------|-------|--------------------|
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | $h_1 + h_2 + h_3$ |
| 1 | 0 | 0 | 1 | 1 | 1 | -1 | $h_1 + h_2 - h_3$ |
| 2 | 0 | 1 | 0 | 1 | -1 | 1 | $h_1 - h_2 + h_3$ |
| 3 | 0 | 1 | 1 | 1 | -1 | -1 | $h_1 - h_2 - h_3$ |
| 4 | 1 | 0 | 0 | -1 | 1 | 1 | $-h_1 + h_2 + h_3$ |
| 5 | 1 | 0 | 1 | -1 | 1 | -1 | $-h_1 + h_2 - h_3$ |
| 6 | 1 | 1 | 0 | -1 | -1 | 1 | $-h_1 - h_2 + h_3$ |
| 7 | 1 | 1 | 1 | -1 | -1 | -1 | $-h_1 - h_2 - h_3$ |

The multi-level detection technique in GDMA system can be easily extended to the system with $U > 2$ and/or $m > 1$. For example, in the case of $m = 1$ and $U = 3$, the mapping rules for code bits of users, denoted as user-1, user-2 and user-3, and superimposed levels are tabulated in Table 2.2. There are eight possible levels in each superimposed symbol s . Let p_l with $l \in \{0, 1, \dots, 7\}$ denote the APPs

of s given in (2.3), the LLRs of code bits are

$$\begin{aligned} e_{\text{LLR}}(c_1) &= \ln \frac{\Pr\{c_1 = 0|r\}}{\Pr\{c_1 = 1|r\}} = \ln \frac{p_0 + p_1 + p_2 + p_3}{p_4 + p_5 + p_6 + p_7}, \\ e_{\text{LLR}}(c_2) &= \ln \frac{\Pr\{c_2 = 0|r\}}{\Pr\{c_2 = 1|r\}} = \ln \frac{p_0 + p_1 + p_4 + p_5}{p_2 + p_3 + p_6 + p_7}, \\ e_{\text{LLR}}(c_3) &= \ln \frac{\Pr\{c_3 = 0|r\}}{\Pr\{c_3 = 1|r\}} = \ln \frac{p_0 + p_2 + p_4 + p_6}{p_1 + p_3 + p_5 + p_7}. \end{aligned} \quad (2.8)$$



2.2 Implementation Method

Although the concept of GDMA detection is straightforward, there is no general method to implement the GDMA detector [2] [4], which establishes a mapping table for a specific U . With the rise of U , the mapping table will grow exponentially, which will significantly improve the complexity of GDMA, and it is challenging to implement. Therefore, we propose a general method that can be implemented more simply and efficiently in this chapter.

2.2.1 BPSK-modulated GDMA Implementation

For general express for BPSK GDMA, we divide the detection into three steps: definition of levels, derivation of APPs, and derivation of LLRs. Firstly, the super-imposed signals $S[l], l \in 1, 2, \dots, L$ can be express as



$$\begin{aligned}
S[1] &= h_1 + h_2 + \cdots + h_U &= (-1)^0 \cdot h_1 + (-1)^0 \cdot h_2 + \cdots + (-1)^0 \cdot h_U, \\
S[2] &= h_1 + h_2 + \cdots - h_U &= (-1)^0 \cdot h_1 + (-1)^0 \cdot h_2 + \cdots + (-1)^1 \cdot h_U, \\
&\vdots \\
S[L] &= -h_1 - h_2 - \cdots - h_U &= (-1)^1 \cdot h_1 + (-1)^1 \cdot h_2 + \cdots + (-1)^1 \cdot h_U.
\end{aligned} \tag{2.9}$$

Convert the decimal l_{10} into binary l_2 , and define $l_2[u]$ is the u -th bit of binary l_2 . $l_{10} = l_2 = (l_2[1] \ l_2[2] \ \cdots \ l_2[U]), u = 1, 2, \cdots, U, l_{10} = 0, \cdots, L-1$. We consider that l_{10} is the serial number to record superimposed levels and consider that $l_2[u]$ is the code bit for user- u . Moreover, $l_2[u]$ is also a mapping index for u -user's channel gain h_u . We can derive mapping index from $S[l]$ directly by converting from decimal to binary. In the following expression, l_{10} and l_2 are used alternately.

$$\begin{aligned}
S[l_{10}] &= (-1)^{l_2[1]} \cdot h_1 + (-1)^{l_2[U]} \cdot h_2 + \cdots + (-1)^{l_2[U]} \cdot h_U \\
&= \sum_{u=1}^U (-1)^{l_2[u]} \cdot h_u, l_{10} = 0, 1, \cdots, L-1
\end{aligned} \tag{2.10}$$

where h_u is the complex channel coefficient of user- u . Secondly, APPs of l -th level with $l \in 0, 1, 2, \cdots, L$ given the received symbol r can be calculated as equation.2.3. Thirdly, once the APPs are obtained, the mapping formula can be expressed as



$$\begin{aligned}
 e_{\text{LLR}}[c_u] &= \ln \frac{\Pr\{c_u = 0|r\}}{\Pr\{c_u = 1|r\}} \\
 &= \ln \frac{\sum_{l_{10}=0, l_2[u] \in 0}^L \Pr\{s = S[l]|r\}}{\sum_{l_{10}=0, l_2[u] \in 1}^L \Pr\{s = S[l]|r\}}
 \end{aligned} \tag{2.11}$$

The two equations above and below the fraction are the probability that user- u code bits are equal to zero or one, and there are $\frac{L}{2}$ APPs added respectively. The APPs of the upper fraction must meet h_u superimposition, and the APPs of the lower fraction must meet $-h_u$ superimposition. According to 2.10, we can use mapping index $l_2[u]$ equal to zero or one to judge whether meet the u -user

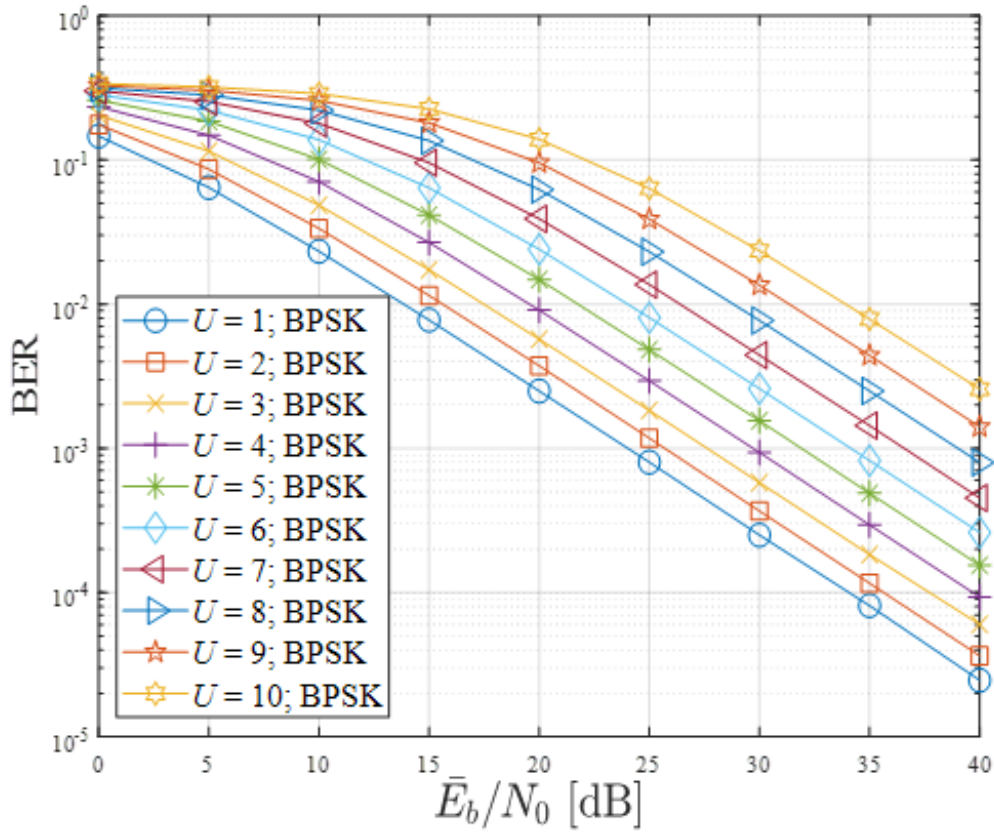


Figure 2.5: BER performances of uncoded GDMA BPSK system over block Rayleigh fading channels.

restriction.



2.2.2 QPSK-modulated GDMA Implementation

In addition to BPSK, we also discuss the general detection of QPSK with modulation order ($m = 2$). According to QPSK modulation, the superimposed signals $S[l], l \in 1, 2, \dots, L$ can be expressed as

$$\begin{aligned} S[0] &= e^{\frac{\pi}{4}j}h_1 + e^{\frac{\pi}{4}j}h_2 + \dots + e^{\frac{\pi}{4}j}h_U, \\ S[1] &= e^{\frac{\pi}{4}j}h_1 + e^{\frac{\pi}{4}j}h_2 + \dots + e^{(\frac{\pi}{4}+\frac{\pi}{2})j}h_U, \\ S[2] &= e^{\frac{\pi}{4}j}h_1 + e^{\frac{\pi}{4}j}h_2 + \dots + e^{(\frac{\pi}{4}+\pi)j}h_U, \\ S[3] &= e^{\frac{\pi}{4}j}h_1 + e^{\frac{\pi}{4}j}h_2 + \dots + e^{(\frac{\pi}{4}+\frac{3\pi}{2})j}h_U, \\ &\vdots \end{aligned}$$

$$S[L-1] = e^{(\frac{\pi}{4}+\frac{3\pi}{2})j}h_1 + e^{(\frac{\pi}{4}+\frac{3\pi}{2})j}h_2 + \dots + e^{(\frac{\pi}{4}+\frac{3\pi}{2})j}h_U. \quad (2.12)$$

Different from BPSK detection, we convert the decimal l_{10} into quaternary l_4 since the modulation order ($m = 2$), and define that $l_4[u]$ is the u -th bit of quaternary l_4 . $l_{10} = l_4 = (l_4[1] \ l_4[2] \ \dots \ l_4[U]), u = 1, 2, \dots, U$. We consider that $l_4[u]$ is the code symbol for user- u , which might be 00, 01, 10, 11. Moreover, $l_4[u]$ is also a mapping index for u -user's channel gain h_u . After simplifying 2.12, we can obtain superimposed levels as



$$\begin{aligned}
S[l] &= e^{(\frac{\pi}{4} + \frac{\pi}{2} \cdot l_4[1])j} \cdot h_1 + e^{(\frac{\pi}{4} + \frac{\pi}{2} \cdot l_4[1])j} \cdot h_2 + \dots + e^{(\frac{\pi}{4} + \frac{\pi}{2} \cdot l_4[1])j} \cdot h_U \\
&= \sum_{u=1}^U e^{(\frac{\pi}{4} + \frac{\pi}{2} \cdot l_4[u])j} \cdot h_u, l_{10} = 0, 1, \dots, L-1; u = 1, 2, \dots, U \quad (2.13)
\end{aligned}$$

In the above definition 2.13, we can define levels architecturally, and then need to derive the APP of each level. The APPs can be calculated as 2.3. Finally, e_{LLR} can be easily obtained with the structure of $S[l]$.

$$e_{\text{LLR}}[c_u] = \ln \frac{\Pr\{c_u = 0|r\}}{\Pr\{c_u = 1|r\}} = \begin{cases} \ln \frac{\sum_{l_{10}=0, l_2[u] \in \{0,1\}}^L \Pr\{s=S[l]|r\}}{\sum_{l_{10}=0, l_2[u] \in \{2,3\}}^L \Pr\{s=S[l]|r\}}, i = 1; u = 1, \dots, U \\ \ln \frac{\sum_{l_{10}=0, l_2[u] \in \{0,2\}}^L \Pr\{s=S[l]|r\}}{\sum_{l_{10}=0, l_2[u] \in \{1,3\}}^L \Pr\{s=S[l]|r\}}, i = 2; u = 1, \dots, U \end{cases}$$

The Fig.2.6 is the QPSK constellation according to gray mapping. We need to consider the code bit code bits corresponding to the first or the second bit on constellation. For first bit $i = 1$, both level- $S[0]$ and level- $S[1]$ are mapping to code bit equal to 0, while level- $S[2]$ and level- $S[3]$ are mapping to code bits equal to 1, where $i = 1, \dots, m$ is the i -bit for constellation. We also can use mapping index $l_4[u]$ to extract the phase information from $S[l]$. Map LLR according to i -constellation location and u -user.

Fig.2.7 is the BER performance of uncoded GDMA QPSK over a block rayleigh fading channel.

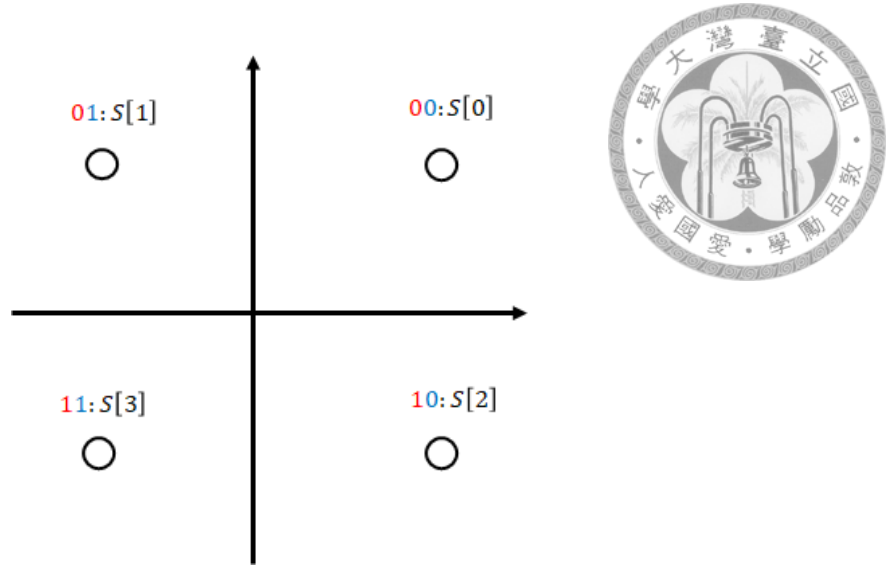


Figure 2.6: QPSK modulation gray mapping.

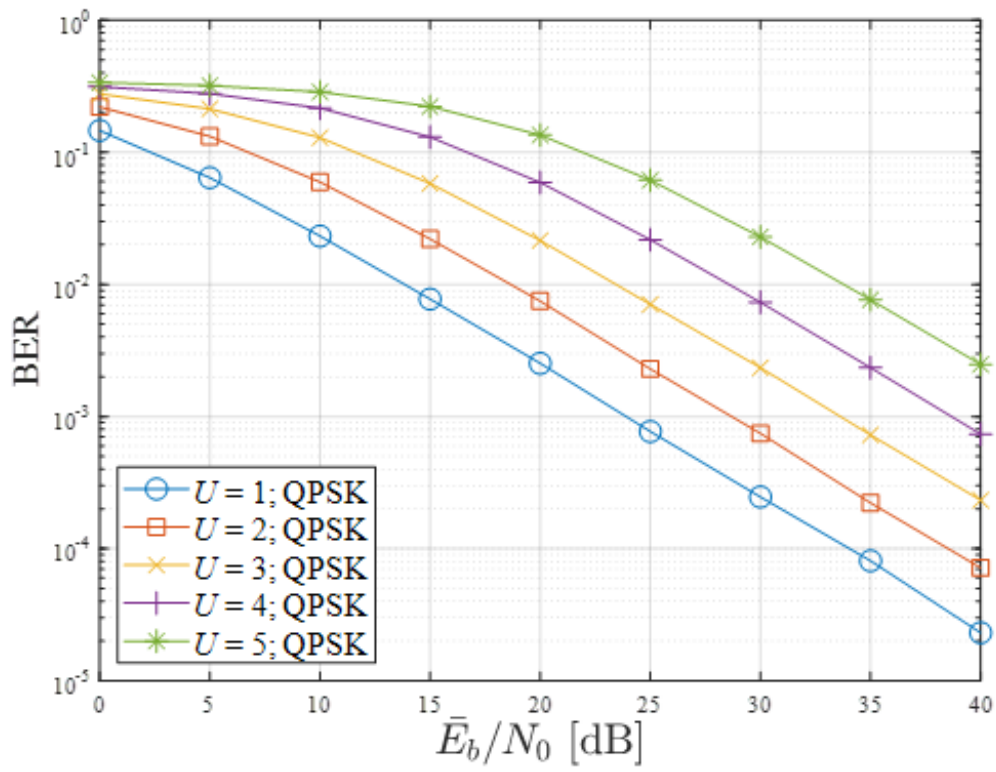


Figure 2.7: BER performances of uncoded GDMA QPSK system over block Rayleigh fading channels.

2.2.3 PAM-modulated GDMA Implementation



Consider a general PAM case: 2^m -PAM; U is integer; m is integer; E_b is the average power; $M = 2^m - 1$ is the mid number.

The superimposed signals $S[l], l \in 1, 2, \dots, L$ can be express as

$$\begin{aligned}
 S[0] &= (2 \cdot 0 - M) \cdot \sqrt{E_s} \cdot h_1 + (2 \cdot 0 - M) \cdot \sqrt{E_s} \cdot h_2 + \dots + (2 \cdot 0 - M) \cdot \sqrt{E_s} \cdot h_U, \\
 S[1] &= (2 \cdot 0 - M) \cdot \sqrt{E_s} \cdot h_1 + (2 \cdot 0 - M) \cdot \sqrt{E_s} \cdot h_2 + \dots + (2 \cdot 1 - M) \cdot \sqrt{E_s} \cdot h_U, \\
 S[2] &= (2 \cdot 0 - M) \cdot \sqrt{E_s} \cdot h_1 + (2 \cdot 0 - M) \cdot \sqrt{E_s} \cdot h_2 + \dots + (2 \cdot 2 - M) \cdot \sqrt{E_s} \cdot h_U, \\
 S[3] &= (2 \cdot 0 - M) \cdot \sqrt{E_s} \cdot h_1 + (2 \cdot 0 - M) \cdot \sqrt{E_s} \cdot h_2 + \dots + (2 \cdot 3 - M) \cdot \sqrt{E_s} \cdot h_U, \\
 &\vdots \\
 S[L-1] &= (2 \cdot M - M) \cdot \sqrt{E_s} \cdot h_1 + (2 \cdot M - M) \cdot \sqrt{E_s} \cdot h_2 + \dots + (2 \cdot M - M) \cdot \sqrt{E_s} \cdot h_U.
 \end{aligned} \tag{2.14}$$

We can convert the decimal l_{10} into quaternary l_M , and define that $l_M[u]$ is the u -th bit of l_M . $l_{10} = l_M = (l_M[1] \ l_M[2] \ \dots \ l_M[U]), u = 1, 2, U$. After simplifying 2.14, we can obtain superimposed levels as

$$\begin{aligned}
 S[l] &= (2 \cdot l_{2^m}[1] - M) \cdot \sqrt{E_v} \cdot h_1 + (2 \cdot l_{2^m}[2] - M) \cdot \sqrt{E_v} \cdot h_2 + \dots + (2 \cdot l_{2^m}[U] - M) \cdot \sqrt{E_v} \cdot h_U \\
 &= \sum_{u=1}^U (2 \cdot l_{2^m}[u] - M) \cdot \sqrt{E_v} \cdot h_u, u = 1, 2, \dots, U
 \end{aligned} \tag{2.15}$$

In the above definition, we can define levels architecturally and then need to derive the APP of each level. The APPs can be calculated as 2.3. Finally, e_{LLR} can

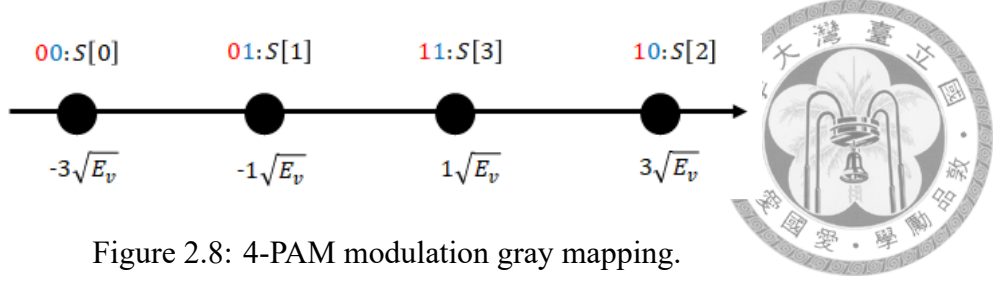


Figure 2.8: 4-PAM modulation gray mapping.

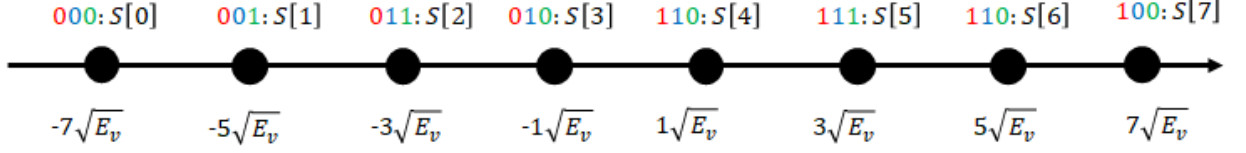


Figure 2.9: 8-PAM modulation gray mapping.

be easily obtained with the structure of $S[l]$. According to m, there will be different gray mapping for PAM. Firstly, we discuss the 4-PAM and the gray mapping in Fig.2.8.

$$e_{\text{LLR}}[c_u] = \ln \frac{\Pr\{c_u = 0|r\}}{\Pr\{c_u = 1|r\}} = \begin{cases} \ln \frac{\sum_{l_{10}=0, l_4[u] \in \{0,1\}} \Pr\{s=S[l]|r\}}{\sum_{l_{10}=0, l_4[u] \in \{2,3\}} \Pr\{s=S[l]|r\}}, i = 1; u = 1, \dots, U \\ \ln \frac{\sum_{l_{10}=0, l_4[u] \in \{0,3\}} \Pr\{s=S[l]|r\}}{\sum_{l_{10}=0, l_4[u] \in \{1,2\}} \Pr\{s=S[l]|r\}}, i = 2; u = 1, \dots, U \end{cases}$$

Secondly, we discuss the 8-PAM and the gray mapping 2.9.

$$e_{\text{LLR}}[c_u] = \ln \frac{\Pr\{c_u = 0|r\}}{\Pr\{c_u = 1|r\}} = \begin{cases} \ln \frac{\sum_{l_{10}=0, l_8[u] \in \{0,1,2,3\}} \Pr\{s=S[l]|r\}}{\sum_{l_{10}=0, l_8[u] \in \{4,5,6,7\}} \Pr\{s=S[l]|r\}}, i = 1; u = 1, \dots, U \\ \ln \frac{\sum_{l_{10}=0, l_8[u] \in \{0,1,6,7\}} \Pr\{s=S[l]|r\}}{\sum_{l_{10}=0, l_8[u] \in \{2,3,4,5\}} \Pr\{s=S[l]|r\}}, i = 2; u = 1, \dots, U \\ \ln \frac{\sum_{l_{10}=0, l_8[u] \in \{0,3,4,7\}} \Pr\{s=S[l]|r\}}{\sum_{l_{10}=0, l_8[u] \in \{1,2,5,6\}} \Pr\{s=S[l]|r\}}, i = 2; u = 1, \dots, U \end{cases}$$

Fig.2.10 and fig.2.11 are the BER performance respectively for uncoded GDMA 4-pam and 8-pam.

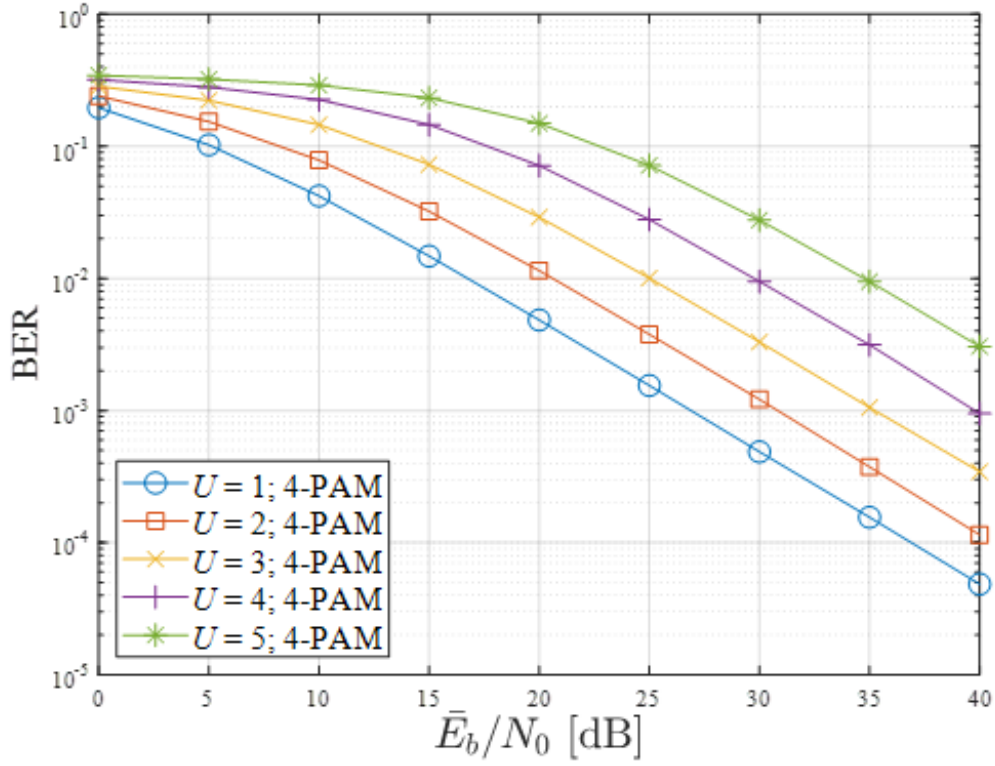


Figure 2.10: BER performances of uncoded GDMA 4-PAM system over block Rayleigh fading channels.

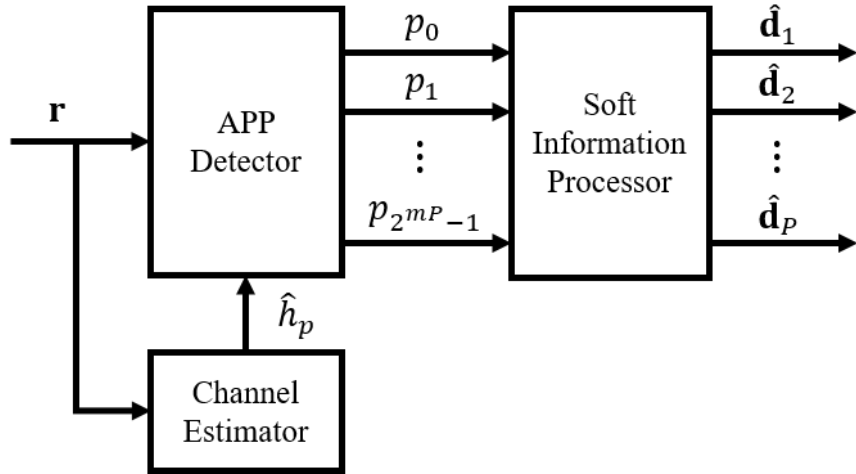


Figure 2.12: Multi-level receiver in GDMA system for recovering the messages of U users.

The LLRs can be directly used for data detection or be further fed to the following soft information processor (SIP) when FEC coding is considered. The block

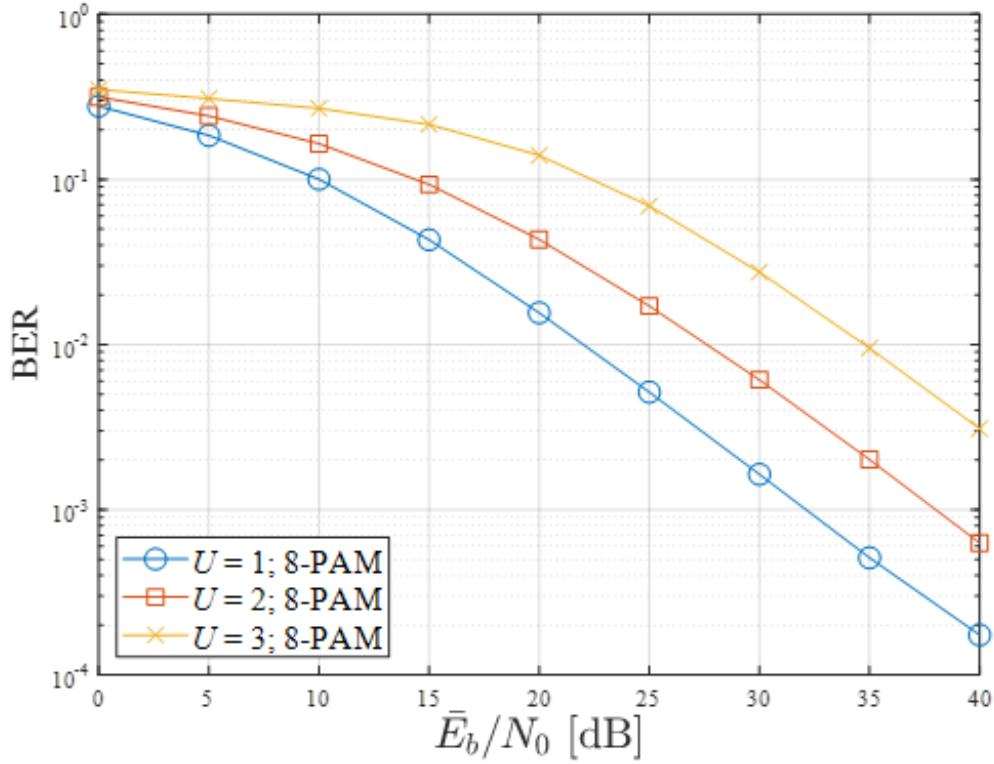


Figure 2.11: BER performances of uncoded GDMA 8-PAM system over block Rayleigh fading channels.

diagram of the multiuser detector in GDMA system is shown in Fig. 2.12. The SIP can be implemented by using several decoders (DECs) and one for each user independently as shown in Fig. 2.13. A joint channel decoder can be an alternative to implement the SIP and extract the messages of all users simultaneously. The implementation of SIP will be discussed in Sec. 2.3. If the messages are transmitted without FEC coding, the decision rule with the APPs derived from multi-level detection is equivalent to finding the nearest neighbor of received symbol among all the levels. The corresponding decision region depends on the channel coefficients.

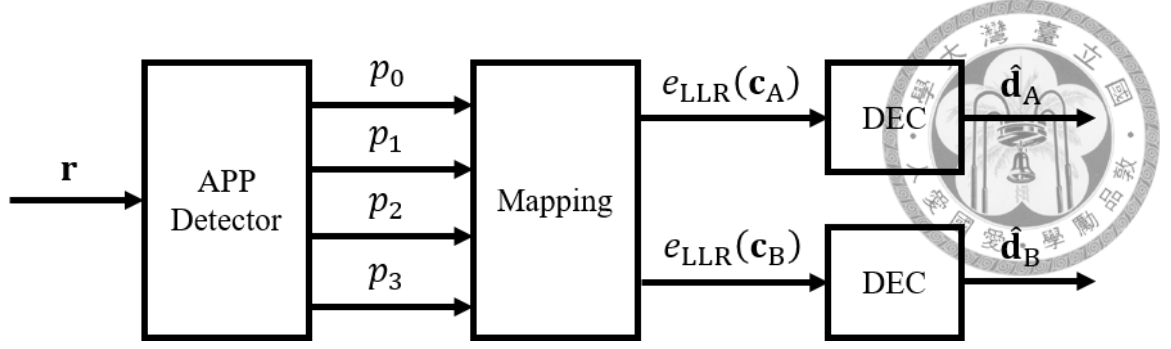


Figure 2.13: Separated channel decoding in the case of $m = 1$ and $U = 2$.

2.3 Joint channel decoding for the GDMA system

In this section, the implementation flow of the soft information processor (SIP) in the multiuser detector is discussed. We firstly review the joint channel decoding for the LDPC coded GDMA system and secondly discuss a novel joint channel decoding for the polar coded system.

2.3.1 Introduction of Joint Channel Decoding

A straightforward method to construct the SIP in GDMA system is called separated channel decoding (SCD) as shown in fig. 2.13. To recover the messages of U users in GDMA-SCD system, the APPs of code bits are first attained and then be fed to the following decoders independently for each user. Note that there is no exchange of informations between collided users in GDMA-SCD scheme.

Another method of FEC-GDMA system is called Joint Channel Decoding (JCD) as shown in 2.14, an alternative to implement the SIP in the case of $U = 2$ and BPSK transmission. In the GDMA-JCD system, The APPs of codeword will pass the message to the JCD decoder before mapping LLRs. In the JCD, L APP vectors

the symbol $s(n)$ with $n \in \{1, 2, \dots, N_c\}$ over the Tanner graph [9]. The four possible levels in each $s(n)$ are shown in Table 2.1. The message passed through the edges in the corresponding Tanner graph are represented by probability vector $\mathbf{p} = [p_0 \ p_1 \ p_2 \ p_3]$ where p_i is the probability that the value of variable $s(n)$ is $S[l]$ with $l \in \{0, 1, 2, 3\}$, and $p_0 + p_1 + p_2 + p_3 = 1$ holds. The initial message of variable node $s(n)$ given the received symbol $r(n)$ is $p_l = \Pr\{s(n) = S[l]|r\}$ with $l \in \{0, 1, 2, 3\}$ given in (2.3).

The message updating rules at variable nodes and check nodes within the G-SPA are the same as that discussed in [9], and the update functions are defined as VAR and CHK for variable nodes and check nodes respectively. The discussion is restricted to nodes of degree three and the messages from the nodes with degree greater than three can be calculated by

$$\begin{aligned} \text{VAR}(\mathbf{p}, \mathbf{q}, \dots) &= \text{VAR}(\mathbf{p}, \text{VAR}(\mathbf{q}, \text{VAR}(\cdot, \cdot))), \\ \text{CHK}(\mathbf{p}, \mathbf{q}, \dots) &= \text{CHK}(\mathbf{p}, \text{CHK}(\mathbf{q}, \text{CHK}(\cdot, \cdot))), \end{aligned} \quad (2.16)$$

where \mathbf{p} and \mathbf{q} denote the corresponding input messages derived from variable nodes or check nodes. The functions of VAR and CHK at the nodes of degree three are illustrated in Fig. 2.15, and the circles and squares represent the variable nodes and check nodes respectively.

Output Message of Variable Nodes When the two input messages $\mathbf{p} = [p_0 \ p_1 \ p_2 \ p_3]$ and $\mathbf{q} = [q_0 \ q_1 \ q_2 \ q_3]$ arrive at the variable node $s(n)$, the probability that $s(n) =$

$S[l]$ with $l \in \{0, 1, 2, 3\}$ is given by

$$\begin{aligned}
 \Pr\{s(n) = S[l] | \mathbf{p}, \mathbf{q}\} &= \frac{\Pr\{\mathbf{p}, \mathbf{q} | s(n) = S[l]\} \Pr\{s(n) = S[l]\}}{\Pr\{\mathbf{p}, \mathbf{q}\}} \\
 &= \frac{\Pr\{s(n) = S[l] | \mathbf{p}\} \Pr\{s(n) = S[l] | \mathbf{q}\} \Pr\{\mathbf{p}\} \Pr\{\mathbf{q}\}}{\Pr\{\mathbf{p}, \mathbf{q}\}} \\
 &= \beta p_l q_l,
 \end{aligned} \tag{2.17}$$

where

$$\beta = \frac{\Pr\{\mathbf{p}\} \Pr\{\mathbf{q}\}}{\Pr\{\mathbf{p}, \mathbf{q}\}} = \frac{1}{p_0 q_0 + p_1 q_1 + p_2 q_2 + p_3 q_3}, \tag{2.18}$$

is a normalization factor. Therefore, the output message of variable node $s(n)$ is

$$\text{VAR}(\mathbf{p}, \mathbf{q}) = \beta [p_0 q_0 \ p_1 q_1 \ p_2 q_2 \ p_3 q_3]. \tag{2.19}$$

Output Message of Check Nodes A specific parity-check equation is satisfied if the \mathbb{F}_4 sum of the connected quaternary symbols in the corresponding Tanner graph is equal to zero, i.e., $s(n) \oplus s(n_1) \oplus s(n_2) = 0$. Assuming that the two input messages derived from the variable nodes $s(n_1)$ and $s(n_2)$ are $\mathbf{p} = [p_0 \ p_1 \ p_2 \ p_3]$

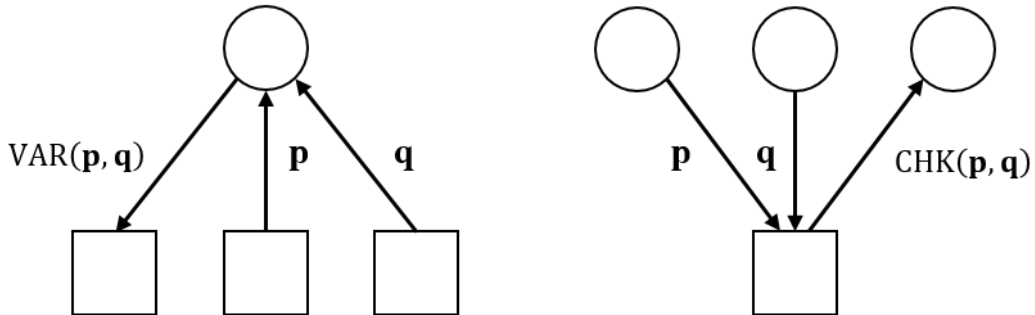


Figure 2.15: The message passing algorithm along the edges in factor graph.

and $\mathbf{q} = [q_0 \ q_1 \ q_2 \ q_3]$, respectively. The probability that the parity-check equation is satisfied under the the assumption that $s(n) = S[0]$ is



$$\begin{aligned}
 \Pr\{s(n) = S[0]|\mathbf{p}, \mathbf{q}\} &= \Pr\{s(n_1) = S[0], s(n_2) = S[0]|\mathbf{p}, \mathbf{q}\} \\
 &+ \Pr\{s(n_1) = S[1], s(n_2) = S[1]|\mathbf{p}, \mathbf{q}\} \\
 &+ \Pr\{s(n_1) = S[2], s(n_2) = S[2]|\mathbf{p}, \mathbf{q}\} \\
 &+ \Pr\{s(n_1) = S[3], s(n_2) = S[3]|\mathbf{p}, \mathbf{q}\} \\
 &= p_0q_0 + p_1q_1 + p_2q_2 + p_3q_3.
 \end{aligned} \tag{2.20}$$

Similarly we can have

$$\begin{aligned}
 \Pr\{s(n) = S[1]|\mathbf{p}, \mathbf{q}\} &= p_0q_1 + p_1q_0 + p_2q_3 + p_3q_2, \\
 \Pr\{s(n) = S[2]|\mathbf{p}, \mathbf{q}\} &= p_0q_2 + p_2q_0 + p_1q_3 + p_3q_1, \\
 \Pr\{s(n) = S[3]|\mathbf{p}, \mathbf{q}\} &= p_0q_3 + p_3q_0 + p_1q_2 + p_2q_1.
 \end{aligned} \tag{2.21}$$

Finally, the message out of one check node equals

$$\text{CHK}(\mathbf{p}, \mathbf{q}) = \begin{bmatrix} p_0q_0 + p_1q_1 + p_2q_2 + p_3q_3 \\ p_0q_1 + p_1q_0 + p_2q_3 + p_3q_2 \\ p_0q_2 + p_2q_0 + p_1q_3 + p_3q_1 \\ p_0q_3 + p_3q_0 + p_1q_2 + p_2q_1 \end{bmatrix}^T. \tag{2.22}$$

Finalization and Symbol-to-Bit Mapping The decoding algorithm is stopped if all parity-check equations are fulfilled or the maximum number of iterations is

reached. Otherwise, the process proceed with further iteration until one of the conditions is satisfied. At the end of the decoding algorithm, the G-SPA generates the APP vector $\mathbf{p} = [p_0 \ p_1 \ p_2 \ p_3]$ with $p_i = \Pr\{s(n) = S[l]|r\}$ for each symbol $s(n)$ with $n \in \{1, 2, \dots, N_c\}$ and the symbol-to-bit mapping is done by (2.7). The combination of multi-level detection and G-SPA is referred to as GDMA-JCD scheme in this thesis. The G-SPA with higher-order modulation scheme was investigated in [10].

Simulation Results The BLER performances of GDMA-JCD and GDMA-SCD systems using a (1008, 504) rate-1/2 binary (3,6)-regular LDPC code with BPSK transmissions over quasi-static Rayleigh flat-fading channels are provided in Fig. 2.16. The maximum number of decoding iterations is set to 50 for both SPA and G-SPA. In the transmissions over block flat-fading channels, the GDMA-JCD scheme outperforms the GDMA-SCD scheme in low-SNR region and the performances of two schemes converge to each other in high-SNR region. We further consider an OFDM-GDMA system that has 16 sub-carrier and 63 bits for each sub-carrier symbol. The BLER performance of GDMA-OFDM-JCD and GDMA-OFDM-SCD using same parameters as above and are provided in Fig. 2.17. We found that JCD can effectively use different user information, even when increasing the number of users, performance has almost no loss.

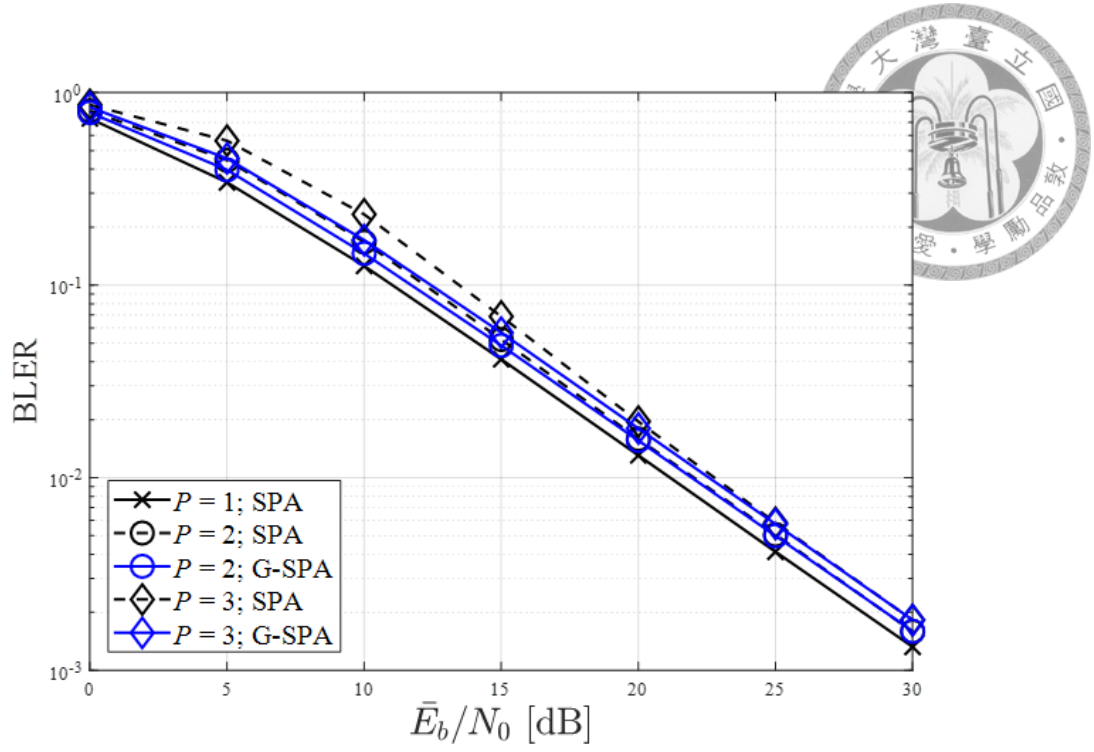


Figure 2.16: BLER performances of (1008, 504) LDPC-coded GDMA systems over quasi-static Rayleigh flat-fading channels.

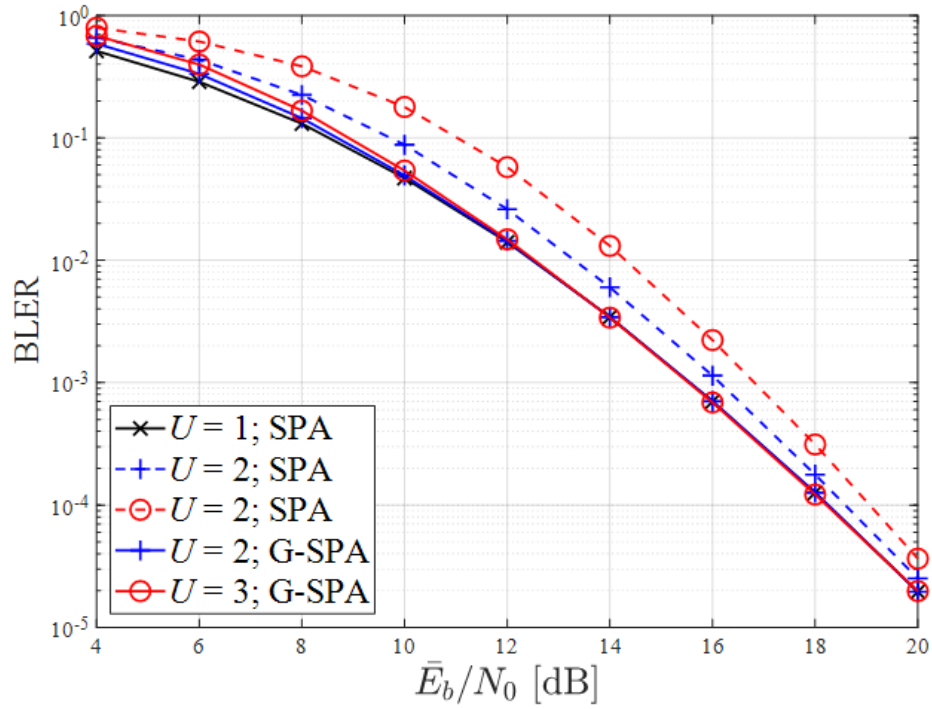


Figure 2.17: BLER performances of (1008, 504) LDPC-coded GDMA OFDM systems over quasi-static Rayleigh flat-fading channels with 16-subcarrier.

2.3.3 Polar Coded GDMA system



Polar code [11] invented by Erdal Arikan in 2006 can achieve symmetric capacity on binary-input discrete memoryless channels. In 5G standardization, polar code has been adopted as a channel coding scheme for 3GPP new radio(NR) access technology. By exploiting channel polarization, one can construct excellent polar codes, and employ low-complexity successive cancellation(SC), successive cancellation list (SCL) [12], and sum-product Algorithm(SPA) [13] for polar codes. In the thesis [4], the polar coded GDMA system was implemented by separated channel decoding (SCD) with SCL decoder. Since the hard decoding structure in SCL decoder, it can not pass APP vectors directly. We need soft value to map the codeword LLR $e_{LLR(n,u)}$ with $n \in \{1, 2, \dots, N_c\}; u \in \{1, \dots, U\}$ for each user. In this section, we firstly introduce a general sum-product algorithm (G-SPA) decoder and secondly a joint successive cancellation list (J-SCL) decoder for polar coded JCD-GDMA system.

Graph representation The graph representation Fig.2.18 for the transformation was noticed by Forney who suggested a SPA decoder for RM codes using factor graph representation. Since Polar code is the family of RM code, [13] use the graph of polar code structure to do SPA decoding. There are four nodes in a base computation block (BCB). Instead of passing LLR value, the JCD passes the probability vector of each level in the factor graph. We define the vectors passing to the right as $\mathbf{R}_{i,j}$ and the vectors passing to the left as $\mathbf{L}_{i,j}$ where $1 \leq i \leq n + 1, 1 \leq j \leq N$, N : codeword length, n : factor graph layer. The nodes of factor graph are

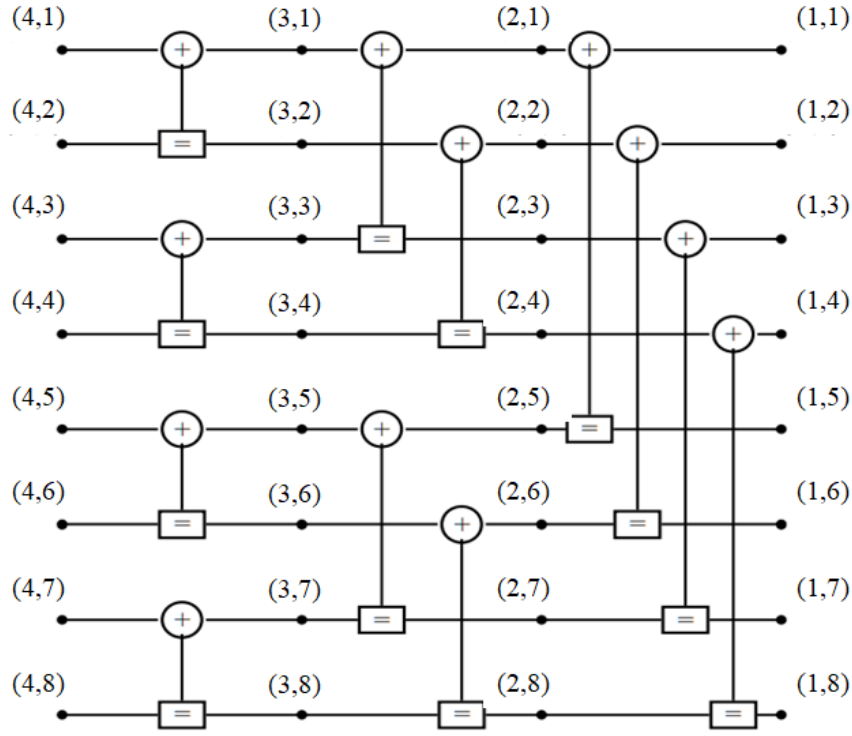


Figure 2.18: The graph factor for polar code, $N = 8, n = 3$.

labelled with pairs of integers (i, j) . From the decoder's perspective, the leftmost nodes, $(n + 1, j)$ are associated with the source vector $\mathbf{u} = [u_0 \ u_1 \ u_2 \ \cdots \ u_L]$ where u_j is the message information of each user, while the rightmost nodes $(1, j)$ are associated with channel input vectors $\mathbf{p} = [p_0 \ p_1 \ p_2 \ \cdots \ p_L]$ where p_l is the probability that are observed through a noisy channel. Instead of min sum operations, we define the BCB upper branch calculations as f -function and lower branch calculations as g -function such as Fig.2.19. For clarity, we predigest g -function a vector product calculation, f -function a vector parity check calculation. The function definition in the case of $U = 2$ and BPSK transmission is described as follows. Fig.2.18 is the graph representation example for $N = 8, n = 3$, where BCB in fig.2.19 is the unit of graph. g -function and f -function are the operations same as generalized joint channel decoding and physical-layer network coding (G-JCNC) in [5].

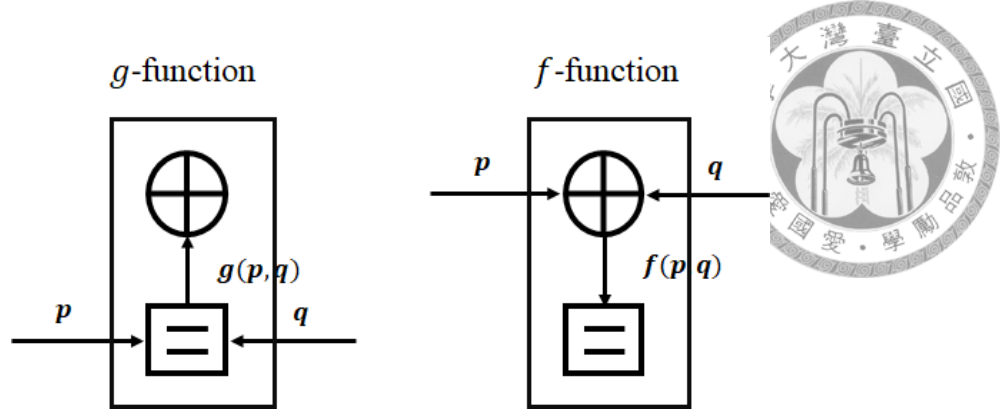


Figure 2.19: The message passing function in factor graph.

f -function When the two input messages vectors $\mathbf{p} = [p_0 \ p_1 \ p_2 \ p_3]$ and $\mathbf{q} = [q_0 \ q_1 \ q_2 \ q_3]$ arrive at the BCB upper branches $s(n)$, the probability that $s(n) = S[l]$ with $l \in \{0, 1, 2, 3\}$ is given by

$$\begin{aligned} \Pr\{s(n) = S[l] | \mathbf{p}, \mathbf{q}\} &= \frac{\Pr\{\mathbf{p}, \mathbf{q} | s(n) = S[l]\} \Pr\{s(n) = S[l]\}}{\Pr\{\mathbf{p}, \mathbf{q}\}} \\ &= \frac{\Pr\{s(n) = S[l] | \mathbf{p}\} \Pr\{s(n) = S[l] | \mathbf{q}\} \Pr\{\mathbf{p}\} \Pr\{\mathbf{q}\}}{\Pr\{\mathbf{p}, \mathbf{q}\}} \\ &= \beta p_l q_l, \end{aligned} \quad (2.23)$$

where

$$\beta = \frac{\Pr\{\mathbf{p}\} \Pr\{\mathbf{q}\}}{\Pr\{\mathbf{p}, \mathbf{q}\}} = \frac{1}{p_0 q_0 + p_1 q_1 + p_2 q_2 + p_3 q_3}, \quad (2.24)$$

is a normalization factor. Therefore, the output message is

$$f(\mathbf{p}, \mathbf{q}) = \beta [p_0 q_0 \ p_1 q_1 \ p_2 q_2 \ p_3 q_3]. \quad (2.25)$$

g -function A specific parity-check equation is satisfied if the \mathbb{F}_4 sum of the connected quaternary symbols in the corresponding Tanner graph is equal to zero, i.e.,

$s(n) \oplus s(n_1) \oplus s(n_2) = 0$. Assuming that the two input messages derived from the variable nodes $s(n_1)$ and $s(n_2)$ are $\mathbf{p} = [p_0 \ p_1 \ p_2 \ p_3]$ and $\mathbf{q} = [q_0 \ q_1 \ q_2 \ q_3]$, respectively. The probability that the parity-check equation is satisfied under the assumption that $s(n) = S[0]$ is

$$\begin{aligned}
 \Pr\{s(n) = S[0]|\mathbf{p}, \mathbf{q}\} &= \Pr\{s(n_1) = S[0], s(n_2) = S[0]|\mathbf{p}, \mathbf{q}\} \\
 &\quad + \Pr\{s(n_1) = S[1], s(n_2) = S[1]|\mathbf{p}, \mathbf{q}\} \\
 &\quad + \Pr\{s(n_1) = S[2], s(n_2) = S[2]|\mathbf{p}, \mathbf{q}\} \\
 &\quad + \Pr\{s(n_1) = S[3], s(n_2) = S[3]|\mathbf{p}, \mathbf{q}\} \\
 &= p_0q_0 + p_1q_1 + p_2q_2 + p_3q_3.
 \end{aligned} \tag{2.26}$$

Similarly we can have

$$\begin{aligned}
 \Pr\{s(n) = S[1]|\mathbf{p}, \mathbf{q}\} &= p_0q_1 + p_1q_0 + p_2q_3 + p_3q_2, \\
 \Pr\{s(n) = S[2]|\mathbf{p}, \mathbf{q}\} &= p_0q_2 + p_2q_0 + p_1q_3 + p_3q_1, \\
 \Pr\{s(n) = S[3]|\mathbf{p}, \mathbf{q}\} &= p_0q_3 + p_3q_0 + p_1q_2 + p_2q_1.
 \end{aligned} \tag{2.27}$$

Finally, the message out of one check node equals

$$\text{CHK}(\mathbf{p}, \mathbf{q}) = \begin{bmatrix} p_0q_0 + p_1q_1 + p_2q_2 + p_3q_3 \\ p_0q_1 + p_1q_0 + p_2q_3 + p_3q_2 \\ p_0q_2 + p_2q_0 + p_1q_3 + p_3q_1 \\ p_0q_3 + p_3q_0 + p_1q_2 + p_2q_1 \end{bmatrix}^T. \tag{2.28}$$

General Sum Product Algorithm decoder



In addition to SCL decoder, SPA decoder can also be used to decode polar code. We can replace this SPA decoder with the G-SPA decoder, e.g., passing APP vector in factor graph by f and g functions. It can be easily converted separated channel decoding into joint channel decoding GDMA (JCD-GDMA) system.

Initialization In G-SPA, we changed the LLR representation to vector representation, and before message passing, we need to set the initial vectors. For simplicity, we set the initial vectors on the most left as the initial APPs of the received superimposed signal. The rightest vectors are determined according to the frozen bits of the polar code, and the other vectors are the message.

The rightmost nodes $(1, j)$ are associated with channel input vectors that are observed through a noisy channel, $1 \leq i \leq n + 1, 1 \leq j \leq N$ where N is the codeword length.

$$L_{1,j} = [p_0 \ p_1 \ \cdots \ p_L]. \quad p_l = \Pr\{s = S[l]|r\} \quad (2.29)$$

The leftmost nodes, $(n + 1, j)$, are associated with the source data that are to be estimated. If u_j is a frozen bit, we can determine that the leftmost $R_{n+1,j}$ is equal to the superimposed signal $S[0]$, so it can be represented as

$$R_{n+1,j} = \begin{cases} [1 \ 0 \ \cdots \ 0], & \text{if is a frozen coordinate} \\ [\frac{1}{L} \ \frac{1}{L} \ \cdots \ \frac{1}{L}], & \text{if is an infrozen coordinate} \end{cases}$$

All the other nodes $R_{i,j}$ and $L_{i,j}$ are setted equal to $[\frac{1}{L} \ \frac{1}{L} \ \cdots \ \frac{1}{L}]$

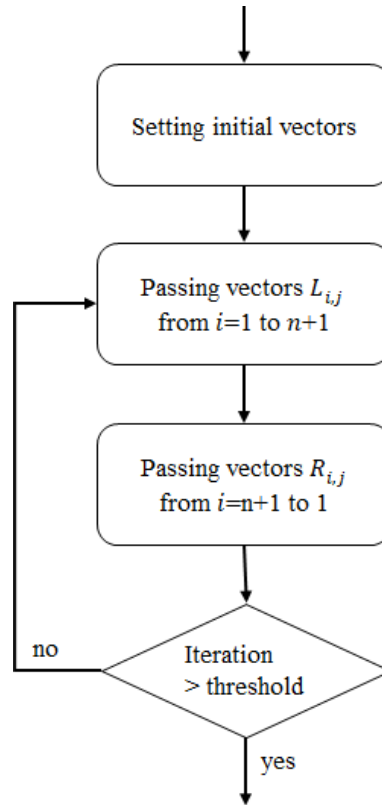


Figure 2.20: The flow chart of polar code G-SPA decoder.

Message passing After setting initial vectors of the factor graph, the message passing procedure is as fig.2.20. Unlike the SCL decoder, all operations in the SPA decoder are parallel. The operations will start from the rightmost nodes, then pass the message vectors to the left in sequence. Until updating to the leftmost node, and then act to the right nodes. When it updates to the rightmost nodes again, it finishes an iteration. The BCB calculation is described as



$$\begin{aligned}
L_{i+1, j} &= f\{L_{i, j+N_i}, g\{L_{i, j}, R_{i, j}\}\} \\
L_{i+1, j+N_i} &= g\{L_{i, j+N_i}, f\{L_{i, j}, R_{i+1, j}\}\} \\
R_{i, j} &= f\{R_{i+1, j}, g\{L_{i, j+N_i}, R_{i+1, j+N_i}\}\} \\
R_{i, j+N_i} &= f\{R_{i+1, j+N_i}, g\{L_{i, j}, R_{i+1, j}\}\}
\end{aligned} \tag{2.30}$$

, where $N_i = 2^{n-i}$ is the connection between each nodes.

Simulation result We now compare the decoders using SPA and using G-SPA. The BLER performances of GDMA-JCD and GDMA-SCD systems using a (256, 128) rate-1/2 Polar code with BPSK transmissions over quasi-static Rayleigh flat-fading channels are provided in Fig. 2.21. The maximum number of decoding iterations is set to 50 for both SPA and G-SPA. In the transmissions over block flat-fading channels, the GDMA-JCD scheme outperforms the GDMA-SCD scheme in low-SNR region, and the performances of two schemes converge to each other in high-SNR region. Since the block fading systems are lack of channel diversity, e.g., the probability of deep fading is very high, and the coding gain is not obvious. Hence, we further consider an OFDM-GDMA system that can obtain channel diversity from each sub-carrier using a (1024, 512) rate-1/2 Polar code that has 16 sub-carrier and 64 bits for each sub-carrier symbol. The BLER performance of GDMA-OFDM-JCD and GDMA-OFDM-SCD using the same parameters as above and are provided in Fig.2.22. We found that JCD can effectively use different user information, even

when increasing the number of users, performance has almost no loss.

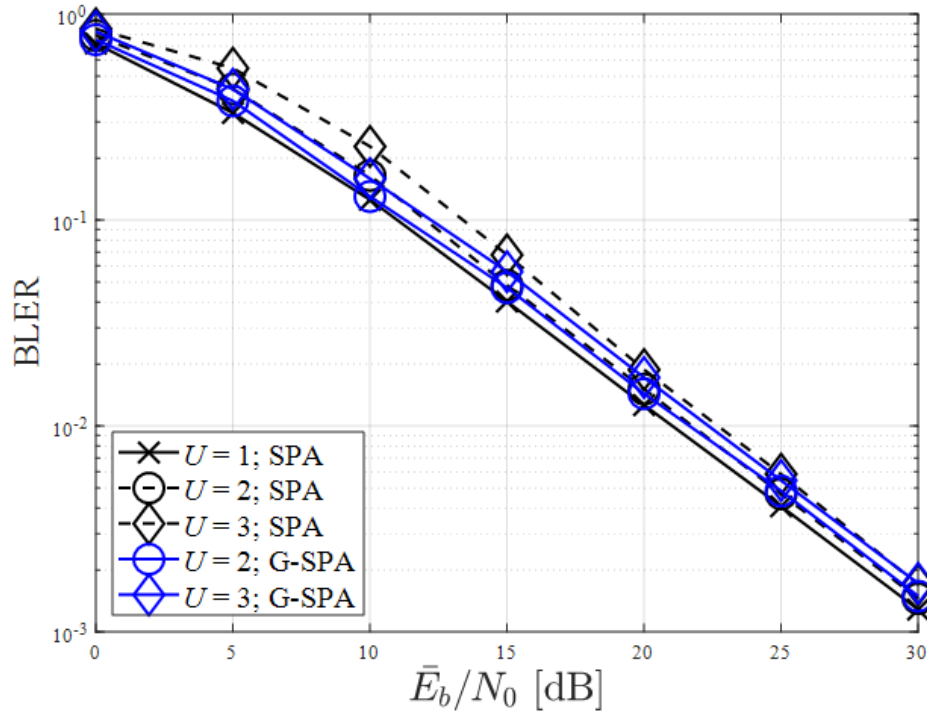
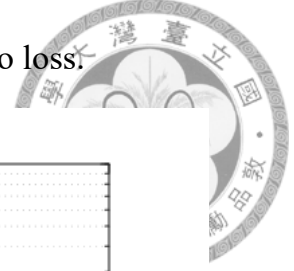


Figure 2.21: BLER performances of (256, 128) Polar-coded GDMA systems over quasi-static Rayleigh flat-fading channels.

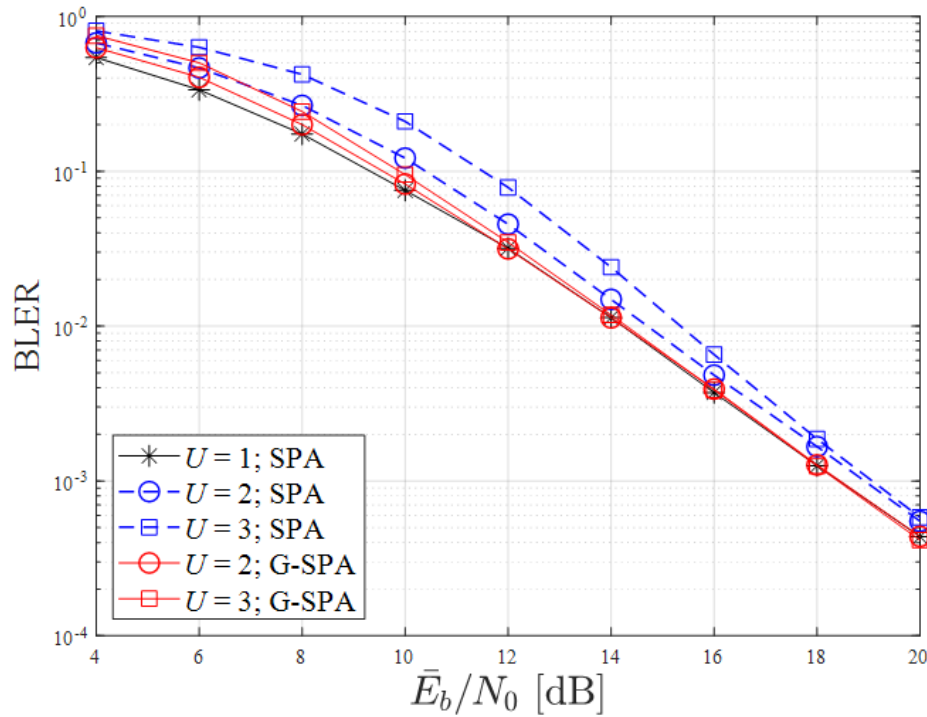
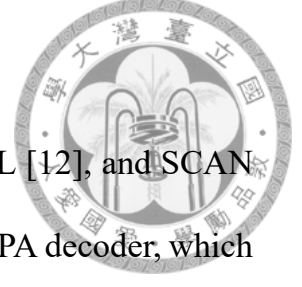


Figure 2.22: BLER performances of (1024, 512) Polar-coded GDMA OFDM systems over quasi-static Rayleigh flat-fading channels with 16-subcarrier.

Joint Successive Cancellation List decoder



There are some well-known decoders such as SPA [13], SCL [12], and SCAN for polar codes. These decoders have their advantages, such as SPA decoder, which is very simple in hardware implementation, and SCL with CRC is able to achieve the best performance in AWGN channel. Compared with the parallel operation of the SPA decoder, the SCL decoder's operation is in a specific order. For implementing a joint successive cancellation list (J-SCL) decoder, we still use $\mathbf{L}_{i,j}$ $\mathbf{R}_{i,j}$ to represent the message vectors and f -function g -function as the operation calculator where g -function a vector product calculation, f -function a vector parity check calculation.

Before message passing, we need to set the initial vectors. For simplicity, we set the initial vectors on the rightmost nodes as the APPs of the superimposed signals. Compared to the G-SPA decoder, the J-SCL decoder does not initialize the other node values.

The rightmost nodes $(n+1, j)$ are associated with channel input vectors \mathbf{p} that are observed through a noisy channel.

$$\mathbf{L}_{i,j} = [p_0 \ p_1 \ \cdots \ p_L]. \quad \mathbf{p}_1 = \Pr\{s = S[l]|r\} \quad (2.31)$$

Update for odd branch Fig.2.23 and equ. 2.32 show a basic computation unit for odd channels. The vectors passing to odd channels can be computed via recursion until leftmost nodes. If passing to the leftmost unfrozen nodes, it will map to the LLR value $e_{\text{LLR}}[c_{j,u}]$ according to mapping rules in chapter-2 and obtain a message

bit $\hat{u}_{j,u}$ that is the hard decoding value for the j -th codeword and u -user.



$$\mathbf{L}_{i+1,j} = f(\mathbf{L}_{i,j}, \mathbf{L}_{i,j+N_t}) \quad (2.32)$$

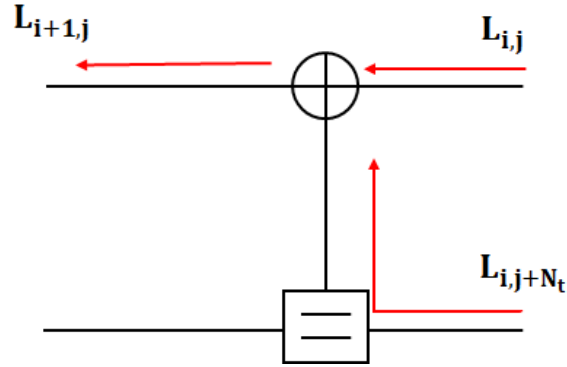


Figure 2.23: The factor graph of J-SCL upper layer (even branch) message passing.

Update for even branch Fig.2.24 and equ.2.33 show a basic computation unit for even channels. The vectors passing for even channels can also be computed via recursion. Instead of passing soft right propagating vectors $R_{i+1,j}$ in the G-SPA decoder, the vectors $\mathbf{R}_{i+1,j}$ are hard feedback vectors according to message bits $\hat{u}_{j,u}, u \in \{1, \dots, U\}$.

$$\mathbf{L}_{i+1,j+N_i} = g(\mathbf{L}_{i+1,j+N_i}, f(\mathbf{L}_{i+1,j}, \mathbf{R}_{i+1,j})) \quad (2.33)$$

Assume that $U = 2$ and BPSK transmission, the feedback vectors $\mathbf{R}_{i+1,j}$ are

described as :

$$\text{if } \hat{u}_{j,1} = 0 \text{ and } \hat{u}_{j,2} = 0, \mathbf{R}_{n+1,j} = [1 \ 0 \ 0 \ 0]$$

$$\text{if } \hat{u}_{j,1} = 0 \text{ and } \hat{u}_{j,2} = 1, \mathbf{R}_{n+1,j} = [0 \ 1 \ 0 \ 0]$$

$$\text{if } \hat{u}_{j,1} = 1 \text{ and } \hat{u}_{j,2} = 0, \mathbf{R}_{n+1,j} = [0 \ 0 \ 1 \ 0]$$

$$\text{if } \hat{u}_{j,1} = 1 \text{ and } \hat{u}_{j,2} = 1, \mathbf{R}_{n+1,j} = [0 \ 0 \ 0 \ 1]$$

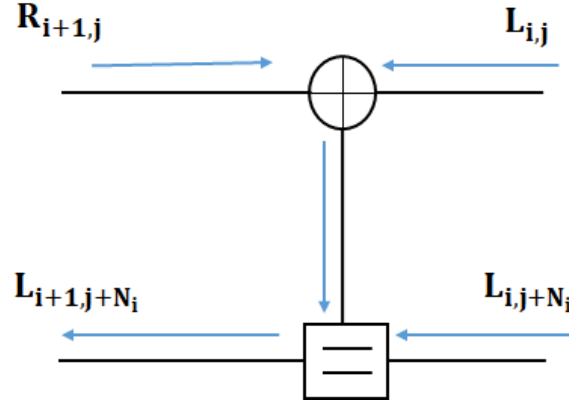


Figure 2.24: The factor graph of J-SCL lower layer (odd branch) message passing.

Once obtain $\mathbf{R}_{n+1,j}$ from the leftmost nodes, we can also derive right propagating vectors $\mathbf{R}_{i,j}$ via recursion as

$$\mathbf{R}_{i,j} = f(\mathbf{R}_{i+1,j}, \mathbf{R}_{i+1,j+N_i})$$

$$\mathbf{R}_{i,j+N_i} = \mathbf{R}_{i+1,j+N_i}$$

Metric path After obtain the leftmost probability vectors, we convert probability vectors to the LLR values $e_{\text{LLR}}[c_{j,u}]$ where $u \in \{1, 2, \dots, U\}$ $j \in \{1, 2, \dots, N\}$ of each user, which can be obtained by the above formula. Besides, we must record the

PM value of this period where PM value is used in list decoding to record the LLR value of each metric path. SCL decoder will only have two forks per path, which are LLR values equal to 0 or 1 respectively. Different from SCL decoder, the forks of each J-SCL path is $L = 2^{mU}$ since the path metric PM_l^i must contain each user hard decoding information, where L is the number of superimposed signals, we can express the expression as follows:

For the l -th path and the i -th level, $i = 1, 2, \dots, N$, the PM value is given by

$$\text{PM}_l^i \triangleq \sum_{j=1}^i \sum_{u=1}^U \ln\{1 + \exp[-(1 - 2\hat{u}_{j,u})e_{\text{LLR}}]\} \quad (2.34)$$

where $\hat{u}_{j,u}$ are the hard decoding value for the j -th codeword and u -th user according to $e_{\text{LLR}}[c_{j,u}]$.

Simulation result The BLER performances of GDMA-JCD and GDMA-SCD systems using a (256, 128) rate-1/2 Polar code with CRC-16 BPSK transmissions over quasi-static Rayleigh flat-fading channels are provided in Fig. 2.25. The maximum number of decoding iterations is set to 50 for both SPA and G-SPA. In the transmissions over block flat-fading channels, the GDMA-JCD scheme outperforms the GDMA-SCD scheme in low-SNR region, and the performances of two schemes converge to each other in high-SNR area. We further consider an OFDM-GDMA system using a (1024, 512) rate-1/2 Polar code with CRC-16 that has 16 sub-carrier and 64 bits for each sub-carrier symbol. The BLER performance of GDMA-OFDM-JCD and GDMA-OFDM-SCD using the same parameters as above

and are provided in Fig. 2.26. We found that JCD can effectively use different user information, even when increasing the number of users, performance has almost no loss.



Since the polar construction is based on the same SNR of received signals, the received signals from OFDM system are distinct according to different energy of each sub-carrier channel gains. Hence, it can use interleaver to force the received signals into similar SNR, and the details were discussed in the paper [14]. In fig.2.27, we compare the performance of G-SPA and J-SCL decoder for polar code and assume code length of 1024, 16 sub-carrier OFDM system, $U = 3$. Besides, add the interleaver to make the received message to avoid sub-block deep fading. Then, add CRC check to SCL and J-SCL for better performance, where CRC length

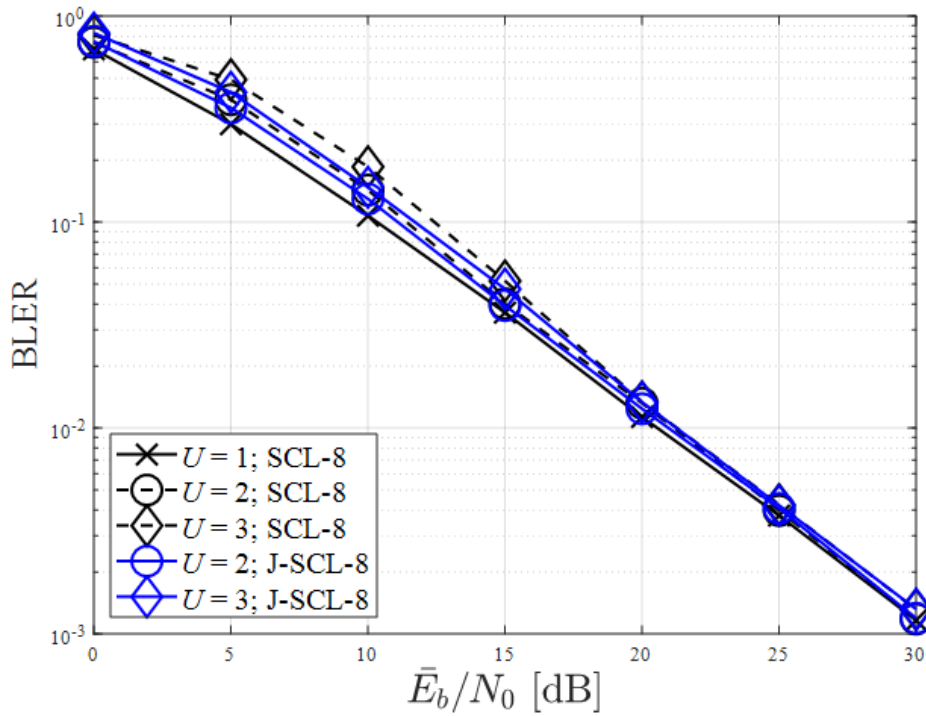


Figure 2.25: BLER performances of (256, 128) Polar-coded GDMA systems over quasi-static Rayleigh flat-fading channels.

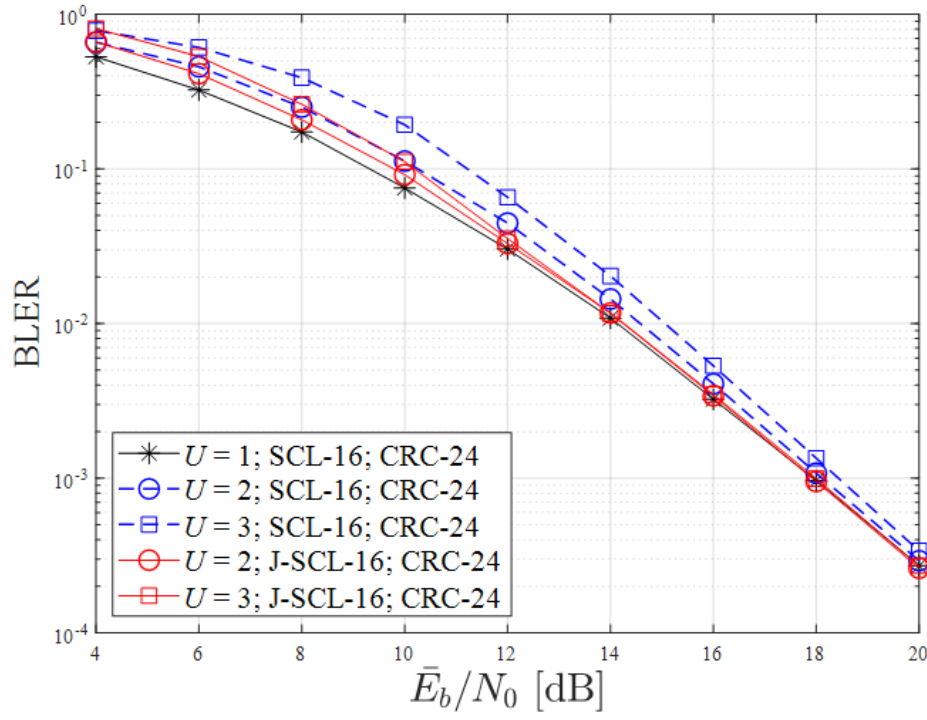


Figure 2.26: BLER performances of (1024, 512) Polar-coded GDMA OFDM systems over quasi-static Rayleigh flat-fading channels with 16-subcarrier.

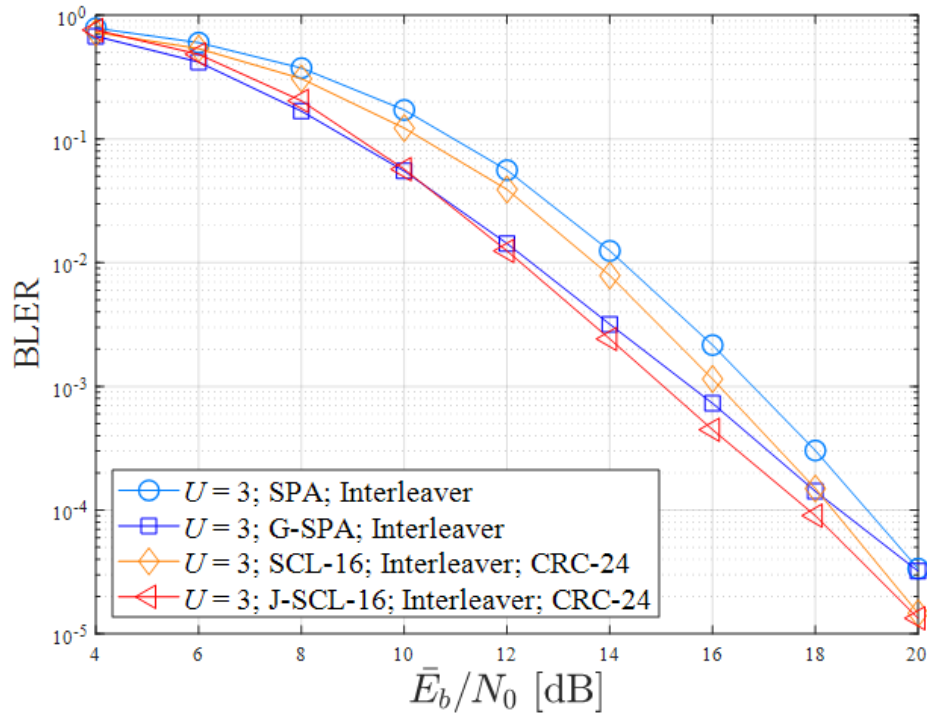


Figure 2.27: BLER performances of Polar-coded GDMA OFDM systems over quasi-static Rayleigh flat-fading channels with 16-subcarrier and interleaver.

24 is same as paper [15]. We can find that the JCD decoder's performance is better than SCD decoders from 4 dB to 18 dB, and different type decoders will converge in different performance in high SNR.





Chapter 3

Cluster-Based Channel Estimation

In the thesis [4], the cluster-based channel estimation technique was proposed for efficient transmission. It used different clustering algorithms to analyze the mean square error (MSE) performance. However, it still couldn't achieve to Cramer-Rao lower bound (CRLB) when U was higher than two, and the proposed methods [4] for deriving channel coefficients was a particular case that was only suitable for $U = 2$ or $U = 3$. Hence, we proposed a general algorithm for clustering received signals and deriving the channel coefficient for any user and modulation.

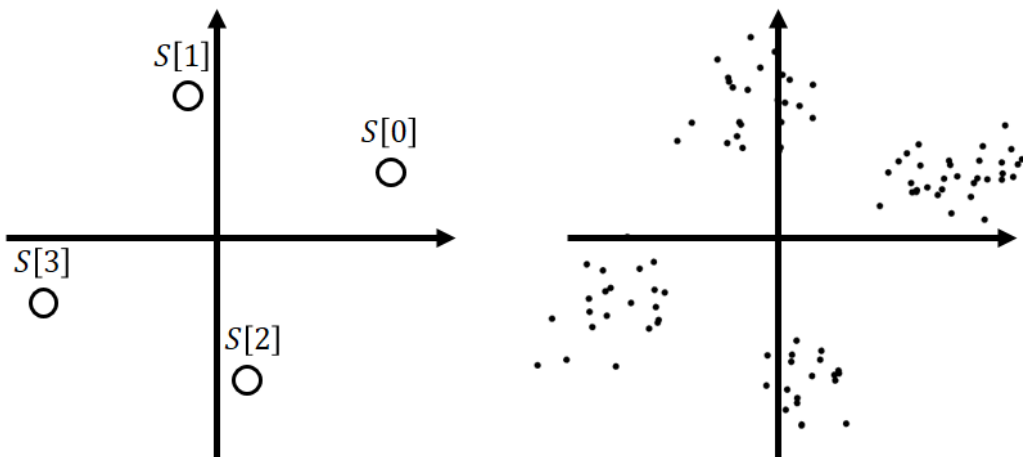


Figure 3.1: Superimposed levels and the corresponding received signals in the case of $m = 1$ and $U = 2$.

Introduction for Cluster-based Channel Estimation When multiple users transmit through independent fading channels, the signals can be further separated by using different channel coefficients without requiring user-specific resources. The requirement for applying multi-level detection in the GDMA system is to accurately recover the channel coefficients at the receiver to construct all possible superimposed signal levels. Due to the presence of channel noise, the received signal in the block where the channel remains unchanged has a two-dimensional Gaussian distribution with mean $S[l]$ where $l \in \{0, 1, \dots, 2^{mU}\}$ according to the messages of U users. If we can classify the received symbols into a 2^{mU} group in some way, where the symbols corresponding to the same level are grouped, the estimated value of the level can be obtained by the average value of the symbols in each group. In this section, we assume that the fading coefficients are constant across a block of N_t consecutive symbols. An example of the received signals with $N_t = 100$ in the case of $m = 1$ and $U = 2$ is shown in Fig.3.1. Therefore, we use clustering algorithm to classify the received signals and blindly estimate the channel coefficients of U users. Clustering is an unsupervised learning technique, which involves grouping training samples from different groups with different characteristics. Note that the clustering technique in the proposed scheme takes the signal with message as the input training sample, and since the estimation does not require additional pilot signals, it can achieve high frequency spectral efficiency and inevitable ambiguity in estimation.



3.1 Review of Clustering Algorithm



Clustering is an unsupervised learning technique and the task of grouping a set of objects that called a cluster. In this section, we review the clustering algorithm survey from the thesis [4].

3.1.1 K-Means Algorithm

One of the most popular clustering techniques is the k-mean algorithm [16], which aims to minimize the sum of the square distance between each sample and its closest centroid. In the k-means problem, it is desirable to choose a set of centroids S such that the following function can be minimized.

$$\phi = \sum_{x \in X} \min_{s \in S} \|x - s\|^2, \quad (3.1)$$

where X is the set of training vectors, and ϕ is the corresponding distortion. Therefore, K-means algorithm classifies samples by iteratively pairing each sample with the closest centroid, and then updating the centroid from the newly derived group.

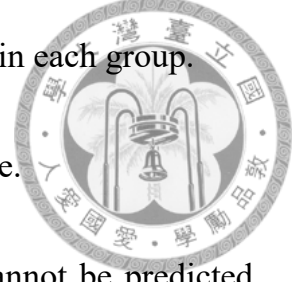
For channel estimation in the GDMA system, all the levels should be distinguished. The number of groups to be classified is 2^{m_U} in the clustering. The steps of the k-means algorithm are organized as follows.

Step 1) Randomly pick the initial centroids from the symbols in the received sequence.

Step 2) Classify the received symbols into 2^{m_U} groups by applying the nearest-neighbor rule with respect to the current centroids.

Step 3) Recompute the centroids by averaging the symbols in each group.

Step 4) Return to Step 2 until the centroids no longer change.



The required number of iterations for clustering algorithms cannot be predicted ahead of time. In addition, the random seeding in the initial step of k-means may yield different grouping results in different runs of the algorithm.

In the k-means algorithm, the most crucial task is selecting acceptable initial centroids in the correct solution's neighborhood since it can only find a local minimum. Several clustering algorithms were proposed to select acceptable initial centroids such as LBG algorithm, k-mean++ algorithm, modified k-means++ algorithm.

3.1.2 The LBG Algorithm

The Linde – Buzo – Gray (LBG) algorithm [6], sometimes referred to as generalized Lloyd algorithm (GLA), is a technique to design vector quantizer and a so-called "splitting" method is employed to select the initial centroids for the following k-means clustering. The steps of LBG algorithm are organized as follows.

Step 1) Split the origin into two close points ϵ and $-\epsilon$ and hence the set of initial centroids $S = \{\epsilon, -\epsilon\}$ with size $M = 2$ is obtained.

Step 2) Classify the received symbols into M groups, based on the current centroids S , by applying the nearest-neighbor rule and compute the centroid ρ_i of each group with $i \in \{1, 2, \dots, M\}$.

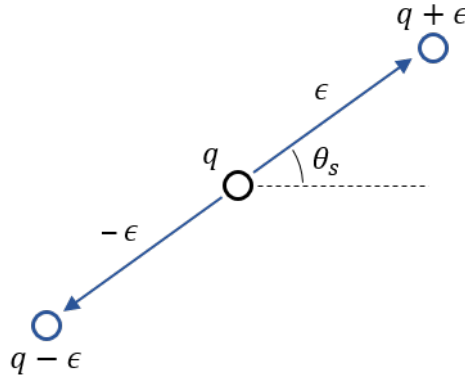


Figure 3.2: Splitting from q to get $q + \epsilon$ and $q - \epsilon$ with a perturbation vector ϵ .

Step 3) Split ρ_i into two close points $q_{2i-1} = \rho_i + \epsilon$ and $q_{2i} = \rho_i - \epsilon$ with $i \in \{1, 2, \dots, M\}$ and the set of new centroids $S = \{q_i | i = 1, 2, \dots, 2M\}$ is obtained.

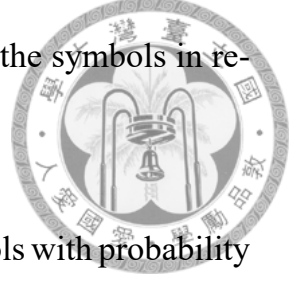
Step 4) Replace M by $2M$ and return to Step 2 until $M = 2^{m_U}$.

Step 5) Proceed with the standard k-means clustering.

3.1.3 K-Means++ Algorithm

The K-means ++ algorithm is proposed in the kmpp07 to select the initial centroid for the following K-means clustering, which can be regarded as a generalization of the standard k-means algorithm. The concept of K-means ++ algorithm is to select the initial centroid as randomly as possible. Note that randomness is preserved in initialization, so K-means ++ may not rely on the accuracy of training samples as LBG does. Let $D(r)$ denote the distance from the received symbol r to the closest centroid we have selected.

The steps of k-means++ algorithm are organized as follows.



Step 1) Pick the first centroid q_1 uniformly at random from the symbols in received sequence and set $i = 1$.

Step 2) Pick the next centroid q_{i+1} from the remaining symbols with probability proportional to $D^2(r)$ and replace i by $i + 1$.

Step 3) Return to Step 2 until $i = 2^{mU}$.

Step 4) Proceed with the standard k-means clustering.

3.1.4 Modified K-Means++ Algorithm

The modified k-means++ algorithm was proposed in [17] to choose the initial centroids for the following k-means clustering, which can also be seen as a generalization of standard k-means algorithm. Compared to the k-means++ algorithm, it restricts the search space to a set Ω when selecting the next centroid q_{i+1} . This restriction might decrease the probability of selecting unsuitable initial centroids and still maintaining the randomness property for selecting centroids. Let $D(r)$ denote the distance from a received symbol r to the closest centroid we have already chosen. The steps of the modified k-means++ algorithm are organized as follows.

Step 1) Pick the first centroid q_1 uniformly at random from the symbols in received sequence and set $i = 1$.

Step 2) Compute $D(r)$ which is the shortest distance between $r \in \mathbf{r}$ and any initial centroid in $\{q_1, \dots, q_i\}$. Decide the set Ω that is comprised of $n/(2^{mU})$ symbols which have larger $D(r)$ values than those symbols

in $\mathbf{r} \setminus \Omega$. Select the next centroid q_{i+1} from Ω according to a probability proportional to $D^2(r)$, and replace i with $i + 1$.



Step 3) Return to Step 2 until $i = 2^{mU}$.

Step 4) Proceed with the standard k-means clustering.

Compared above clustering algorithms, the modified k-means++ algorithm has the best performance in [17], but it still cannot achieve the CRLB of MSE.

3.1.5 Gaussian Mixture Model

Clustering technology can be divided into two types: soft clustering and hard clustering. In hard clustering, each sample is accurately assigned to a group by applying the nearest neighbor rule. However, when samples from different sources are highly overlapped, it is difficult to tell which group the samples belong to in the overlap region. In Fig. 3.3, two sources produce samples with two-dimensional Gaussian distribution, and the mean values are represented as red dots. The Red

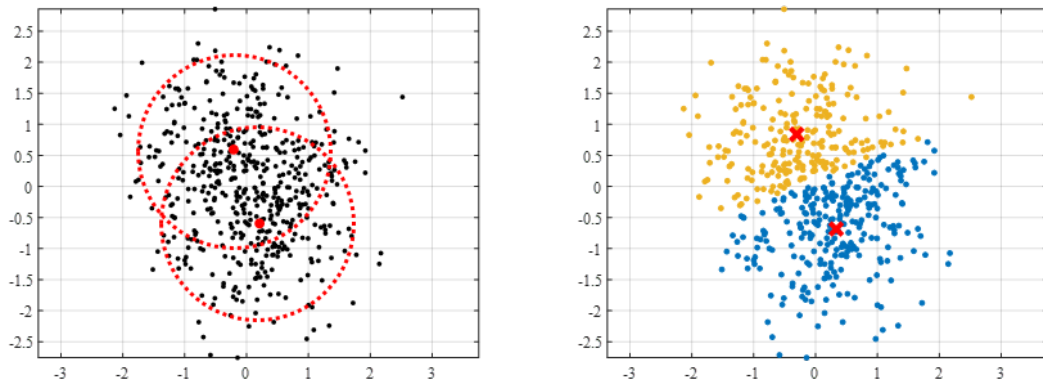


Figure 3.3: Grouping result of hard clustering by applying nearest neighbor rule.

Cross is the center of mass of the composition, and the color is shown in the figure Fig. 3.3. The group classified by hard clustering in clustering. Fig. 3.3 is the received symbol in BPSK modulation transmission of $U = 1$ user, and the SNR is relatively low. Therefore, the interference of channel noise makes it difficult to distinguish the two levels and reduces the accuracy of estimation.

Gaussian mixture model (GMM) is a method to realize soft clustering. It assumes that samples from different sources are Gaussian distributions with different parameters (mean and covariance). Therefore, the clustering problem is transformed into parameter optimization of Gaussian component in GMM. Expectation maximization (EM) algorithm [8] is usually used to find the parameters of Gaussian components in GMM. Since the channel noise is Gaussian distribution, the assumption of Gaussian component in GMM is suitable for the clustering of received symbols. In addition, as mentioned above, since the variance of each Gaussian component is equal to the variance of channel noise, and the probability of occurrence is equal, only the mean value of the component needs to be considered. For the signals received in GDMA system, the steps of EM cluster using GMM are as follows.

Step 1) Initially set the occurrence probability $\Pr(i) = 2^{-mU}$ and the covariance matrix of Gaussian distribution is $\Sigma_i = \sigma_w^2 \mathbf{I}_2$, where σ_w^2 is the variance of channel noise and \mathbf{I}_2 is a two-by-two unit matrix, for each component i with $i \in \{1, 2, \dots, 2^{mU}\}$.

Step 2) Proceed with k-means clustering and the resultant centroids are config-

ured as the initial means of Gaussian components.



Step 3) Compute the soft assignment of each received symbol $r(n)$ by

$$\Pr(i|r(n)) = \frac{\mathcal{CN}(r(n)|\mu_i, \Sigma_i)}{\sum_{j=1}^{2^{mU}} \mathcal{CN}(r(n)|\mu_j, \Sigma_j)}, \quad (3.2)$$

where $i \in \{1, 2, \dots, 2^{mU}\}$ and \mathcal{CN} denotes the PDF of complex Gaussian distribution.

Step 4) Derive the ML estimate of mean for component i with

$$\mu_i = \frac{1}{N_i} \sum_{n=1}^{N_t} r(n) \Pr(i|r(n)), \quad (3.3)$$

where $i \in \{1, 2, \dots, 2^{mU}\}$ and $N_i = \sum_{n=1}^{N_t} \Pr(i|r(n))$.

Step 4) Return to Step 3 until the means no longer change.

The procedure of EM clustering is quite similar to that of the k-means algorithm, which needs the initialization of the parameters. Here we directly use the centroids derived from the k-means++ algorithm to be the initial means of Gaussian components in GMM, as stated in Step 2. After the EM clustering, the means of elements become the resultant centroids.

3.2 Cluster-based Estimation Analysis and Design

Cluster-based estimation can increase the efficiency of transmission, but it still has many problems. First, we analyzed how many samples are needed to achieve cluster reliability. Second, compared to [4] only using the symmetric condition to

judge whether it has correctly converged, we added more restrictions to increase algorithm security. Third, compared to the case where [4] will plunge into fail loops to find correct convergence, we propose a method that can significantly reduce the number of iteration. Fourth, compared to the limitation [4] of derivating channel coefficient, the algorithm we proposed can be applied in various cases.



3.2.1 Clustering Sample Size Analysis

The mean-square error (MSE) performances are simulated to evaluate the performance of cluster-based estimation applied to Rayleigh flat fading channels ,and the MSE is defined as

$$\text{MSE}(h, \hat{h}) = \frac{1}{2}|h, \hat{h}|^2 \quad (3.4)$$

where h is the exact channel coefficient and \hat{h} is the estimate of h . Note that the average energy of channel coefficient and transmitted signals are both normalized to one. The Cramer-Rao lower bound (CRLB), for the minimum variance unbiased (MVU) estimator using N_t samples in the estimation, is also provided to evaluate the performance. Consider the observations when the signals are propagated through the channel with AWGN as $r(n) = h + w(n), n = 0, 1, \dots, N_t - 1$.

,where h is the parameter to be estimated and $w(n)$ is AWGN with variance σ_w^2 .

The MVU estimator of h can be derived as



$$\hat{h} = \frac{1}{N_t} \sum_{n=0}^{N_t-1} r(n) \quad (3.5)$$

and the variance of \hat{h} is the CRLB written as

$$\sigma_h^2 = \frac{\sigma_w^2}{N_t} \quad (3.6)$$

Assume that $U = 2$ and BPSK transmission, we changed the sample size arbitrarily. Observe the fig.3.4 and find that the smaller the size, the earlier the error floor will occur. Because the number of samples is lack, some superimposed levels

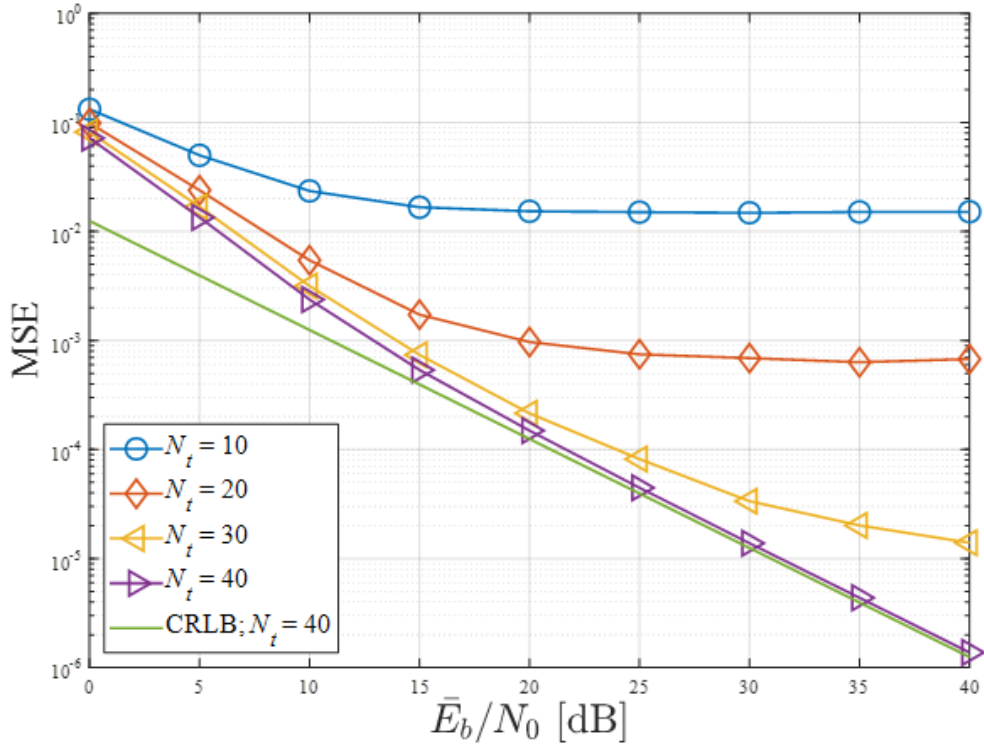


Figure 3.4: MSE of estimated channel coefficient over AWGN channels.

may not be generated and must lead to errors. We called it the worst case. In the following formula derivation, we derived the probability that the sample size will lead to the worst case.



Assume that $L = 2^{mU}$ is the number of superimposed levels, N_t is the number of clustering points and A_i is the random variable that i -th superimposed level is empty where $i \in \{1, 2, \dots, L\}$.

$$\begin{aligned}
 \Pr(A_i) &= \left(\frac{L-1}{L}\right)^{N_t} \\
 \Pr(A_i \cap A_j) &= \left(\frac{L-2}{L}\right)^{N_t}, i \neq j \\
 &\vdots \\
 \Pr(A_i \cap A_j \cdots \cap A_r) &= \left(\frac{1}{L}\right)^{N_t}, i \neq j \cdots \neq r
 \end{aligned} \tag{3.7}$$

$$\Pr(\text{error}) = \Pr(\text{at least one level is empty})$$

$$\begin{aligned}
 &= \Pr(A_i \cup A_j \cdots \cup A_r) \\
 &= C_1^L \Pr(A_i) - C_2^L \Pr(A_i \cup A_j) + C_3^L \Pr(A_i \cup A_j \cup A_j) + \cdots \\
 &+ (-1)^{L-2} C_{L-1}^L \Pr(A_i \cup A_j \cup A_j \cdots) \\
 &= C_1^L \left(\frac{L-1}{L}\right)^{N_t} - C_2^L \left(\frac{L-2}{L}\right)^{N_t} + C_3^L \left(\frac{L-1}{L}\right)^{N_t} + \cdots + (-1)^{L-2} C_{L-1}^L \left(\frac{1}{L}\right)^{N_t}
 \end{aligned} \tag{3.8}$$

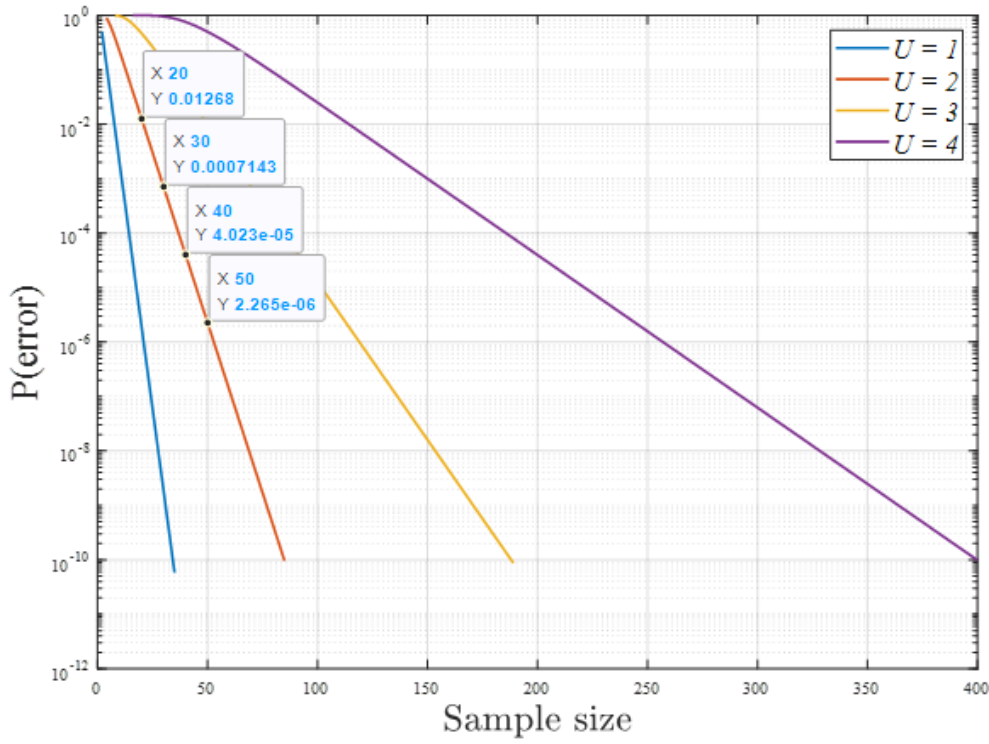


Figure 3.5: The probability of worst case with BPSK transmission for different user U .

In Fig.3.5, derive the probability that superimposed levels is empty according to above equality. and list the following table.3.1 that is probability with different clustering samples size. Fig.3.6 is the MSE of paired sample. Expect for $N_t = 10$, sample size $N_t = 20$ and $N_t = 30$ both achieve the CRLB.

We can make up the problem according to the structure of modulation. Generate a sample with π phase shift for BPSK transmission and create three copies of the received sample for QPSK transmission with $\frac{\pi}{2}$, π , $\frac{3\pi}{2}$ phase shift respectively.

Table 3.1: The probability that at least one superimposed level is empty for $L = 4$.

| Sample size | 10 | 20 | 30 | 40 | 50 |
|-----------------------|--------|---------|---------------|---------------|---------------|
| $\Pr\{\text{error}\}$ | 0.2194 | 0.01268 | $7.143e^{-4}$ | $4.023e^{-5}$ | $2.265e^{-6}$ |

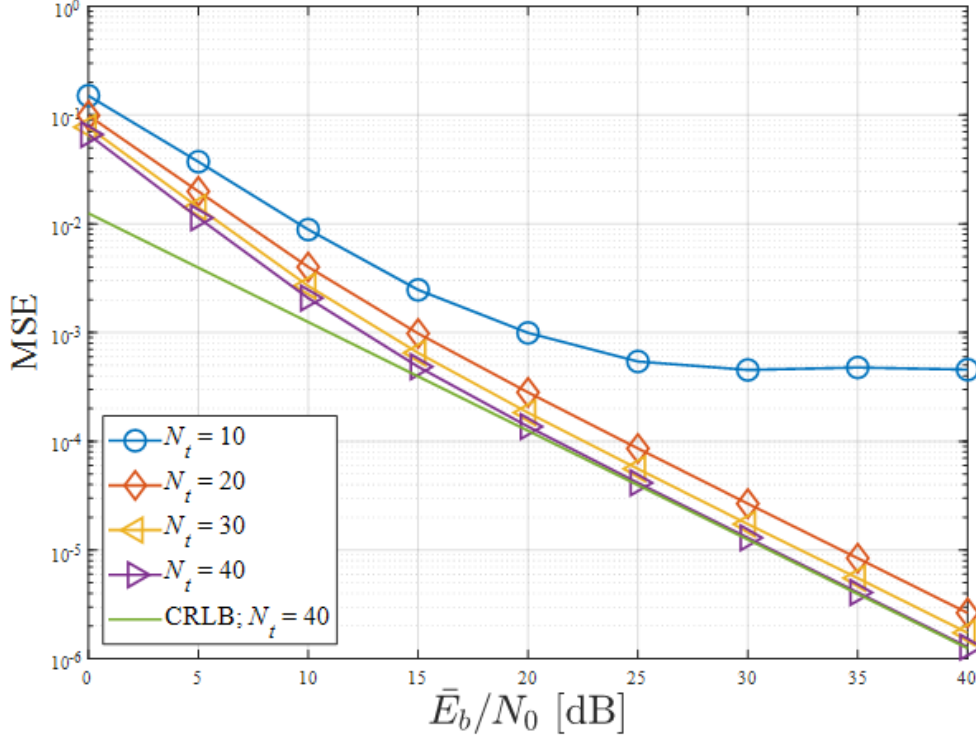


Figure 3.6: MSE of estimated channel coefficient with pair samples over AWGN channels.

3.2.2 Proposed Clustering Algorithm

In the thesis [4], the main focus was on selecting good initial centroids, and a modified k-mean algorithm has also been proposed to make the initial centroid choosing better. In most of the clustering literature, it is mainly in the study of the initial centroids. Although the initial centroid selection is very important, it isn't easy to cluster correctly by completely modifying the initial selecting algorithm. We have come up with a different idea, not the study of initial centroids. The clustering algorithm in [4] will re-execute if the symmetric condition is not met. The re-executed clusterings are independent of the previous one, resulting in inefficient or falling into a wrong loop. Hence, we came up with new ideas to make these re-executed clustering related to the last grouping and make them more efficient.

Fig. 3.7 is the flow chart of proposed clustering algorithm. The cluster check is the algorithm to check whether convergence centroids are correct, roughly using the symmetric properties, group size, and group MSE. According to the cluster check, process the algorithm of resetting special centroids, which only reset the incorrect centroids. The structure check is the function to check centroids more accurately by using the superimposed level structure, and the channel coefficient derivation function is the novel algorithm different from [4]. The details of each flow block are described as the following subsection.

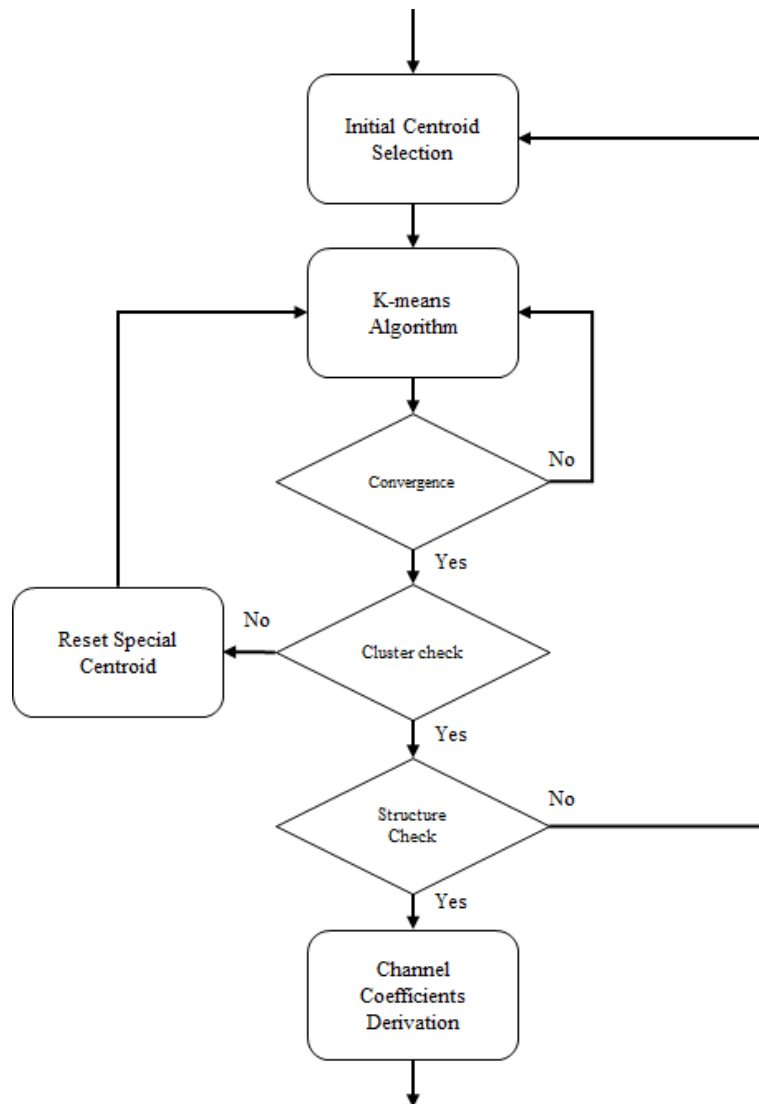


Figure 3.7: The flow chart of proposed cluster-based channel estimation algorithm.

Cluster Check

In addition to selecting better initial centroids, we also change the algorithm after convergence. The fig.3.8-(a) is a cluster convergence case, but it does not conform to the superimposed levels symmetry condition. Thus, in the algorithm [4], the clustering algorithm would re-execute until the estimated centroids symmetry, but since each clustering algorithm is independent, it is possible to fall into a loop. Besides, even if it meets the symmetry condition, it could still be the wrong convergence.

Fig.3.8-(a), one case of convergence, it does not conform to the symmetric condition, but green centroids converge to the correct superimposed levels, where only red centroids do not. Hence, we can estimate whether those centroids converge correctly. Here, we use group MSE and group size to estimate centroids individually to check which centroid is reliable.

Group MSE is the mean square error (MSE) of each cluster's centroid and its members. Under normal circumstances, group MSE should be similar to the noise variance. Therefore, if the group MSE is unusually large or small, this cluster is an error. Group MSE can be expressed as

$$MSE_k = \sum_{i=0}^{N_k} \frac{(r_{k,i} - C_k)^2}{N_k}, \quad k = 1, 2, \dots, L \quad (3.9)$$

, where N_k is the group size of k -th cluster, C_k is the centroid of k -th cluster.

Group size is the number of members of each cluster. Under normal circum-

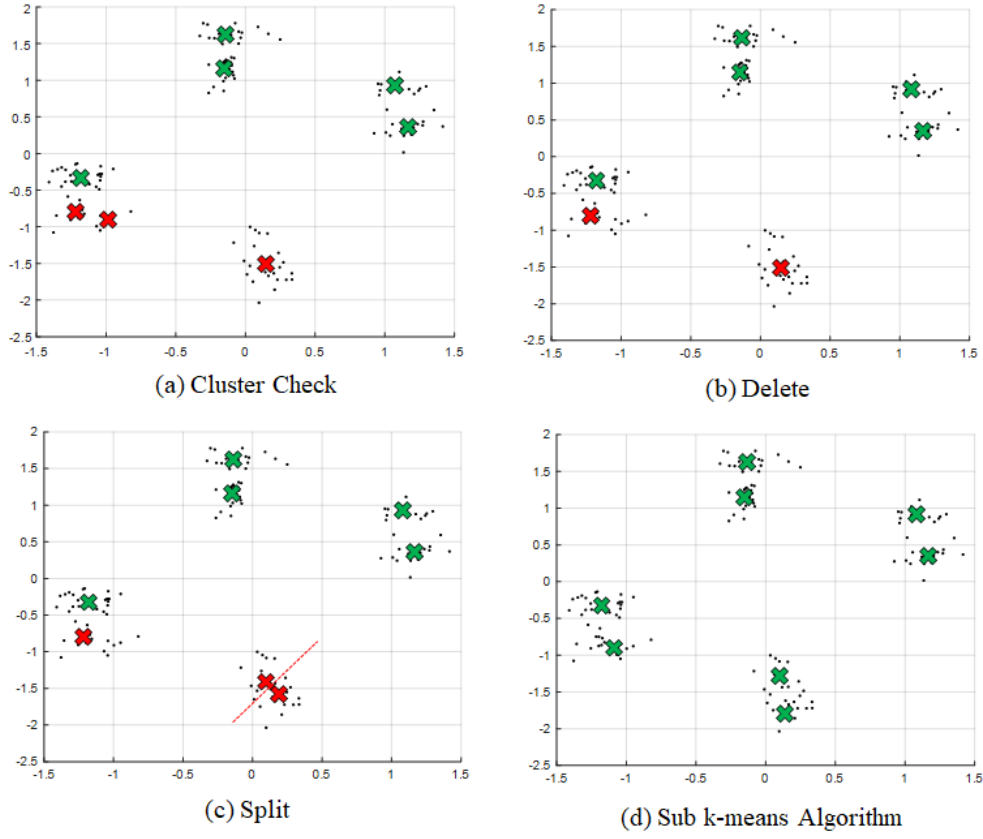


Figure 3.8: An example of cluster check.

stances, the group size should be $\frac{N_t}{L}$ approximately and it can also be utensil to estimate. $N_k, k = 1, 2, \dots, L$

We map group MSE or size into a reliability value for each centroid. If the value is smaller or larger than a threshold, label it a bad centroid. However, neither group MSE nor group size can not be entirely correct. We use them alternately until the estimated superimposed levels are the right structure.

Reset Special Centroid

We obtain the bad centroids and good centroid after cluster check. It is only necessary to re-estimate the bad centroids that save a lot of operations and also ensure that the entire algorithm will not enter an error loop again. If the reliability value is

higher than the threshold, it indicated that the cluster group might miss centroids. In comparison, the value is less than the threshold indicated that the cluster group might have too many centroids. Firstly, delete the centroid with the smallest reliability value, which might be the redundancy centroids in fig.3.8-(b). Secondly, split the centroid with the most significant parameter, which cluster might miss centroids in fig.3.8-(c). The concept of splitting is same as LBG [6] algorithm where splitting from centroid q to get $q + \epsilon$ and $q - \epsilon$ with a perturbation vector ϵ . Both add a random vector to the original centroid, splitting the centroid into two. Third, perform the k-mean algorithm on these reset centroids, so that all cluster parameters are close to convergence. Hence, the proposed clustering estimation is more efficient than the algorithm [17], and we used the clustering iteration to compare two algorithms in fig.3.9 at $N_t = 128$, where iteration is the number of operations

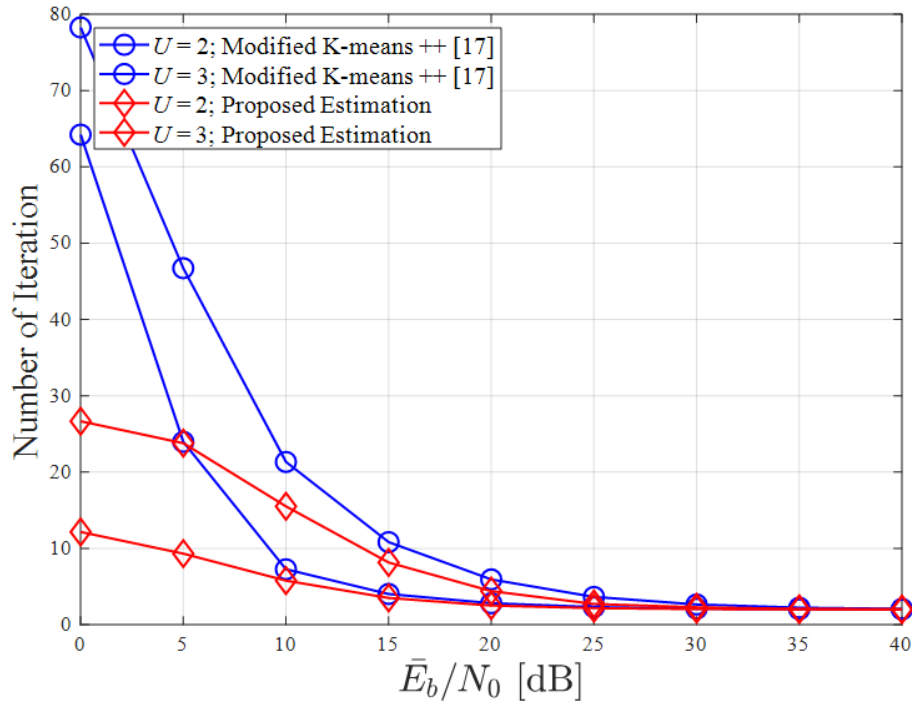


Figure 3.9: The iteration number of estimated channel gain at $N_t = 128$.

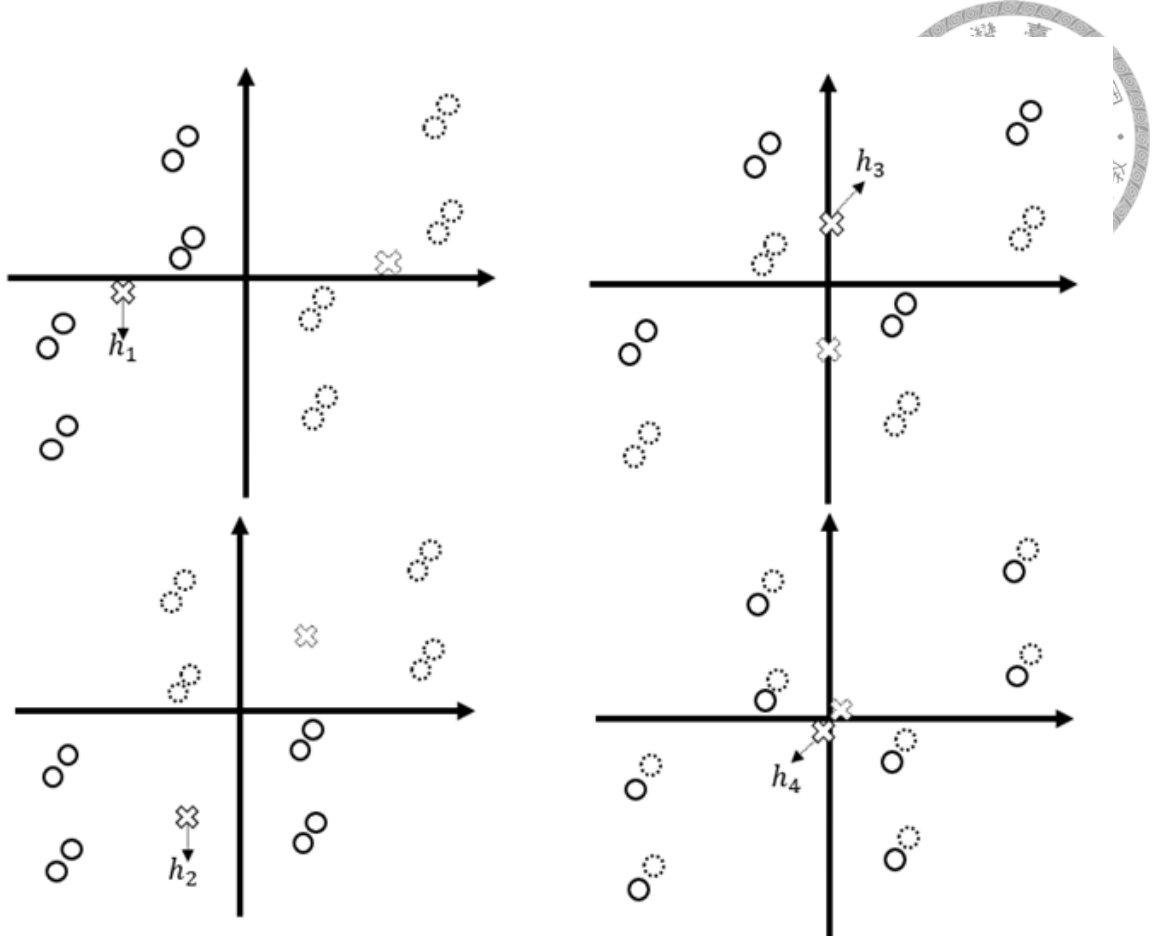


Figure 3.10: An example of structure check.

required for the clustering algorithm to converge to meet all structural constraints. The proposed clustering algorithm can greatly reduce iteration, especially in low SNR.

Structure Check

After the above steps, the estimated superimposed levels can be correctly solved with a very high probability. This step is mainly to detect whether estimated centroids are entirely correct. Because this step requires a higher degree of complexity and will be used in the channel estimation, it is set in the final step detection. The fig.3.12-(a) is $U = 4$ BPSK transmission where h_u is channel gain $u \in \{1, 2, 3, 4\}$

and the solid circles are the superimposed levels. We can find it has the following characteristics: dividing the superimposed level into two groups: solid circles and dotted circles in fig.3.10. Because of half of the superimposed added with h_u and the other half added with $-h_u$, we can find the different symmetrical property with h_u ; $u \in \{1, 2, 3, 4\}$ as the center. If it conforms to the structure, we can be very sure that the estimated centroids are correct. This structure will also be used in the next section, so there will be a more detailed discussion on how to implement it in the next section .3.2.3.



3.2.3 Derivation of Channel Coefficients

We can classify the received symbols into 2^{mU} groups by the clustering techniques introduced in Sec. 3.1.1, and the centroids of groups are the estimated superimposed levels. Recall that the fading coefficients are constant across a block of N_t consecutive symbols, and the occurrence probabilities of superimposed levels are equally like in assumption. Using the derivation channel coefficient algorithm as following, we can further estimate the channel coefficients of U users from the resultant centroids. In this section, ideal clustering that the centroids are exactly equal to superimposed levels is assumed to clearly illustrate the procedure.

Before introducing the derivation of the channel coefficients algorithm, we must know how the superposition levels compose, and here we find the rules for its composition. As shown in the fig.3.11, the levels are superimposed from channel gains h_1 to h_4 in sequence. The fig-(a) is the signal generated by h_1 . The amplitude and phase of levels are the same as h_1 because of the BPSK transmission. The fig-(b)

is the superimposed levels generated by h_1 and h_2 . The superimposed levels are obtained by splitting $\pm h_2$ from fig-(a), so the center of these two splitting groups is $\pm h_2$. Fig-(c) is the superimposed levels generated by h_1 , h_2 and h_3 . Similarly, $\pm h_3$ can be added to the four levels in fig-(b) separately, so it can be split into eight levels. Then, we can derive 2^{mU} superimposed levels from U channel gains, which is the rule of superimposed signals. On the contrary, we can also derive channel gains from superimposed signals by the iterative process.

From the rule of superimposed level composition, channel gains can be solved one by one, which removes the level of information of solved gains. In this way, we only need to find the remaining channel gains, which can save a lot of operations. The steps of the derivation of channel coefficients is organized as follows.

Step 1) Pair the 2^{mU} levels with the sum of each pair is equal to zero according to modulation.

Step 2) Group 2^{mU} levels into $2m$ groups, each signal of groups are not in the same pair, and derive the centroids of each group, where m is the modulation order.

Step 3) Return to Step 2 until each group is symmetrical with their centroids.

Step 4) Derive the channel coefficient that is the group centroid.

Step 5) Shift $2m$ groups to the original point.

Step 6) Return to Step 1 and $U = U - 1$ until all coefficients are derived.

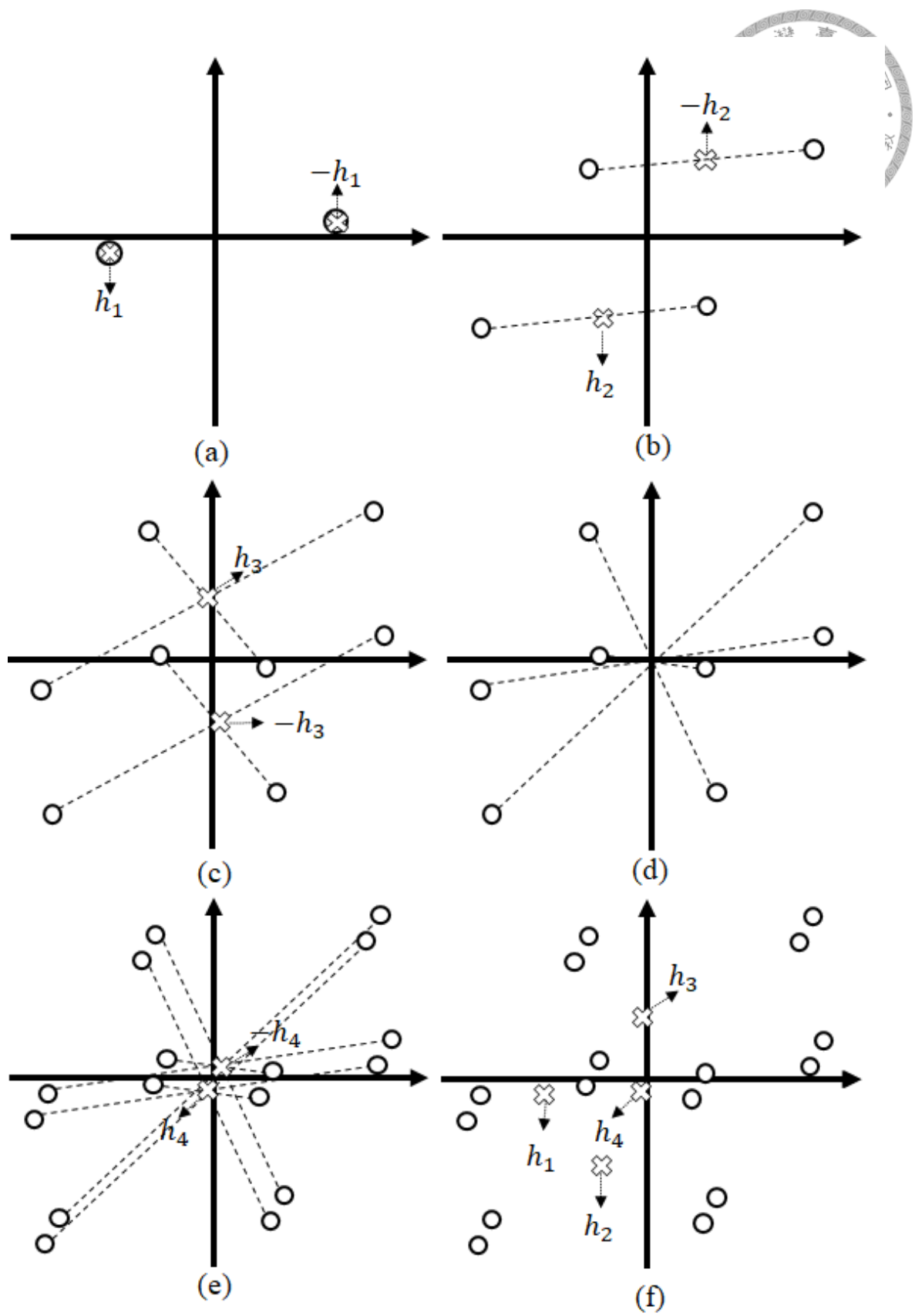


Figure 3.11: The steps to compose superimposed levels.




Fig.3.12 is process of derivation of channel coefficient. The first step is to pair the centroids with each pair's sum equal to zero in fig-(b). Step 2 is the brute force method for getting all compositions, and there are $2^{2^{mU}-1}$ combinations in total, but only $2mU$ combinations are correct that is an exponential operation according to U user. Fig-(b) is the false case, and fig-(c) is the correct case. Then, deposit the derived estimated channel gain, which is the centroid of the two groups. We only need to derive the remaining channel gains. According to modulation order m , the group number is also different. In step 5, shift the symmetrical groups into the original point and average the pair levels, which can reduce 2^{mU} levels into 2^{mU-1} levels as fig-(e).

The fig.3.12 is an example of $U = 4$, BPSK transmission for deriving channel gains. The first step is to pair all levels. After pairing, group levels into two groups with the restrictions, as shown in (c), it is obvious that the levels in fig-(c) are not symmetrical with their centroids. Therefore, return to step 2 and generate two groups again until it meets the symmetrical condition, as shown in fig-(d). Then, shift the corresponding groups to the original point, and search the remaining levels for deriving rest channel gains.

Compared with the previous geometric method, the novel method does not limit the size of U , even if the modulation is different. In terms of implementation, the method only needs a loop to make it work. Compared with the previous implementation complexity, which has been improved a lot. However, in terms of computational complexity, especially in step 2, it will grow exponentially with the U users,

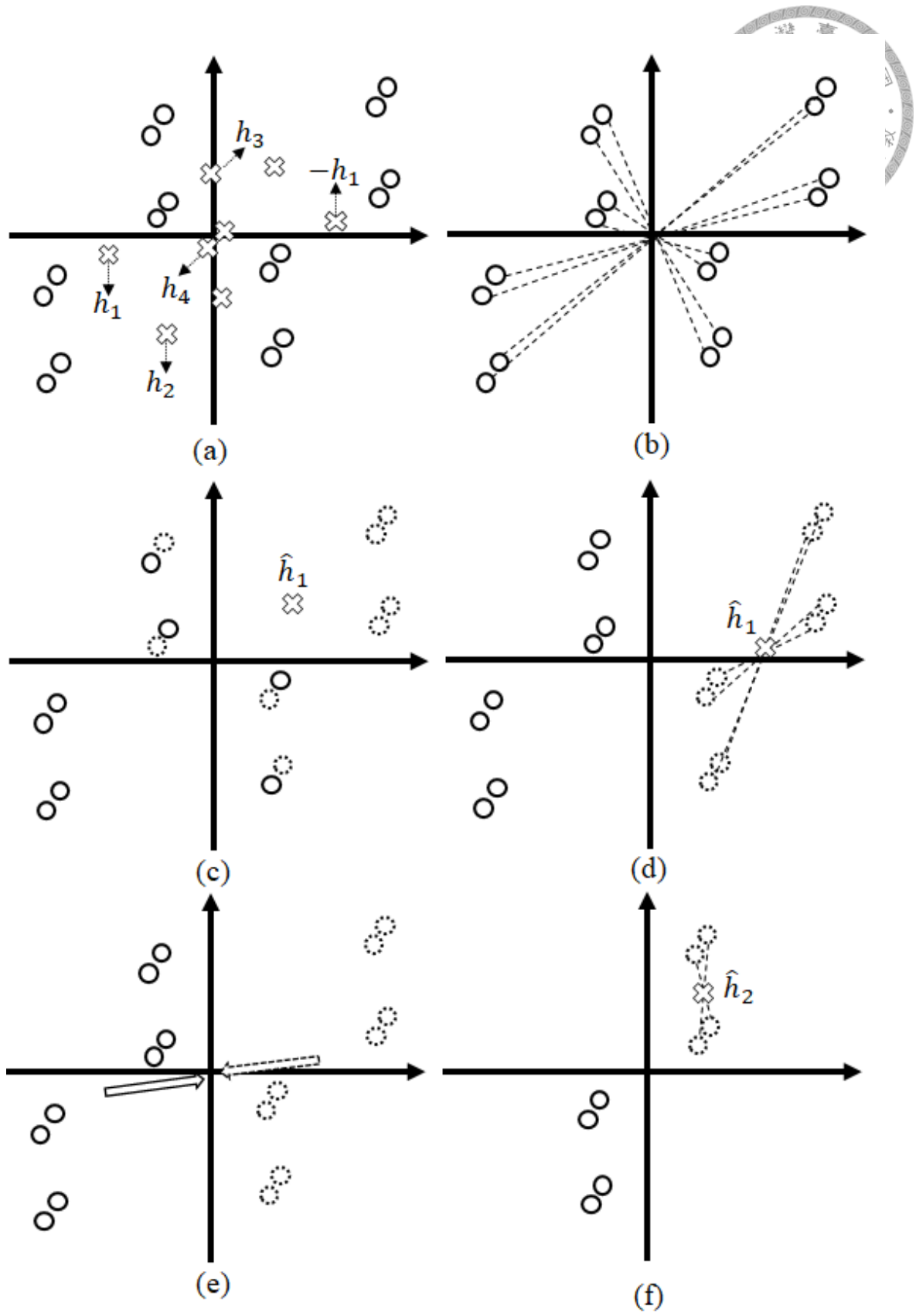


Figure 3.12: The steps of derivation of channel coefficients.

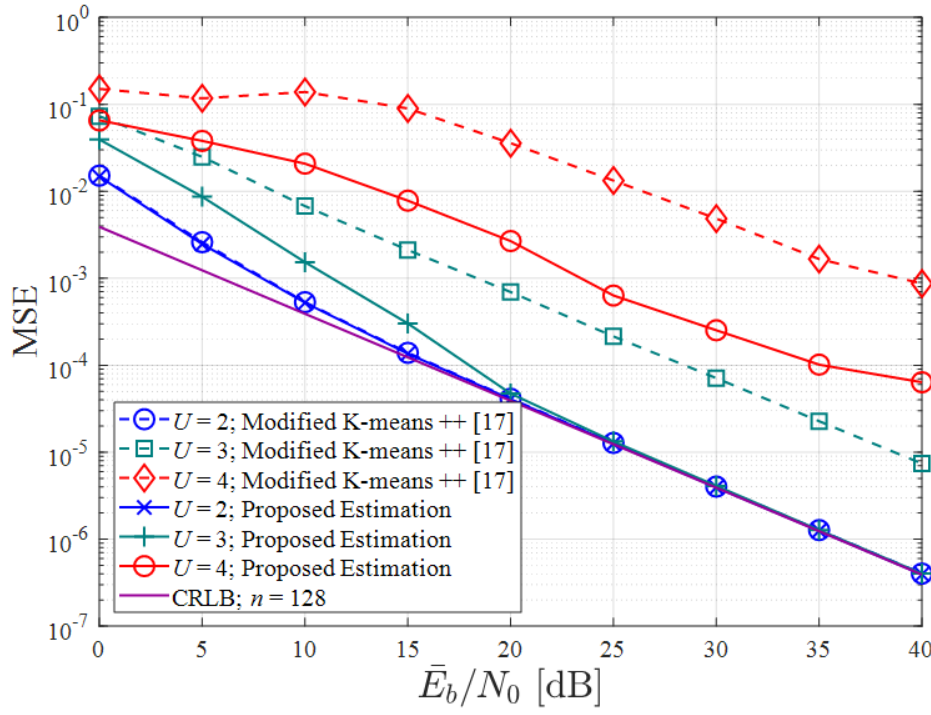


Figure 3.13: The MSE of estimated channel gain at $N_t = 128$. GMM is used.

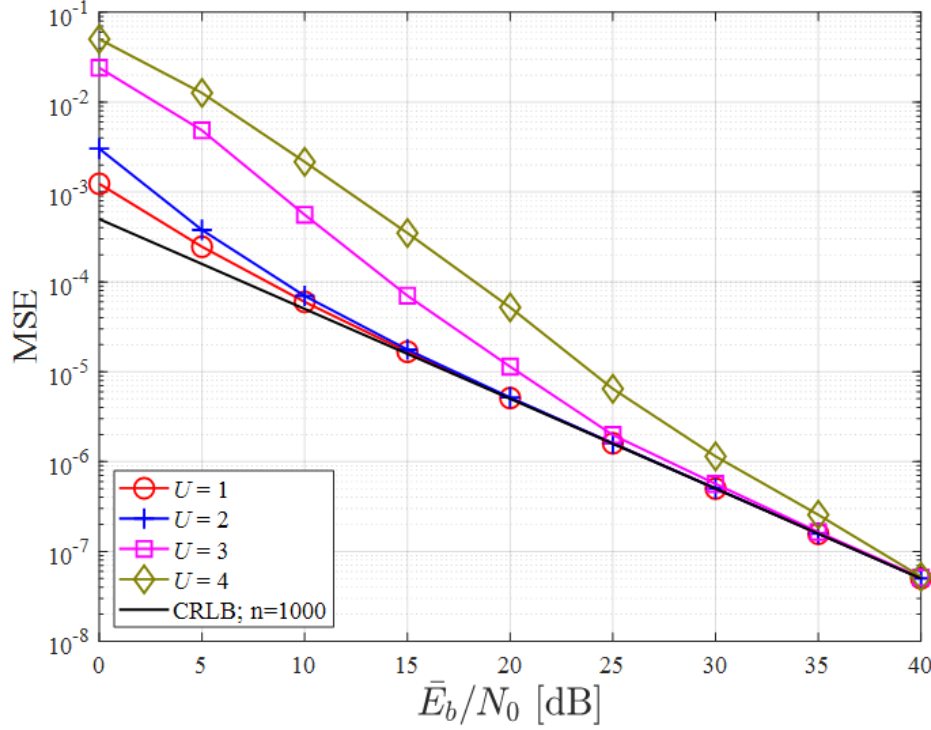


Figure 3.14: The MSE of estimated channel gain at $N_t = 1000$. GMM is used.

which might be improved in future work.

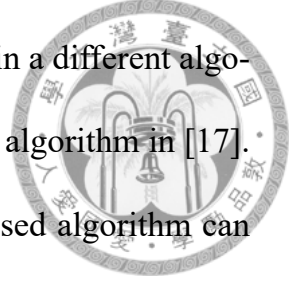


Fig.3.13 is the MSE performance of estimated channel gain in a different algorithm, where we compare the navel algorithm with the previous algorithm in [17]. Assume received samples $N_t = 128$. When $U = 3$, the proposed algorithm can approach CR bound. Compared with paper [17], the effect is excellent, and when $U = 4$, it is also good. Because there are too few samples, there is no way to ensure that all superimposed levels are reliable, so it still can not approach CRLB. When the number of samples is increased, we can solve this problem. Fig.3.14 is the MSE of $N_t = 1000$, When high SNR, $U = 4$ can still be close to CRLB. As long as the sample is enough, the proposed clustering algorithm can achieve CRLB.

3.3 Resolving Phase Ambiguity

Although we can have the estimates of channel coefficients through the clustering, there is still an uncertain phase ambiguity due to the estimation using signals that carry messages. An easy way to remove the uncertainty(ambiguity) is to differentially encode the message. However, the differential encoding results in a loss of error performance because of error propagation. Hence, several methods like differential encoder, BCJR decoder, joint decoding, and NBC are introduced to solve phase ambiguity in the thesis [4]. Here, we introduce a novel method called Q -section NBC using the polar code structure. In a single carrier system, only need NBC to resolve the phase ambiguity but can not in Q multiple carrier system since there are Q phase ambiguities in a codeword. In this section, we introduce differential encoder, NBC, Q -section NBC in sequence.

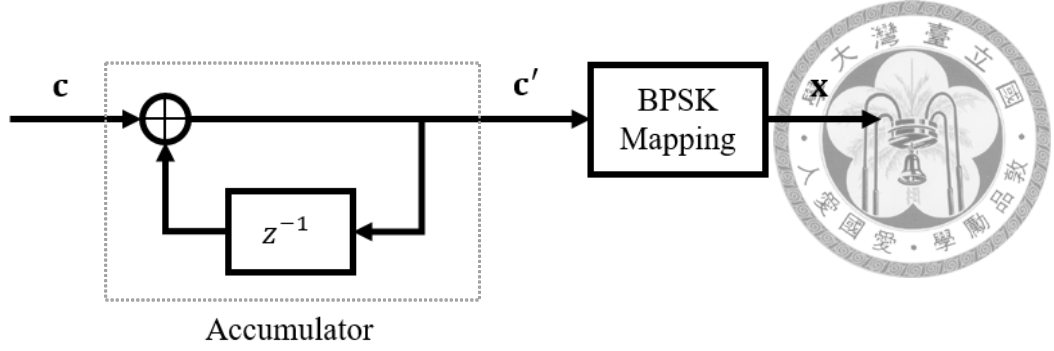


Figure 3.15: Modulation scheme for DE-BPSK transmission.

3.3.1 Differential Encoding

Differentially encoded (DE) modulation uses the difference between symbols to carry a message. In the transmission of PSK, a message is embedded in the difference of phases between two consecutive symbols, and coherent detection can still be applied, resulting in differentially encoded coherently detected PSK (DE-PSK). Therefore, the phase offset resulted from uncertain ambiguity in blind estimation will not affect the detection of data in DE-PSK transmissions. The block diagram of the DE-BPSK scheme is provided in Fig. 3.15.

3.3.2 Noncoherent Block-Coded Modulation

In [18], a novel block-coded modulation (BCM) scheme with noncoherent detection called noncoherent block-coded MPSK (NBC-MPSK) was proposed. The ambiguity can be easily removed without applying differential encoding by partitioning the code space as long as the generator matrix of a linear block code containing an all-one row.

The original code space C' is divided into $C' = \{C, C \oplus \mathbf{1}\}$, where $\mathbf{1}$ is an all-one vector, and the generator matrix \mathbf{G} of C is found by removing the all-one row in the

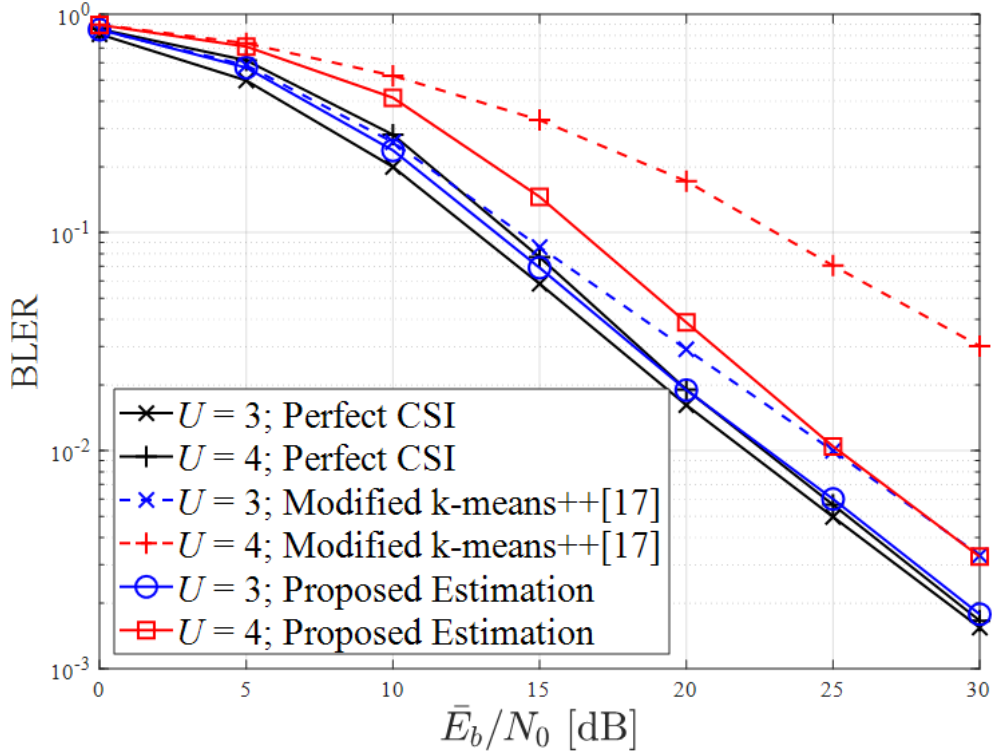


Figure 3.16: BLER performances for a (128,64) Polar-coded GDMA system over quasi-static Rayleigh flat-fading single carrier channels. NBC and GMM are both used.

generator matrix of C' . We only select the codewords from C to be transmitted, i.e., encode the message with G , resulting in a rate loss of one bit. At the receiver side, if the signal is BPSK-modulated, the codeword c' corresponding to the received signal would be c or $c \oplus \mathbf{1}$ due to an uncertain phase ambiguity of 180° where c is the transmitted codeword in C . To remove the uncertain ambiguity, we just need to detect whether the decoded codeword \tilde{c} is in C or not and the resultant codeword is

$$\hat{c} = \begin{cases} \tilde{c}, & \text{if } \tilde{c} \in C \\ \tilde{c} \oplus \mathbf{1}, & \text{if } \tilde{c} \notin C \end{cases} \quad (3.10)$$

For a higher-order modulation scheme, the multi-level coding in BCM can be decoded by multi-stage decoding, as shown in [18]. The code space identification

in (3.10) can be implemented by checking the number of satisfied parity-check equations. However, the bit errors after decoding might result in erroneous identification and redundant bit-flipping.



We consider the (128, 64) Polar-coded and the BPSK modulated GDMA system over quasi-static Rayleigh fading channels based on perfect timing. The block error rate (BLER) performances from the simulation are provided in fig. 3.16. Successive cancellation list (SCL) decoding is used where the list size is 32. Set $n = 128$. In this case, the phase ambiguity is resolved via NBC. For $U = 3$ and $U = 4$, the BLER using proposed estimation algorithm is obviously better than algorithm in [17].

3.3.3 Section Noncoherent Block-Coded Modulation

Consider a codeword transmits into a multi-carrier system and passes through a Q multi-carrier system. Since the cluster-based channel estimation still has the phase ambiguity problem in each sub-carrier channel gains, the received codeword might be split into Q segments. For example, codeword 0000 passes through a two multi-carrier system. The received codeword may be 0011, 1100, 0000, 1111, and the Q-section NBC was proposed to solve this problem.

In Q-section NBC, the original code space C' is divided into $C' = \{C, C \oplus \mathbf{v}_1, C \oplus \mathbf{v}_2, \dots, C \oplus \mathbf{v}_Q\}$, where $\mathbf{v}_i, i \in \{1, \dots, Q\}$ are the vectors for splitting the code space, and the generator matrix \mathbf{G} of C is found by removing all control vectors in the generator matrix of C' . We only select the codewords from C to be transmitted, i.e., encode the message with \mathbf{G} , which results in a rate loss of Q bit.

The control vectors v_i are designed according to multi carrier mapping as following matrix $\mathbf{V} = [\mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \ \mathbf{v}_Q]^T$.



Assume that the code length $N = 2^n$, the section $Q = 2^q$, \mathbf{v}_0 is an all-one vector. Firstly, generate the control vectors $v_i, i \in \{1, 2, \cdots, q-1\}$, and the \mathbf{v}_i is described as 3.11. Then, utensil j control vectors \mathbf{v}_i to generate remaining vectors, where $j \in \{1, 2, \cdots, q\}$ is the combination number. Finally, we would derive Q vectors, where $Q = 2^q = 1 + C_1^q + C_2^q + \cdots + C_q^q$.


$$\mathbf{v}_i = (\underbrace{1 \cdots 1}_{2^{n-i}} \underbrace{0 \cdots 0}_{2^{n-i}} \underbrace{1 \cdots 1}_{2^{n-i}} \underbrace{0 \cdots 0}_{2^{n-i}}); i \in \{1, 2, \cdots, q\}. \quad (3.11)$$

The equation.3.12 is an example for $N = 8$ and $Q = 4$. Generate \mathbf{v}_1 and \mathbf{v}_2 according to section $q = 2$, and derive $\mathbf{v}_3 = \mathbf{v}_1 \oplus \mathbf{v}_2$.

$$\mathbf{V} = \begin{bmatrix} \mathbf{v}_0 \\ \mathbf{v}_1 \\ \mathbf{v}_2 \\ \mathbf{v}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{v}_0 \\ \mathbf{v}_1 \\ \mathbf{v}_2 \\ \mathbf{v}_1 \oplus \mathbf{v}_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; Q = 4; N = 8 \quad (3.12)$$

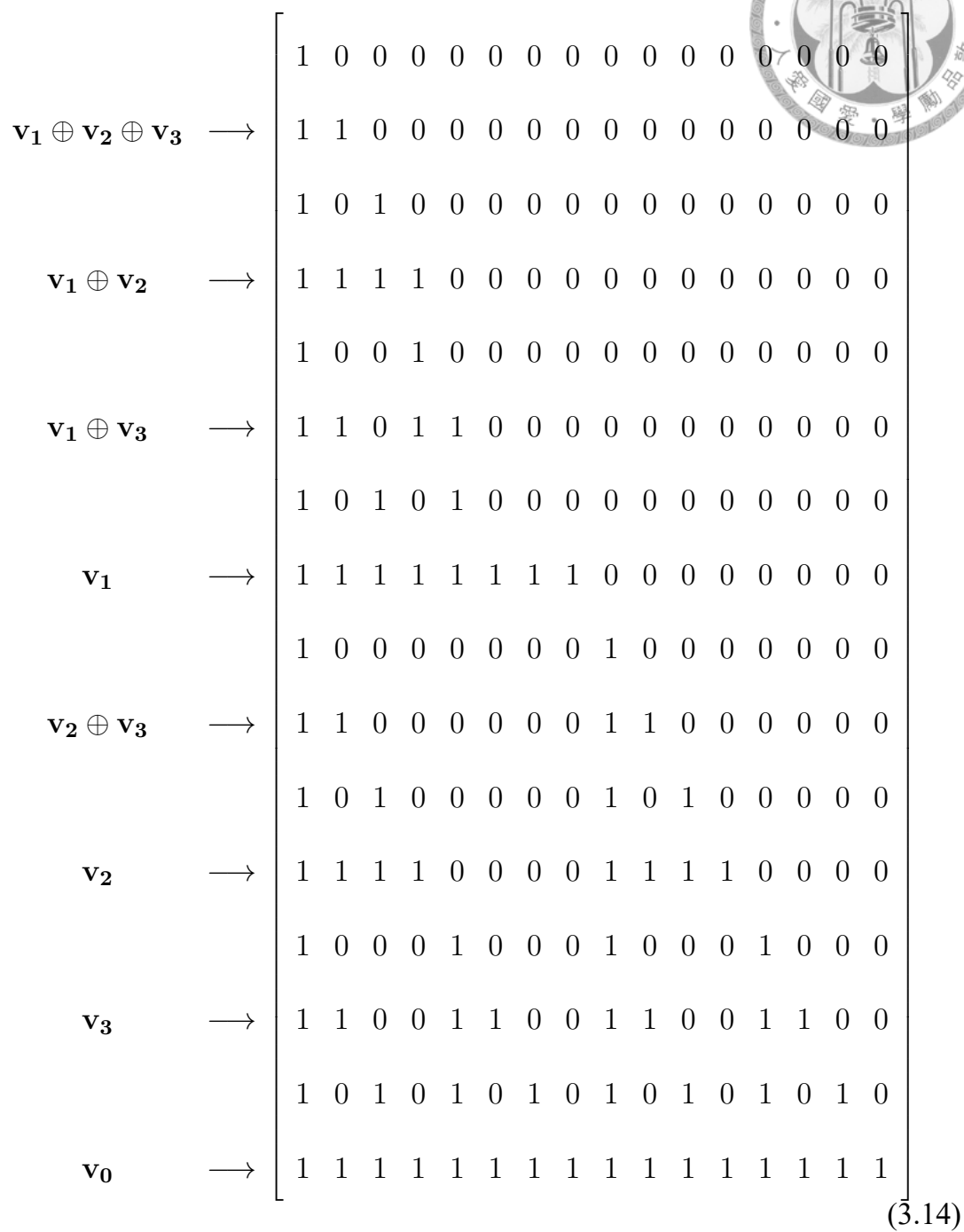
The equation.3.13 is an example for $N = 16$ and $Q = 8$. Generate $\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ according to section $q = 4$. Then, derive $\mathbf{v}_4, \mathbf{v}_5, \mathbf{v}_6$, which is combined by two vec-

tors and \mathbf{v}_7 is combined by three vectors.



$$\mathbf{V} = \begin{bmatrix} \mathbf{v}_0 \\ \mathbf{v}_1 \\ \mathbf{v}_2 \\ \mathbf{v}_3 \\ \mathbf{v}_1 \oplus \mathbf{v}_2 \\ \mathbf{v}_1 \oplus \mathbf{v}_3 \\ \mathbf{v}_2 \oplus \mathbf{v}_3 \\ \mathbf{v}_1 \oplus \mathbf{v}_2 \oplus \mathbf{v}_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; Q = 8; N = 16 \quad (3.13)$$

Matrix.3.14 is the polar code construction for $N = 16$. The section NBC control vector \mathbf{v}_i is the $\frac{N}{Q} \cdot i$ -th row of polar code structure, $i = 1, \dots, Q$. If \mathbf{v}_i is information bits for polar code, we do not put message on it. Since the ambiguity phase shift, these bits are used for control phase, e.g., they are not message bits. If \mathbf{v}_i is frozen bits for polar code, different from conventional SCL decoding, we can not feedback 0 for message passing. Since these bits are also used for control phase, it might be 0 or 1 randomly. Here, we use list to record these frozen bits for fetching up this issue. 3.17 is the BLER performance for 16 phase ambiguity polar coded OFDM system. where SCD SCL-32 is used.



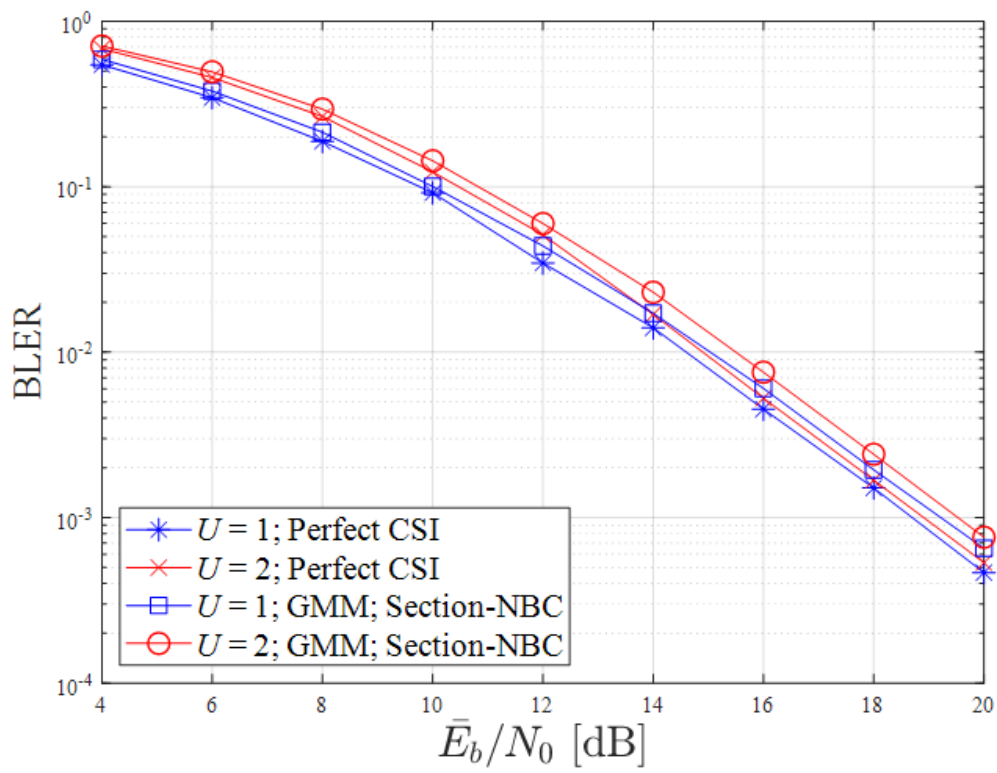


Figure 3.17: BLER performances of Polar-coded GDMA OFDM systems over quasi-static Rayleigh flat-fading channels with 16-subcarrier.



Chapter 4

A Random Access Scheme based on Gain Division Multiple Access

Recently, random access protocol has not only caused the rise of satellite communication, but also caused the application of Internet of things (IOT) and machine to machine. In this chapter, we will implement a random access scheme based on gain division multiple access. Assumed that the system is UL transmission, and numerous devices use the same channel resources to transfer in a common evolved NodeB (eNB). Unlike ordinary random access, when a packet collision occurs, the eNB does not directly feedback HARQ (hybrid automatic repeat request), but multiuser detection (MUD) using the concept of GDMA. Then, we will introduce RACH and Grant to discuss various situations in our system further. In the second section, we will present the RA system more clearly and derive the ALOHA system's throughput. In the third section, we will introduce the random access orthogonal-frequency division multiplexing gain division multiple access (RA-OFDM-GDMA) system. Discuss the different cases of the MUD based on

the variation of RACH and Grant. Then, add a preamble sequence to execute synchronization and hybrid channel gain estimation.



4.1 Introduction of RACH and Grant

RACH stands for Random access channel, which is the first message from UE to eNB when power on it. RACH is used in many places, e.g., achieve uplink synchronization between user equipment(UE) and eNB, obtain the resource for RRC connection request, etc. From the perspective of eNB, the UE signal is almost in a random fashion, e.g., casual time, random frequency, arbitrary identification.

Before UE decides to send RACH signals (RACH preamble), many preconditions are called “Power-On to PRACH” . The steps of power-on to PRACH are described as :

Step 1) UE is off.

Step 2) Power On UE.

Step 3) UE tune to a specific frequency.

Step 4) Time and Frame synchronization (PSS SSS decode).

Step 5) Physical Cell ID(PCI) , PBCH detection.

Step 6) Initial RACH process.

, where Primary Sync Sequence(PSS) provides Radio Frame Boundary using preamble sequence and Secondary Sync Sequence (SSS) provides Subframe Boundary using correlation windows.

There are two type of RACH process : contention-based and contention-free.

The steps of Contention-base RACH in 5G NR are summarized below.



Step 1) Preamble transmission : The preamble is assigned by the UE. The UE selects one of the 64 preambles from the system information provided by the eNB to compete (Msg1).

Step 2) Random Access Request : If a collision occurs, perform a back off. If successful, the eNB knows the UE information.

Step 3) Radio Resource Control Connection Request : Reply Random Access response to UE, including Timing Advance and UL grant.

Step 4) RRC Connection Setup : After receiving the RAR, the UE will upload the ID with the UE information in the UL grant. If no other UE uses the same preamble in Msg1, the UE will compete for the channel resources of the RAP.

The steps of Contention-free are summarized below.

Step 1) Preamble Allocation : The preamble is assigned by the eNB.

Step 2) Preamble Transmission : The eNB knows the UE information.

Step 3) 5) Radio Resource Control Connection Request : Reply Random Access response to UE, including Timing Advance and UL grant.

The paper [1] classified that RACH-based Grant-base, RACH-based Grant-free, RACH-free Grant-free system, where Grant means whether the number of users

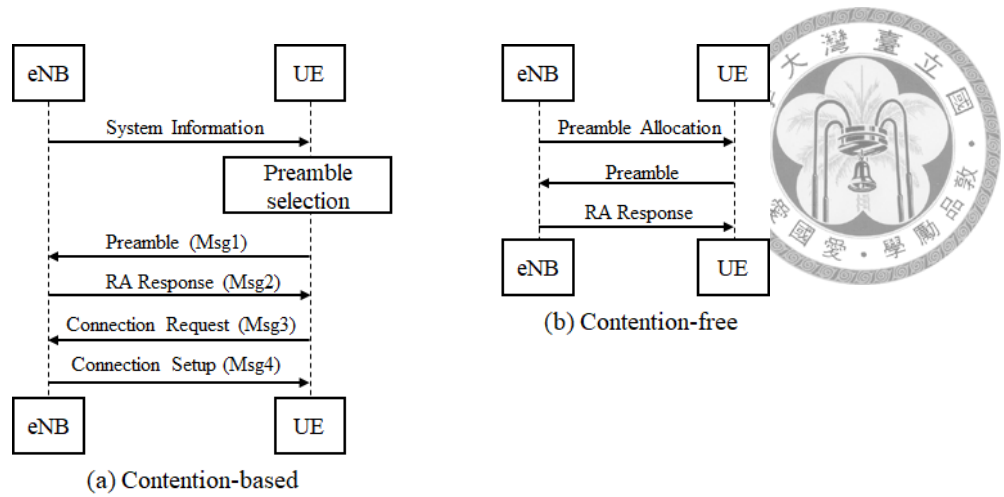


Figure 4.1: Two type of RACH process.

choosing a particular frequency sub-band is random. Users transmit data on an “arrive and go” basis (grant-free), or the receiver already allocates some specific number of users a particular subband. The transmission is scheduling(grant-based), e.g., users are aligned/connected in typical cellular communication.

4.2 Introduction of Random Access System

In this section, we consider a multiuser communication system in which users transmit information in packets with common channel resource. In contrast to the CDMA or TDMA, the transmission would not be scheduled in advance. In conventional MA, the channel resource would be wasted during the period that device is

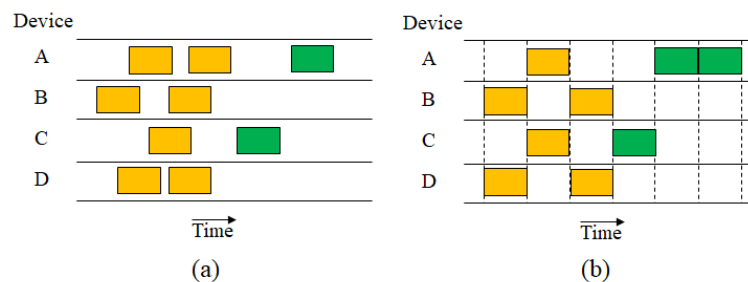


Figure 4.2: (a) An unslotted ALOHA system. (b) A slotted ALOHA system.

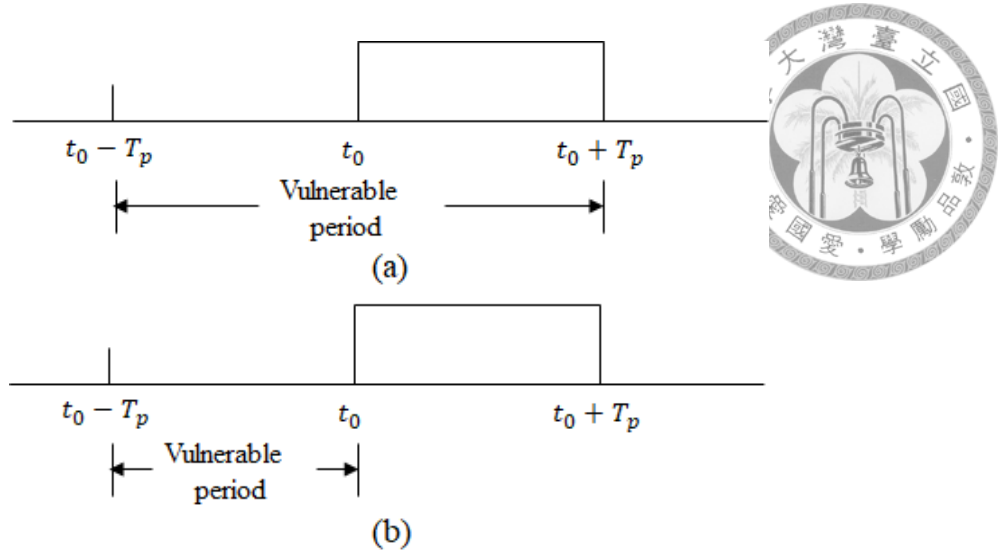


Figure 4.3: (a) The vulnerable period for unslotted system. (b) The vulnerable period for slotted system.

granted to access, but it has nothing to say. If the number of device is massive, these MA will waste more resources. However, this is just a situation where random access techniques tend to become efficient. The access methods described below are basically random because packets are generated according to some statistical models. Users access the channel when they have one or more packets overlap in time, they collide, and, hence, a conflict results. If more than one frame are transmitted, they interfere with each other and are lost. If ACK not received within the timeout, then a user picks random backoff time. An ALOHA scheme [19] was originally analyzed for random access system at the University of Hawaii in 1970, which provides access to a common communication channel from multiple independent packet transmitters by the simplest of all mechanisms. The ALOHA system can be classified into two types in fig by slotted(RACH).4.2.

We assume that the start time of packets that are transmitted is a Poisson point process having an average rate of λ packets/s at the receiver. Let T_s denote the time

duration of a packet. Then, the normalized channel traffic G , is defined as $G = \lambda T_s$ packets/duration. For a time duration, the probability of there being k transmission

For the unslotted ALOHA system, the packets are transmitted at any arbitrary time, e.g., a user transmits whenever it has data to transmit. A successful transmission probability is a probability that there is no additional packet transmission in the vulnerable period. The vulnerable period of unslotted is $2T_p$, as shown in fig4.3. If there are two or more packets arrived at the vulnerable period, the packets will collide. For the slotted ALOHA system, the packets are transmitted in the time slots, which can also be regarded as the RACH-based system, which vulnerable period is T_s . The throughput of the two systems are described as:

The throughput of unslotted ALOHA system:

$$\begin{aligned}
 S &= G \cdot Pr\{\text{no collision}\} \\
 &= G \cdot Pr\{0 \text{ transmissions in } 2 T_p\} \\
 &= G \cdot \frac{2G^0}{0! \cdot e^{-2G}} \\
 &= G \cdot e^{-2G}
 \end{aligned} \tag{4.1}$$

The throughput of slotted ALOHA system:

$$\begin{aligned}
 S &= G \cdot Pr\{\text{no collision}\} \\
 &= G \cdot Pr\{0 \text{ transmissions in } 1 T_p\} \\
 &= G \cdot \frac{G^0}{0! \cdot e^{-G}} \\
 &= G \cdot e^{-G}
 \end{aligned} \tag{4.2}$$

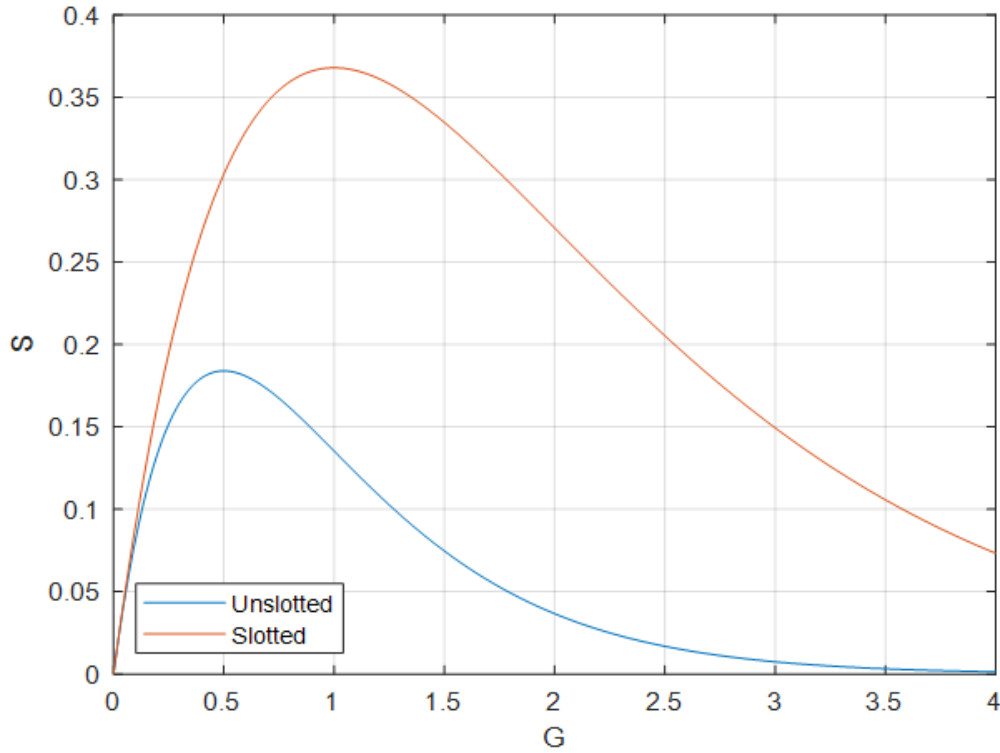


Figure 4.4: The throughput of ALOHA system.

Fig. 4.4 is the throughput of ALOHA system. For unslotted system, $G = 0.5$ has the largest throughput $S \approx 0.1839$, and for slotted system, $G = 1$ has the largest throughput $S \approx 0.3679$.

Based on the conventional ALOHA scheme, several random access protocols were developed [20] [21] [22] [23]. Most of them used more channel resources to avoid collisions, e.g., Spread-ALOHA will increase the spectrum usages, carrier sensing multiple access (CSMA) will increase the carrier usages, etc. Some researches are to resolve collision by contention tree or contention stack based on ALOHA re-transmission. However, there are more device connections, fewer channel resources, and low latency in modern communication systems. We introduced a novel RA scheme to resolve the collisions without additional channel

resources, which is a similar GDMA concept.



4.3 System Model

4.3.1 OFDM-GDMA system

The proposed scheme can be applied to an orthogonal-frequency division multiplexing (OFDM) system. For each $i \in \{0, 1, \dots, N/(mQ) - 1\}$, the subblock $[x_u(iQ+1), \dots, x_u((i+1)Q)]$ is input into the Q -point inverse fast Fourier transform (IFFT) device to obtain the time domain signal. A cyclic prefix (CP) is inserted so as to avoid inter-symbol interference (ISI) and inter-carrier interference (ICI). On the receiver side, following CP removal and the Q -point FFT process, the frequency domain signal is then sent to the multi-level receiver. Let $H_u(j)$ be the channel gain between user- u and the receiver over the j -th sub-channel for $j \in \{1, \dots, Q\}$. For the frequency domain signal, the component in the j -th sub-channel is

$$r(iQ + j) = \sum_{u=1}^U H_u(j)x_u(iQ + j) + W(iQ + j), \quad (4.3)$$

where $W(iQ + j)$ is the AWGN. The clustering-based channel estimation is performed for each of the sub-channels separately.

4.3.2 Doppler-effect and unknown timing drift

We now consider applying the proposed scheme to time-variant channels. The basic settings are those depicted in Fig. 4.5 with $U = 2$ and GMM. To trace the traversal of channel gains which vary with time, we employ a sliding-window ap-

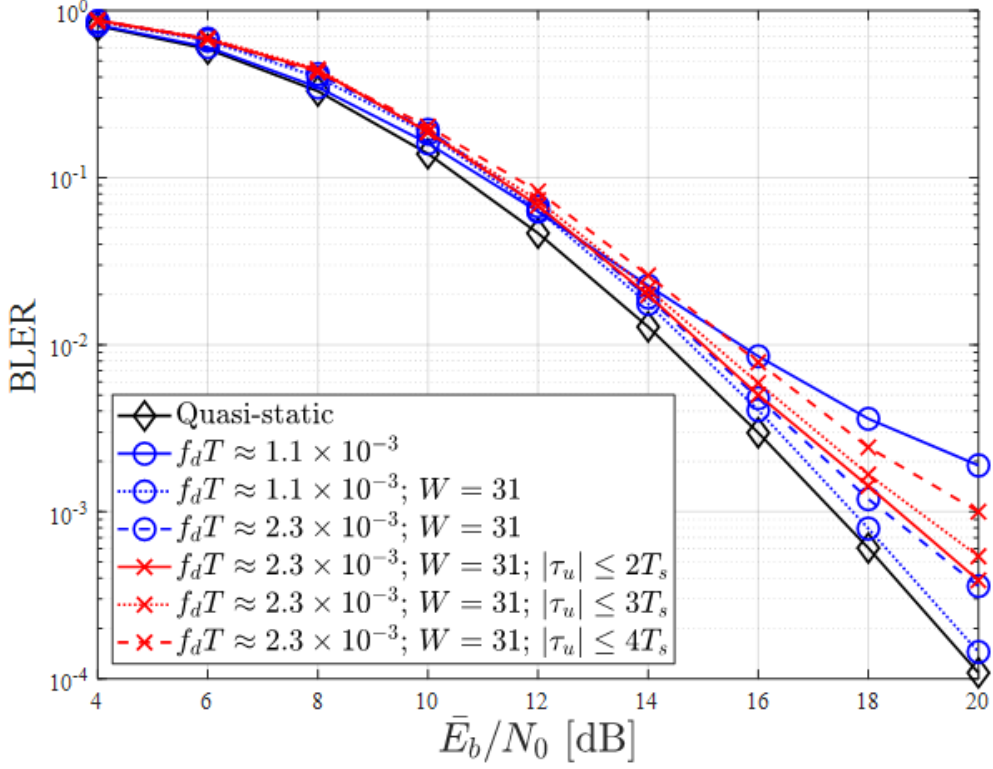


Figure 4.5: BLER performances for a (1008,504) LDPC-coded OFDM-GDMA system where $U = 2$ users over frequency-selective Rayleigh fading channels.

proach in the estimation process and the size of the window is set to W . Consider the i -th windowing operation for $i \in \{0, 1, \dots, n - W\}$. The estimator only takes the received symbols $[r(iQ + j), \dots, r((i + W - 1)Q + j)]$ as input samples for the clustering in order to estimate the channel gain $\hat{h}_{i,j}$ at symbol time $i + \lfloor W/2 \rfloor$ over the j -th sub-channel. The estimate $\hat{h}_{i,j}$ at $i = 0, 1, \dots, \lfloor W/2 \rfloor - 1$ is set to $\hat{h}_{\lfloor W/2 \rfloor, j}$. Moreover, $\hat{h}_{i,j}$ at $i \in \{n - \lfloor W/2 \rfloor - 1, \dots, n - W\}$ is set to $\hat{h}_{n - \lfloor W/2 \rfloor, j}$. In the simulations shown in Fig. 4.5, the maximum Doppler frequency f_d is normalized using the period of an OFDM symbol T which does not include CP. For $f_d T \approx 1.1 \times 10^{-3}$, the proposed estimation scheme using the sliding window procedure with $W = 31$ can achieve a BLER very close to that of the quasi-static case. If windowing is not

used, the BLER performance is significantly degraded for $f_d T \approx 1.1 \times 10^{-3}$ as indicated by the solid line with square symbols.

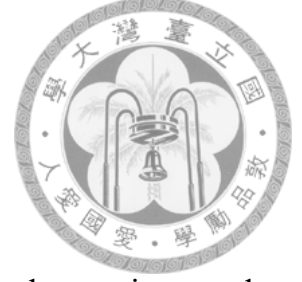


We further consider an OFDM-GDMA system that has imperfect synchronization. The simulation results are also provided in Fig. 4.5. Assume that user u out of the $U=2$ users transmits a signal that is independent of the other user and has a uniformly distributed timing offset τ_u which is unknown to the receiver. Both the transmitting and receiving filters are square root raised cosine filters with a roll-off factor $\beta = 0.22$. In each transmitter, the subblock $[x_u(iQ + 1), \dots, x_u((i + 1)Q)]$ is up-sampled by a factor of 4, a $4Q$ -point IFFT is used, and a cyclic suffix (CS) is added to prevent the interference from the other user. By setting the CP length to $4T_s$ and the CS length to $4T_s$, we see that for $|\tau_u| \leq 2T_s$, there is no performance loss when compared to the $|\tau_u| = 0$ case, where $T_s = \frac{T}{Q} = \frac{T}{16}$.

4.3.3 Slotted RA OFDM-GDMA system

In this system, all devices are synchronized before transmitting packets, e.g., RACH based system, and the length of each slot is OFDM symbol length. For the grant-based system, the receiver known the number of users. Multiple users can be detected by conventional GDMA. We can design a tolerance range, which can be designed by different cyclic prefix (CP) or cyclic suffix (CS) as section.4.3.2. If the unknown drift of each device time is within the tolerance range, it can be solved. However, the wider tolerance range will waste more transmission. For the Grant-free system, the receiver does not know the number of users, and it can use the parity check of the LDPC code to determine which U is most suitable for the

current case as the paper [17].



4.3.4 Unslotted RA OFDM-GDMA system

In this system, each device has not been synchronized with the receiver, and may be transmitted at any time, e.g., a RACH-free system. Unlike Slotted GDMA OFDM, we define the unit of GDMA is symbol length because the messages between each symbol are transmitted in parallel by different sub-carriers. If an OFDM symbol collides and the collision's time drift is within the tolerance range, it can be successfully resolved in this symbol. If the collision's time drift is outside the range, inter symbol interference (ISI) will be generated. Assume that τ_1 is the transmission time of user-1 and τ_2 is for user-2, and divide it into two case for discussions as fig.4.6 from the perspective of user-1.

Fig.4.6-(a) is the condition for $i \cdot N_s - CP < \tau_2 - \tau_1 < i \cdot N_s$, where $i = 0, 1, \dots, N/mQ$, $N_s = CP + Q$. For $j = 0, 1, \dots, i-1$, derive the user-1 messages directly by using APP detector, since the symbol of user-1 does not collide. For $j = i, \dots, N/mQ$, set $U = 2$ to detect user-1 and user-2 messages by using GDMA

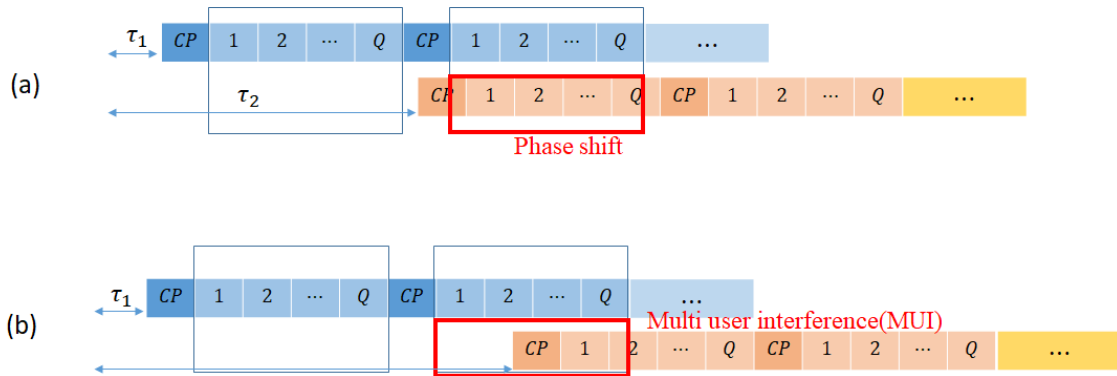


Figure 4.6: (a) case 1 (b) case 2

detector, since the truncated OFDM symbols of user-2 still have complete symbol information, which is only a phase shift in frequency domain.

Fig4.6-(b) is the condition on $i \cdot N_s < \tau_2 - \tau_1 < (i + 1) \cdot N_s + \text{CP}$, where $i = 0, 1, \dots, N/mQ$. If $j = 0, \dots, i-1$, derive the user-1 messages directly by using APP detector, since the symbol of user-1 does not collide. If $j = i, \dots, N/mQ$, set user $U = 2$ for detecting two user. However, the truncated symbol only contain complete user-1 OFDM symbol, e.g., the user-2 symbol is incomplete that will generate inter symbol interference(ISI). Hence, we also need consider the ISI variance for APP detector. σ_w^2 must be adjusted for detecting APPs more accurately and be expressed as $\sigma_w^2 = \text{ISI variance} + \text{AWGN variance}$. Since the ISI noise has different effect for different SNR received signal, we adjust σ_w^2 with SNR, which is expressed as $\sigma_w^2 = (\sigma_{AWGN} + \mu \cdot \sqrt{i})^2$, where i is the SNR(dB). In practical situation, we do not known the SNR(dB) information beforehand. In future work, we may use the preamble to estimate the SNR(dB) information for accurately estimating MUI.

4.4 Preamble Synchronization

4.4.1 Preamble

There are some widely used preamble sequences, z adoff-chu (ZC) sequence, Pseudorandom (PN) sequence, Gold sequence, etc. In this thesis, we used the z adoff-chu sequence [24]. Zadoff-Chu sequences are used in the 3GPP NR in Primary Synchronization Signal(PSS), random access preamble(PRACH), etc.

The zadoff-chu sequence at position n is given by

$$x_u(n) = e^{-j\pi un(n+1)}, 0 \leq n \leq N_{ZC} - 1 \quad (4.4)$$

$$u \in \{1, \dots, N_{ZC} - 1\}, \gcd(N_{ZC}, u) = 1 \quad (4.5)$$



, where u is the root index.

Zadoff-Chu sequences exhibit following useful property:

- (1) The amplitude of the sequences is constant, and the sequence is orthogonal to their cyclic shift.
- (2) They are periodic with period N_{ZC} if N_{ZC} is odd. $x_u(n + N_{ZC}) = x_u(n)$.
- (3) If N_{zc} is prime, the Discrete Fourier Transform of a Zadoff-Chu sequence is another Zadoff-Chu sequence conjugated. $X_u[k] = x_u^*(\widetilde{u}_k)X_u[0]$, where \widetilde{u}_k is the multiplicative inverse of u modulo N_{ZC} .
- (4) The auto correlation of a Zadoff-Chu sequence with a cyclically shifted version of itself is zero.
- (5) The cross-correlation between two prime length Zadoff-Chu sequences, i.e. different values of $u, u = u_1, u = u_2$ is constant $\frac{1}{\sqrt{N_{ZC}}}$, provided that $u_1 - u_2$ is the relatively prime to N_{ZC} .

4.4.2 Synchronization

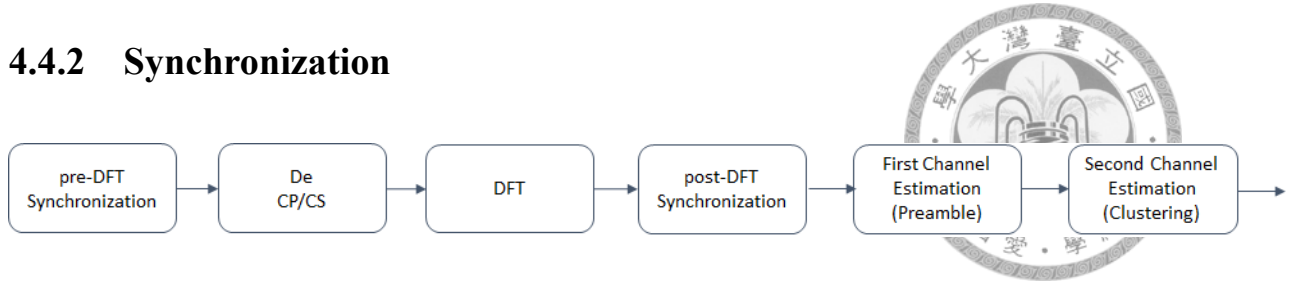


Figure 4.7: Synchronization and channel estimation process

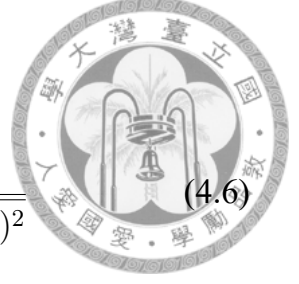
Fig.4.7 is the flow chart of synchronization and channel estimation, where pre-DFT synchronization is the coarse-timing estimation in the time domain, and post-DFT is the fine-timing estimation in the frequency domain. The channel estimation can also be divided into two steps, where first channel estimation only uses preamble to derive coarse gains. The second channel estimation use clustering to acquire fine gains based on the first channel estimation result.

Coarse-timing estimate

Coarse synchronization can be categorized into autocorrelation method and cross-correlation method. For autocorrelation method, the transmitter adopts a repetitive structure. It only needs to calculate the autocorrelation of the received signal for estimating timing offset and frequency offset. For crosscorrelation method, the resultant approaches take advantage of cross-correlating received signals with receiver-generated preamble symbols. Derive estimated rough time offset by choosing the mean of timing metric trajectory peak.

In this thesis, we use the crosscorrelation method for implementation. Define the timing metric $c[n]$, and the crosscorrelation is expressed as

$$c[n] = \frac{\sum_{i=0}^{N_{ZC}} (x_u[i] - \bar{x}_u)^* (y[n+i] - \bar{y})}{\sqrt{\sum_{i=0}^{N_{ZC}} (x_u[i] - \bar{x}_u)^2 \sum_{i=0}^{N_{ZC}} (y[n+i] - \bar{y})^2}} \quad (4.6)$$



, where $y[i]$ is the received sample from time i , \bar{y} is the mean of y , \bar{x}_u is the mean of x_u , timing metric $|c[n]| < 1$. If the timing metric is greater than threshold μ at time $\hat{\theta}$, and estimate Time offset is $\hat{\theta}$.

In an additive white Gaussian noise (AWGN) channel, the coarse estimated time offset is at the exact timing point, but in multipath channels it would be shifted (delayed) due to the channel dispersion.

Fine-timing estimate

For deriving exact timing offset from a multipath channel, the fine time offset is necessary. Even under noiseless conditions, the timing estimate would be varying according to the time-varying channel. Accordingly, at different channel taps, the channel estimates would be delayed due to timing offset errors. If the delay in channel estimate can be found, the effect of the time-varying channel response on the timing estimation can be removed by simply delaying the coarse-timing estimation. In other words, the coarse timing estimate can be fine-tuned by adding the delay of the first actual channel tap. Define the fine time offset $\hat{\epsilon} = \hat{\theta} - \theta$, where θ is the exact timing offset. For finding the delay of the first actual channel tap [25], firstly, the strongest tap gain estimate \hat{h}_{max} is found as $\hat{h}_{max} = \max_l \{E_h(l) : l = 0, 1, \dots, K^\dagger - K'\}$. Then, the delay estimate of the first actual channel tap $\hat{\epsilon}$ is given by



$$\hat{\epsilon} = \arg \max_l \{E_h(l) : l = 0, 1, \dots, K^\dagger - K'\} \quad (4.7)$$

where

$$E_h(j) = \begin{cases} \sum_{k=0}^{K'-1} |\hat{h}_{l+k}|^2 & , \text{if } |\hat{h}_l| > \mu \cdot |\hat{h}_{\max}| \\ 0 & , \text{otherwise.} \end{cases} \quad (4.8)$$

$E_h(l)$ is the channel energy estimation contained in a length- K' window starts from the tap l with the condition that the channel energy estimate of the tap should be greater than threshold μ .

4.5 Hybrid Channel Estimation

Once obtaining the fine time offset $\hat{\epsilon}$, we can find the first tap of time-domain channel gains $\hat{h}[k]$ and derive the frequency domain channel gains $\hat{H}[k]$ directly by equ.4.9.

$$\hat{H}[k] = \frac{r[k]}{S[k]} e^{\frac{j2\pi k\hat{\epsilon}}{N}}, k = 1, \dots, N_{ZC} \quad (4.9)$$

$$\hat{h}[k] = \text{IDFT}_{N_{ZC}}(\hat{H}[k]), k = 1, \dots, N_{ZC} \quad (4.10)$$

where N_{ZC} is the preamble length and $S[k]; k = 1, \dots, N_{ZC}$ are the frequency domain preambles.

If preamble length N_{ZC} is different from sub-carrier size Q , the estimated chan-

nel gains $\hat{H}[k]$ are also distinguished from sub-carrier channel gains \tilde{H} which can be derived as



$$\tilde{H}[k] = \text{FFT}_Q(h[\hat{k}]), k = 1, \dots, Q \quad (4.11)$$

,where the length of $h[\hat{k}]$ is N_{ZC} . It is only need the first Q bits to derive sub-carrier channel gains \tilde{H} .

Define $\tilde{H}[k]$ as first channel estimated gains and $\check{H}[k]$ as second channel estimated gains. In chapter 3, we introduce several clustering algorithms for channel estimation. Hence, we use the clustering algorithm for estimating gains more accurately, which utensil $\tilde{H}[k]$ as the initial centroids. Fig. 4.8 is the MSE performance of first and second channel estimation, and the second estimation can achieve to CRLB, where we assume that perfect timing estimation.

4.6 Simulation

In following simulation, each packet has identical settings and contains a preamble sequence and OFDM sequence. In time domain, a packet length is expressed as : preamble length + OFDM symbol \times OFDM segment = $71 + 20 \times 63 = 1331$, where the unit is T_s . Each packet is encoded by (1008,504) LDPC code. A OFDM symbol contain 4 CP and 16 IDFT sequence. A OFDM segment is 63. A preamble sequence is added to the packet header. The preamble sequence length is 71 , which contains 4 CP, 4 CS, and 63 z adoff-chu sequence with $N_{ZC} = 63; u = 25$.

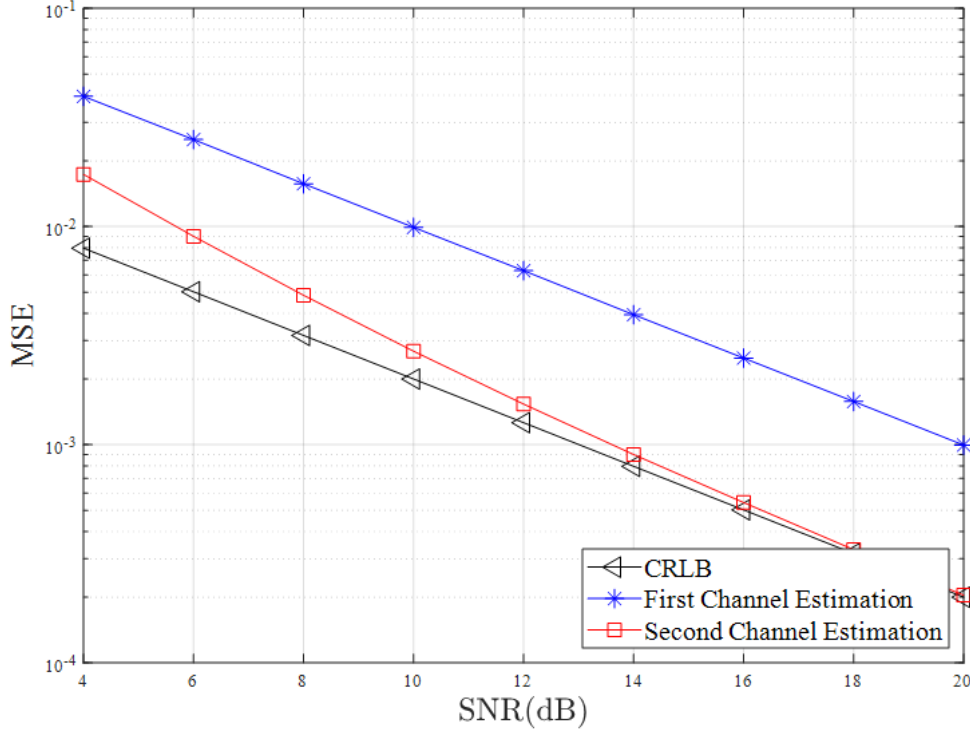


Figure 4.8: MSE comparison of first and second channel estimation

Firstly, we discuss the case that no collision occurs in fig.4.10. Define that the success rate is $\text{success} = \frac{\text{transmitted packets}}{\text{total transmitted packets}}$. We found that the gap between timing perfect and timing estimation is larger than the gap between perfect CSI and second channel estimation. In our parameter setting, the synchronization success rate is more important than channel estimation.

The situation in fig.4.11 is that two users will collide. Set τ_1 as user-1 timing offset and τ_2 as user-2 timing offset. Define that $|\tau_2 - \tau_1|$ is the uniform distribution with range from 0 to packet length. We found that the success rate based on second channel estimation will very close to perfect CSI. However, the time estimation success rate will also decrease obviously due to the preamble sequence collision.

We assume that the start time of packets transmitted is a Poisson process having

an average rate of λ packets/s at receiver in 20 SNR(dB). Let T_s denote the time duration of a packet. Then, the normalized channel traffic G , is defined as $G = \lambda \cdot T_s$ packets/duration. For a time duration, the probability of there being k transmission is $\frac{G^k e^{-G}}{k!}$. We use the exponential distribution $f(x) = Ge^{-Gx}$, $x \geq 0$ to simulate the time offset at receiver, e.g., the difference between i -packet and $i + 1$ -packet is $T_s \cdot f(x)$. To simulate precisely, we set simulation block equal to $100 \cdot T_s$, and the exponential time offset is consequent in the block. The pseudo code of Poisson process is described as

Algorithm 1 The pseudo code of Poisson process

Require: The time offset of each user is set as $TN_{u,i}$, where u -user has TN_u packets, and $i = 1, \dots, TN_u$. $TO_{u,i}$ is the u -user i -packet time offset, which is a random number in the simulation block.

Ensure: Time offset differential of same user $TO_{u,i} - TO_{u,i-1}$ must be larger than a packet duration;

```

1: Set  $T_{\text{last}} = 0$  as last packet time
2: Set  $T_{\text{packet}}$  as packet duration
3: Set  $T_{\text{block}}$  as simulation block duration
4: Set  $f(x) = Ge^{-Gx}$ ,  $x \geq 0$  as exponential distribution
5: while  $T_{\text{last}} + T_{\text{packet}} \leq T_{\text{block}}$  do
6:    $T_{\text{last}} = T_{\text{last}} + f(x)$ ;
7:   for  $u = 1; u \leq U; u++$  do
8:     if  $TN_u = 0 \parallel T_{\text{last}} - TO_{u,i} \geq T_{\text{packet}}$  then
9:        $TN_{u,TN_u} = T_{\text{last}}$ ;
10:       $TN_u = TN_u + 1$ ;
11:      break;
12:     end if
13:   end for
14: end while
15: return  $TN_{u,i}$  and  $TN_u$ ;

```

Fig.4.9 is the throughput for simulated ALOHA system by above simulator, and we can find that it is equal to theory throughput in the simulator setting.

Firstly, we found that the Multi-User Interference (MUI) noise term influence greatly in high SNR. It take MUI noise as zero in fig.4.12, and take MUI noise as

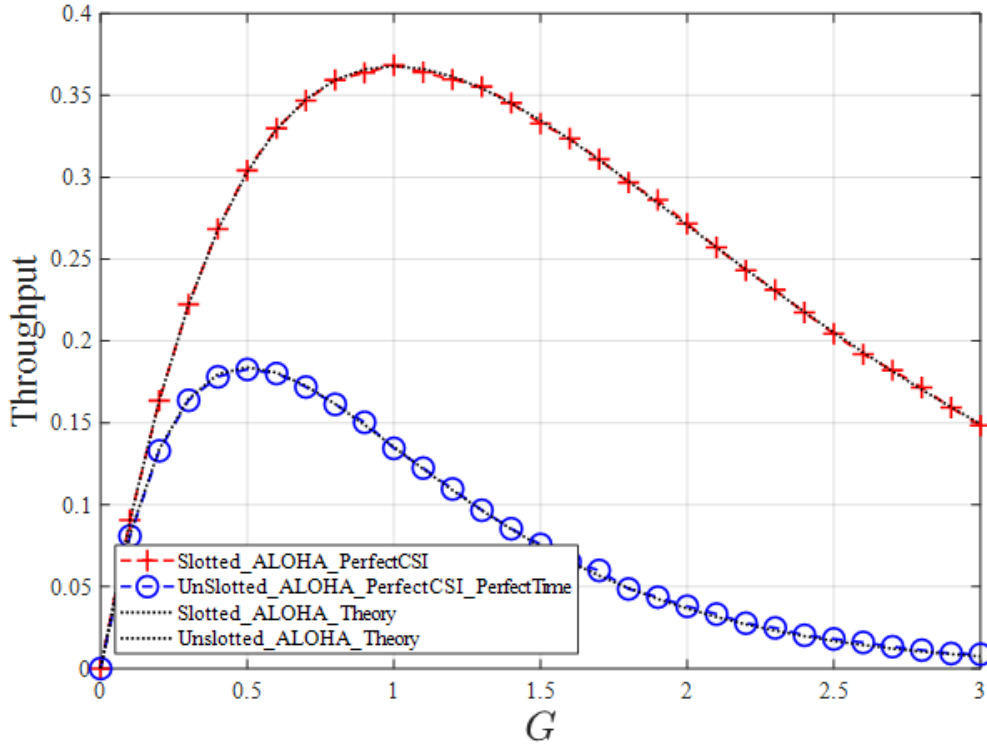


Figure 4.9: ALOHA system simulation

adapt variance in fig.4.13, where the largest throughput $S \approx 0.4492$ in fig.4.12, $S \approx 0.6088$ in fig.4.13. Compare to unslotted ALOHA, the throughput of RA-OFDM-GDMA is raise obviously. For unslotted ALOHA, $G = 0.5$ has the largest throughput $S \approx 0.1839$, and for RA-OFDM-GDMA, $G = 1.8$ has the largest throughput $S \approx 0.6088$ with perfect timing estimation and perfect CSI. For timing estimation and channel estimation case, the largest throughput $S \approx 0.4107$ at $G = 1.4$, which is still better than the slotted ALOHA theory.

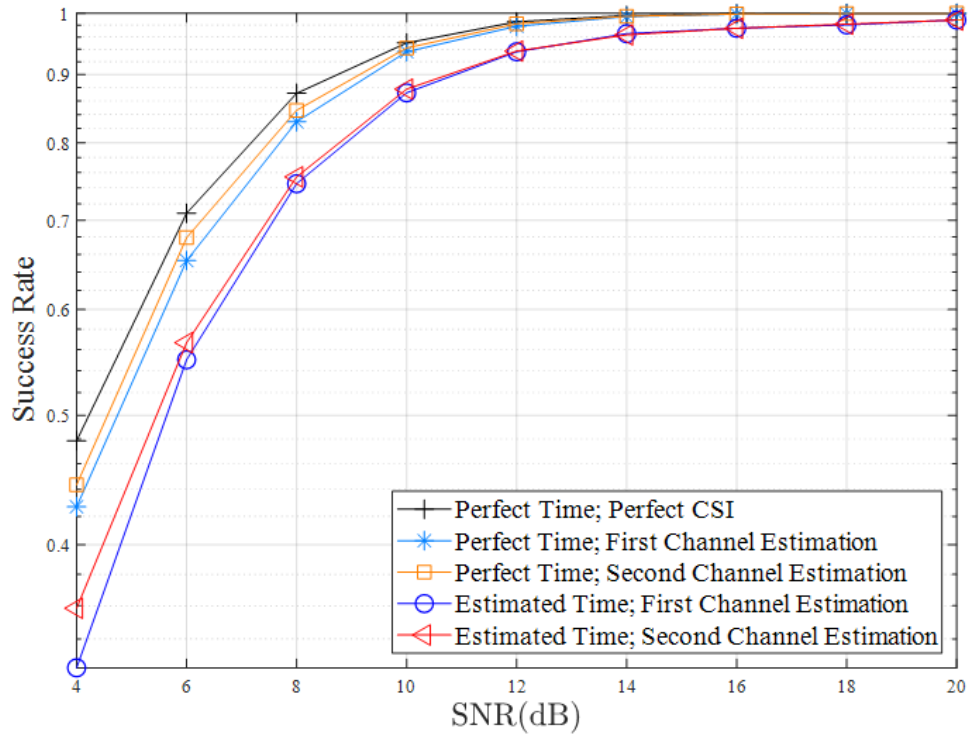


Figure 4.10: No user collision case

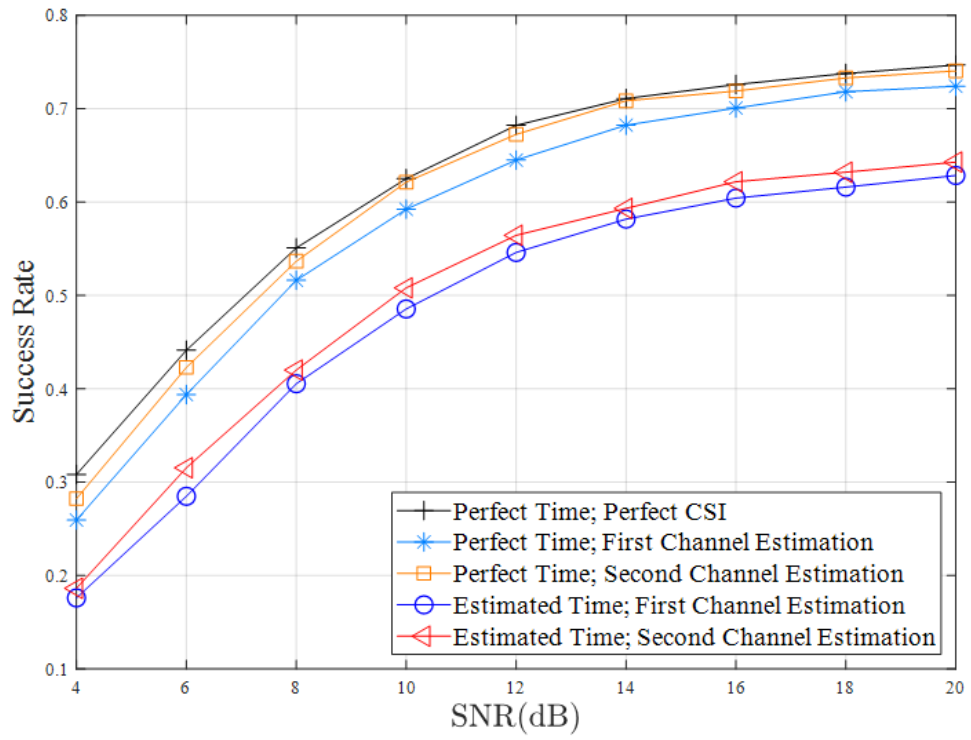


Figure 4.11: Two user collision case

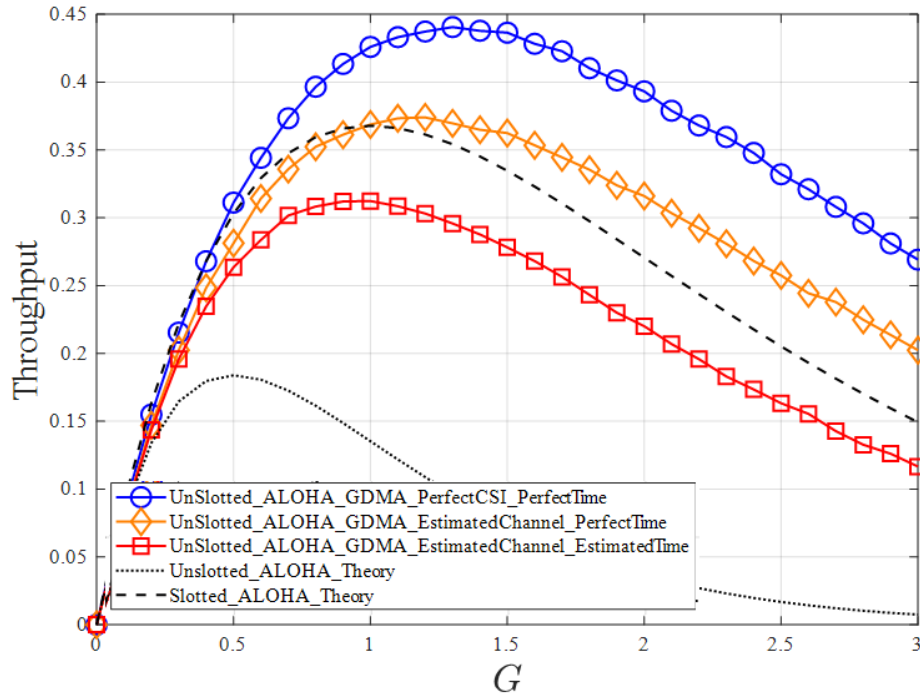
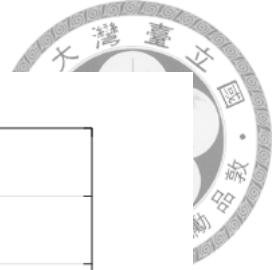


Figure 4.12: The APP detector variance without considering MUI noise term in Poisson distribution collision case

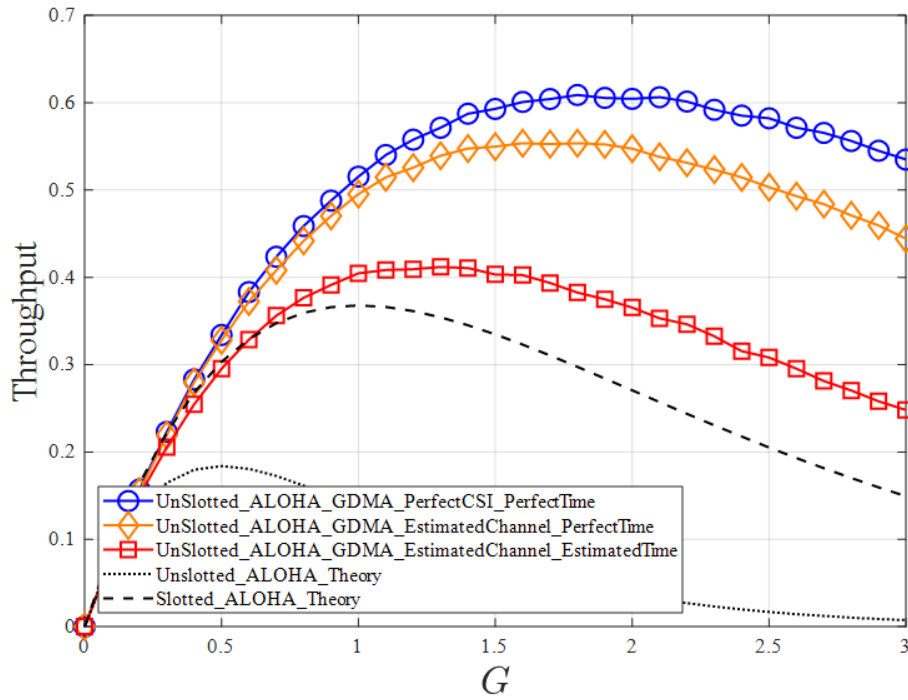


Figure 4.13: The APP detector variance without considering MUI noise term in Poisson distribution collision case



Chapter 5

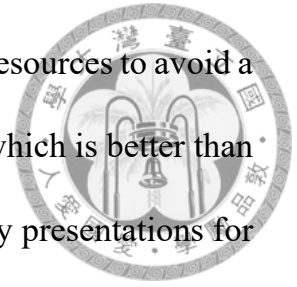
Conclusion and Future Works

In Chapter.2, we reviewed the GDMA concept, which exploited the distinct channels to separate the signals transmitted from multiple users over independent fading channels and introduced a general GDMA detector in BPSK, QPSK, and PAM for implementing practically. We also introduced a Polar-coded joint channel decoder that achieved better BLER performance, which can also be applied to the physical network coding area. However, the path metrics operation of the J-SCL decoder will grow enormously. In future work, the metric reduction is indeed.

In Chapter.3, The proposed cluster-based scheme for blind channel estimation can be applied to the transmissions over scalar channels, and high spectral efficiency can be obtained since no additional pilot signal is needed for channel estimation. We further proposed a general clustering algorithm and deriving channel gain algorithm for cluster-based channel estimation, which can achieve CRLB for any user U . However, the operation of deriving channel gain is also increased exponentially. For practical implantation, the algorithm reduction is also indeed.

In Chapter.4, we utensil the GDMA concept to detect collision RA packets in an

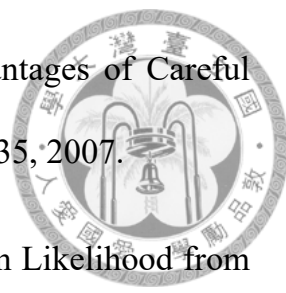
OFDM system, which does not require additional user-specific resources to avoid a collision. The main research is unslotted RA-OFDM-GDMA, which is better than a blindly random access (ALOHA) system. However, the theory presentations for throughput and ISI noise are still weak. In future work, the theory derivation is indeed, and we can further apply it to the Doppler channel system for discussing the channel estimation effect.





Bibliography


- [1] M. B. Shahab, R. Abbas, M. Shirvanimoghaddam, and S. J. Johnson, "Grant-free non-orthogonal multiple access for iot: A survey," *arXiv preprint arXiv:1910.06529*, 2019.
- [2] B.-H. Chang, "Multiple Access for Transmissions Over Independent Fading Channels," Master's thesis, National Taiwan University, 2018.
- [3] S. Zhang, S. Liew, and P. Lam, "Hot Topic: Physical Layer Network Coding," in *Proc. International Conference on Mobile Computing and Networking (MobiCom)*, p. 358–365, 2006.
- [4] Y.-T. Hsu, "Analysis and Design of Gain-Division Multiple Access," Master's thesis, National Taiwan University, 2019.
- [5] D. Wübben and Y. Lang, "Generalized Sum-Product Algorithm for Joint Channel Decoding and Physical-Layer Network Coding in Two-Way Relay Systems," in *IEEE Proc. Global Communications Conference (GLOBECOM)*, Feb. 2010.
- [6] Y. Linde, A. Buzo, and R. M. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Trans. Commun.*, vol. COM-28, pp. 84–95, Jan. 1980.

- 
- [7] D. Arthur and S. Vassilvitskii, “K-Means++: The Advantages of Careful Seeding,” in *Proc. Symp. Discrete Algorithms*, p. 1027–1035, 2007.
- [8] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum Likelihood from Incomplete Data via the EM Algorithm,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 84–95, 1977.
- [9] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, “Factor Graphs and the Sum-Product Algorithm,” *IEEE Trans. Inf. Theory*, vol. 47, pp. 498–519, Feb. 2001.
- [10] D. Wübben, “Joint Channel Decoding and Physical-Layer Network Coding in Two-Way QPSK Relay Systems by a Generalized Sum-Product Algorithm,” in *Proc. 7th International Symposium on Wireless Communication Systems (ISWCS)*, Sept. 2010.
- [11] E. Arıkan, “Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels,” *IEEE Trans. Inf. Theory*, vol. 55, pp. 3051–3073, July 2009.
- [12] I. Tal and A. Vardy, “List Decoding of Polar Codes,” *IEEE Trans. Inf. Theory*, vol. 61, pp. 2213–2226, May 2015.
- [13] E. Arıkan, “Polar codes: A pipelined implementation,” in *Proc. 4th Int. Symp. on Broad. Commun. ISBC 2010*, pp. 11–14, 2010.
- [14] S.-K. Lee, Y.-T. Hsu, B.-H. Chang, M.-C. Lin, and H.-C. Lee, “A curve fitting method to construct polar coded ofdm systems with channel side informa-

tion for the transmitter,” in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, pp. 1–5, IEEE, 2019.



- [15] K. Niu and K. Chen, “Crc-aided decoding of polar codes,” *IEEE Communications Letters*, vol. 16, no. 10, pp. 1668–1671, 2012.
- [16] S. P. Lloyd, “Least Squares Quantization in PCM,” *IEEE Trans. Inf. Theory*, vol. IT-28, pp. 129–136, Mar. 1982.
- [17] Y.-T. Hsu, Y.-Q. Yang, S.-K. Lee, H.-C. Lee, and M.-C. Lin, “An uplink multiple access scheme using pilot-free channel estimation,” *IEEE Communications Letters*, 2020.
- [18] R.-Y. Wei, “Noncoherent Block-Coded MPSK,” *IEEE Trans. Commun.*, vol. 53, pp. 978–986, June 2005.
- [19] N. Abramson, “The aloha system: another alternative for computer communications,” in *Proceedings of the November 17-19, 1970, fall joint computer conference*, pp. 281–285, 1970.
- [20] D. B. Zeleke and M. A. V. Castro, “Aloha versus single code spread aloha for satellite systems,” in *2005 IEEE 61st Vehicular Technology Conference*, vol. 4, pp. 2692–2696 Vol. 4, 2005.
- [21] J. L. Massey, “Collision-resolution algorithms and random-access communications,” in *Multi-user communication systems*, pp. 73–137, Springer, 1981.

- 
- [22] M. S. Chowdhury, K. Ashrafuzzaman, and K. S. Kwak, “Saturation throughput analysis of ieee 802.15.6 slotted aloha in heterogeneous conditions,” *IEEE Wireless Communications Letters*, vol. 3, no. 3, pp. 257–260, 2014.
- [23] H. Okada, P. M. Grant, I. W. Band, and A. Ogawa, “Application of mmse multi-user detection to cdma unslotted aloha system,” in *11th IEEE International Symposium on Personal Indoor and Mobile Radio Communications. PIMRC 2000. Proceedings (Cat. No.00TH8525)*, vol. 1, pp. 529–533 vol.1, 2000.
- [24] H.-J. Zepernick and A. Finger, *Pseudo random signal processing: theory and application*. John Wiley & Sons, 2013.
- [25] H. Minn, V. K. Bhargava, and K. B. Letaief, “A robust timing and frequency synchronization for ofdm systems,” *IEEE Transactions on Wireless communications*, vol. 2, no. 4, pp. 822–839, 2003.