國　立　交　通　大　學

電信工程學系碩士班

碩士論文

使用分散式低密度奇偶校驗碼

之壓縮傳送合作式通訊

Compressed-and-forward cooperation

with distributed LDPC coding

研　究　生：呂姵璁

指導教授：吳文榕　博士

中華民國九十八年七月

使用分散式低密度奇偶校驗碼之壓縮傳送合作式通訊

Compressed-and-forward cooperation with distributed LDPC coding

研 究 生:呂姵璁　　Student : Pei-Tsung Lu

指導教授:吳文榕 教授　　Advisor : Dr. Wen-Rong Wu

國 立 交 通 大 學

電信工程學系碩士班

碩 士 論 文

A Thesis
Submitted to Department of Communication Engineering
College of Electrical and Computer Engineering
National Chiao-Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Master of Science
In
Communication Engineering
July 2009
Hsinchu, Taiwan, Republic of China

中華民國九十八年七月

使用分散式低密度奇偶校驗碼之壓縮傳送合作式通訊

Compressed-and-forward cooperation with distributed LDPC coding

研究生:呂姵璁　　指導教授:吳文榕 教授

國立交通大學電信工程學系碩士班

摘要

文獻指出合作式通訊(cooperative communication)是一種有效探索虛擬空間多樣度的方法。除了傳統的放大傳送(AF)跟解碼傳送(DF)外，壓縮傳送(CF)也被提議為合作式通訊的一種對策。壓縮傳送的方式為：中繼端(relay)將傳送位元的Log-Likelihood Ratio (LLR)值先經量化、觀察或是估計後再傳送給終點端(destination)。在本篇論文中，我們提議用壓縮傳送的結構搭配分散式低密度奇偶校驗碼來做分析。傳統的壓縮傳送使用的是 BPSK 的調變方式且在中繼端的吞吐量比較低。為了解決這個問題，我們提議在中繼端使用 QAM 的調變方式。為了達到這個目的，我們必須估計每個位元 LLR 的分布情形。接著我們將此分布情況模擬成高斯混和(Gaussian Mixture)並使用 EM algorithm 來識別高斯混和分布中不知道的參數。模擬顯示所提出的壓縮傳送結構在性能上勝過放大傳送和解碼傳送。

Compressed-and-forward cooperation with distributed LDPC coding

Student : Pei-Tsung Lu   Advisor : Dr. Wen-Rong Wu

Department of Communication Engineering

National Chiao-Tung University

Abstract

Cooperative communication has been shown to be an effective way to explore virtual spatial diversity. Except for conventional amplify-and-forward (AF) and decode-and-forward (DF), the compressed-and-forward (CF) has been proposed for the cooperative strategy. In CF, the relay forwards the quantized/observed/estimated data information, which is usually the LLR values of the transmit bits, to the destination. In this thesis, we propose a CF scheme with distributed LDPC coding. Conventional CF only uses BPSK modulation at the relay and the throughput in the relay link is low. To solve the problem, we propose to use a QAM modulation at the relay. To do that, we have to estimate the distribution of the likelihood-ratio (LLR) of each information bit. We then model the distributions as Gaussian mixtures, and use the expectation-maximization (EM) algorithm for the identification of the unknown parameters. Simulations show that the proposed CF scheme can outperform AF and DF.

誌謝

首先要先感謝我的指導教授 吳文榕老師，在這兩年來在課業以及研究上的用心指引以及努力教導，與我討論研究上的問題以及花許多的時間解決學習上的困難和疑惑，使我在通訊的領域上受益良多，並順利的完成此論文。

還要感謝口試委員 黃家齊教授與李彥文教授給予這篇論文的寶貴意見與指導，使這篇論文更加完善及嚴謹。感謝實驗室的博班學長們，在我遇到問題時適時的給予意見及協助；還要感謝寬頻傳輸與訊號處理實驗室所有同學與學弟妹們的支持與協助，使得研究的路上有許多寶貴的回憶。

另外還要感謝我的家人、親戚、朋友和奕嘉，給予我生活上無後顧之憂還有精神上的支持與鼓勵，讓我能順利的完成碩士學位。

Contents

List of figures

1 Introduction

Diversity, operated in the time, frequency or spatial domains, is an effective technique to combat fading. In practice, spatial diversity maybe most desirable since it does not have to scarify the spectrum efficiency. To have spatial diversity, multiple antennas are required. While this is feasible in base stations, it may be difficult in mobile stations due to size, costs, hardware complexity, or other constraints. To address this limitation, the concept of cooperative diversity was introduced. In cooperative communications, mobile stations can achieve uplink transmit diversity by relaying. In a simple case, a cooperative system consists of a source, a relay and a destination. The essential advantage of user cooperation is that the relay provides an additional transmission link by forwarding part or all of the signals originated from the source to the destination. Cooperative communication provides a simple but effective means to leverage the processing and transmit power of the relay, as well as the spatial diversity of the relay channel in a wireless scenario.

Despite the many theoretic advances in wireless user cooperation, practical strategies at the relay mainly focus on three basic forms proposed by Cover and El Gamal in 1979 [1], namely, amplify-and-forward (AF), sometimes also appear in the name of scale-and-forward and reflect-and-forward, decode-and-forward (DF) and compress-and-forward (CF). In AF, it let the relay rescale, retransmit or reflect the analog signal waveforms received from the source. In DF, the source signals received at the relay are demodulated, decoded and possibly re-encoded before being forwarded to the destination. It has extended from its basic mode of the repetition DF to more complicated mode, such as distributed space time codes and network codes. CF is also referred to as observe-and-forward, quantize-and-forward or estimate-and-forward. In CF, the relay forwards the quantized/observed/estimated data information. The information is usually the LLR values of the transmit data.

The work in [2] introduces another form of the cooperative strategy, i.e., decode-amplify-forward (DAF). In DAF, the relay computes the decoder LLR, maps its analog value to the QAM plane, and then transmits to the destination. At the destination, the receiver can combine the LLR calculated from the source and that from the relay. It has been shown that the DAF can perform better than DF and AF. However, the problem with DAF is the mapping between the LLR and transmission signal. Except for BPSK and QPSK, there is not easy and straightforward approach for the mapping. Although a method was proposed for high QAM modulation schemes in [16], how to obtain a simple and effective mapping remains an open problem.

In this thesis, we will focus on the CF strategy. A practical CF scheme was proposed in [11]. In the scheme, only BPSK was considered as the modulation scheme in the relay link. Also, the turbo code is used as the coding scheme. As we know, the QAM scheme is more frequently used in actual communication systems, and the LDPC code is becoming more and more popular. In this thesis, we will extend the method in [11] and develop a practical CF scheme with the LDPC code. Specifically, we will consider the QAM modulation scheme in the relay link. To do that, we have to estimate distribution of the likelihood-ratio (LLR) of each information bit. We then model the distributions as Gaussian mixtures, and use the expectation-maximization (EM) algorithm for the identification of the unknown parameters. Using the method, we can have a higher spectral efficiency for the relay link. Simulation shows that the proposed CF scheme can outperform AF and DF.

This thesis is organized as follows: In Chapter 2, we brief review the LDPC codes. In Chapter 3, we describe the principle of cooperative communication and its two basic forwarding strategies, amplify-and-forward (AF) and decode-and-forward (DF). In Chapter 4, we review the expectation-maximization (EM) algorithm for Gaussian mixture identification. This algorithm will be used in modeling the LLR distribution. This distribution is required for the soft relaying method in our CF scheme. Then in Chapter 5, we describe the details of the proposed CF scheme. Finally, we give simulation results in Chapter 6 and draw conclusions in Chapter 7.

2 LDPC Codes

LDPC coding history starts with seminal work of Claude Shannon on his most important paper, "A Mathematical Theory of Communication", in 1948[4]. He demonstrated that there exists a coding method which can reduce the errors induced by a noisy channel to any desired level as long as the information rate is less than the capacity of the channel. The theoretical maximum information transfer rate is called Shannon limit. Although he didn't tell how to design this error correction code, his theory provided a specific goal of communication engineering. Linear Block Codes, Hamming Codes, Convolutional Codes, and Reed-Solomon Codes are well known error correction codes these days. In recent years, people start to pay much attention to Turbo Codes and LDPC Codes, and both of them are considered most complicated error correction codes.

In 1962, Gallager proposed a low-density parity-check code in his doctoral dissertation[3]. It provides near-capacity performance but the computational complexity is very high and its implementation is difficult. Also, the concatenated RS and convolutional codes were considered perfectly suitable for error control coding. Thus, his remarkable thesis was forgotten by coding researchers for almost 30 years. In 1981, Tanner generalized LDPC codes and created a bipartite graph used to represent those codes[5]. However, it was still ignored by coding theorists. LDPC codes were noticed again by some researchers until the mid-1990's because by that time the VLSI technology is mature enough to implement the code. Since that time, a lot of papers have been published and LDPC codes have become popular again.

Low-density parity-check (LDPC) codes are a class of linear block codes. The name comes from the characteristic of their parity-check matrix which contains only a few 1's in comparison to the amount of 0's. Generally, the performance of a long LDPC codeword is better than a short one. The iterative decoding principle of the LDPC code is similar to that of the Turbo Code. However, the LDPC decoding allows a parallel processing architecture, a feature that the Turbo code does not have. As a result, the LPDC code can be used in very high-speed transmission systems. Another advantage is that the patent of the LDPC code was overdue, which facilitate its wide spread real-world applications.

2.1 Encoder

LDPC encoder uses a generator matrix $\mathbf{G}$ multiplying the information vector $\vec{\mathbf{u}}$ to produce a codeword vector $\vec{\mathbf{v}}$; it can be expressed as $\vec{\mathbf{u}}\mathbf{G}=\vec{\mathbf{v}}$. Since the LDPC code is linear, $\vec{\mathbf{v}}$ multiplies to the parity check matrix $\mathbf{H_p}$ should become a zero vector. It can be written as $\mathbf{H_p}\vec{\mathbf{v}}^{\mathbf{T}}=\vec{\mathbf{0}}$.

$$\begin{aligned}&\mathbf{H_p}\vec{\mathbf{v}}^{\mathbf{T}}=\mathbf{H_p}(\vec{\mathbf{u}}\mathbf{G})^{\mathbf{T}}=\vec{\mathbf{0}}\\&\Rightarrow \mathbf{H_p}\mathbf{G}^{\mathbf{T}}=\vec{\mathbf{0}}\end{aligned} \tag{2.1}$$

Thus, the generator matrix $\mathbf{G}$ can be founds using the above equation. We now use an (6,3) LDPC code as an example. In (N,K)=(6,3) LDPC codes, there are M=N-K parity check bits. Let the parity check matrix $\mathbf{H_p}$ is given by

$$\mathbf{H_p}=\begin{bmatrix}1&1&1&1&0&0\\0&0&1&1&0&1\\1&0&0&1&1&0\end{bmatrix}$$

Using Gaussain elimation, we can have

$$\mathbf{H_p} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{H_p}' = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} = [\mathbf{P}_{3\times3} \vdots \mathbf{I}_{3\times3}] = [\mathbf{P_{M\times K}} \vdots \mathbf{I_{M\times M}}] \tag{2.2}$$

where $\mathbf{P_{M\times K}}$ is an M×K matrix, and $\mathbf{I_{M\times M}}$ is an M×M identity matrix. Then its generator matrix $\mathbf{G}$ can be found using the following equation.

$$\mathbf{G} = [\mathbf{I}_{3\times3} \vdots \mathbf{P}^{\mathbf{T}}_{3\times3}] = [\mathbf{I_{K\times K}} \vdots \mathbf{P^T_{K\times M}}] = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \tag{2.3}$$

$$\mathbf{H_p}'\mathbf{G^T} = [\mathbf{I_{M\times M}} \vdots \mathbf{P_{M\times K}}] \begin{bmatrix} \mathbf{P_{M\times K}} \\ \cdots \\ \mathbf{I_{K\times K}} \end{bmatrix} = \mathbf{P_{M\times K}} \oplus \mathbf{P_{M\times K}} = \vec{\mathbf{0}}$$

Generally, the generator matrix $\mathbf{G}$ is not a sparse matrix.

2.2 LDPC Codes Principle [3][5]

2.2.1 Message passing

A LDPC decoder uses the message passing principle to conduct decoding. We can use a simple example to explain the idea of message passing. Assuming that there is a line, and you want to know how many people are in it, you can use the following steps to obtain the answer:

Step 1: While you are informed the number from the person at a location, plus 1 and transmitting to the other side. The idea is that if a person receives a number N, that means there are N people at his/her right or left side now.

Using Figure 2-1 to illustrate the idea. At first, everyone knows there is at least one person (himself/herself), it is called intrinsic information.

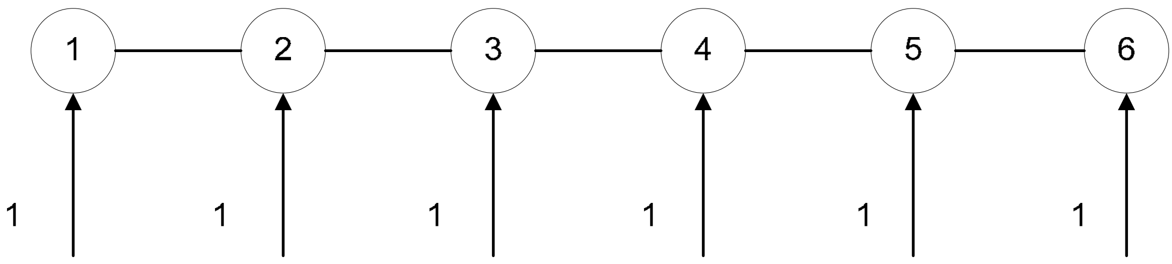


Figure 2-1 : An example of message passing (intrinsic information)

Step 2: The most left or right person transmits the 1 to his/her right/left side. Then the person receives this message add the number by 1, and informs the next person (See Figure 2-2). For the second person, he/her knows two messages; the first one is that there are 1 person at his/her left side, and the other is that there are 4 people at his/her right side. Both messages are called extrinsic information.

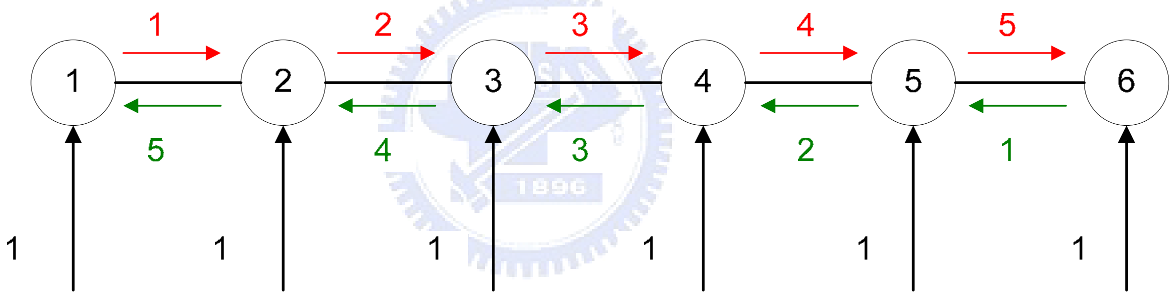


Figure 2-2 : Extrinsic information flow

Step 3: After step 2, everyone will know the total number in the line. For the second person, the total number is equal to 1(intrinsic) + 1(extrinsic) + 4(extrinsic) = 6. We then have the formula that Total number = intrinsic information + extrinsic information.

2.2.2 Tanner Graph

Analyzing the LDPC decoder with the Tanner graph is necessary for message passing decoding. An LDPC parity check matrix can be depicted in a Tanner graph, it is an effective graphical representation for LDPC codes. Not only provide these graphs a complete representation of the code, they also help to describe the decoding algorithm.

Tanner graphs are bipartite graphs meaning that the nodes of the graph are separated into two distinctive sets and edges are only connecting nodes of two different types. The two types of nodes in a Tanner graph are called variable nodes (v-nodes) and check nodes (c-nodes). Figure 2-3 is an example for such a graph. The creation of such a graph is straightforward; it consists of m check nodes (the number of parity bits) and n variable nodes (the number of bits in a codeword). Check node C_i is connected to bit node B_i if the corresponding element in $\mathbf{H_p}$ is 1.

$$\mathbf{H_p} = \begin{array}{c} \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix} \\ B_1\ B_2\ B_3\ B_4\ B_5\ B_6 \end{array} \begin{array}{c} C_1 \\ C_2 \\ C_3 \\ \end{array}$$

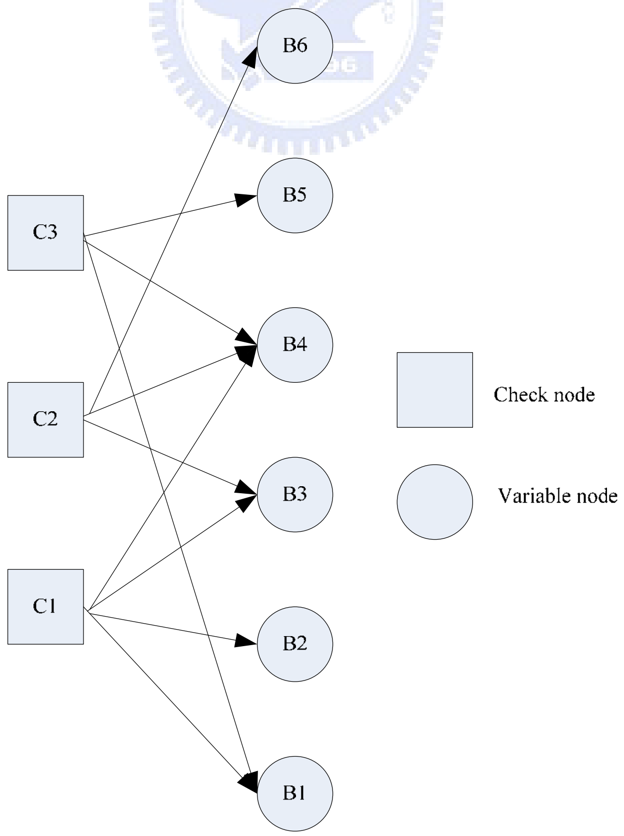


Figure 2-3: An example of Tanner Graph

2.3 Sum Product Algorithm in LDPC Codes Decoding [6]

There are many decoding algorithms for LDPC Codes, such as Majority-logic (MLG) decoding, Bit-flipping (BF) decoding, and Sum product algorithm (SPA). The last one is also called Belief propagation algorithm (BPA) and Message passing algorithm (MPA). The sum product algorithm is the basic and standard decoding algorithm for LDPC codes.

Figure 2-4 depicts the structure of a LDPC decoder. In the figure, bit nodes and check nodes compute the bit probabilities in their nodes before send the message to each others. The probability in one of the bit nodes is decoded by all the check nodes (except for the check node receives the probability from the bit node) and another node connect to that bit node. After the computation, the bit node will send this probability to one of check nodes. In a similar way, the probability in one of the check nodes is decoded by all the bit nodes (except for the bit node receives the probability from the check node) connect to that bit node.

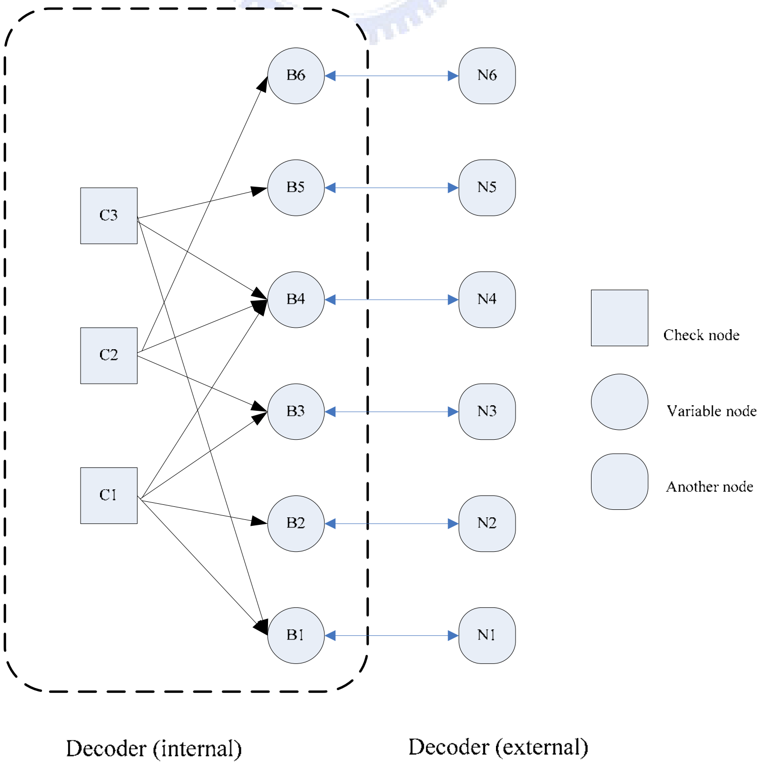


Figure 2-4 : Structure of LDPC decoder

2.3.1 Bit nodes to Check nodes

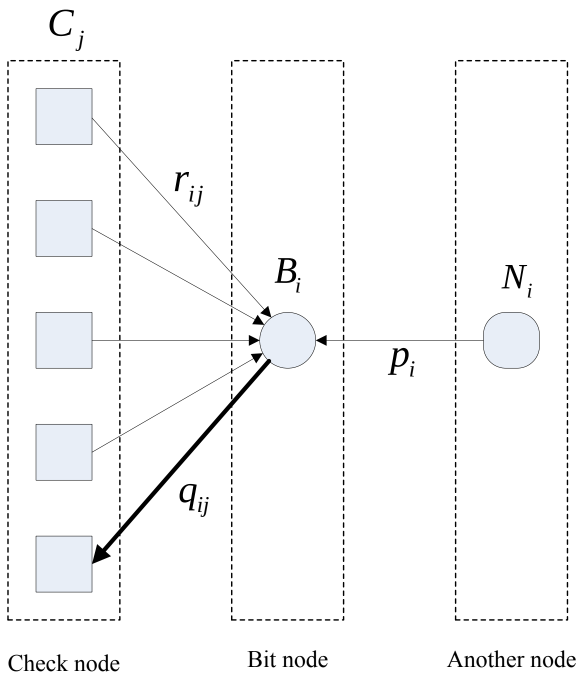


Figure 2-5 : Probabilities of bit nodes to check nodes

From Figure 2-5, we see that p_i is the probability transmitted from another node N_i to bit node B_i; r_{ij} the probability transmitted from check node C_j to bit node B_i; q_{ij} the probability transmitted from bit node B_i to check node C_j. If bit node B_i connects to another node N_i and K check nodes, and they are independent, we then have

$$P(q_{ij}=\zeta)=P(p_i=\zeta)\prod_{j'\in M(i)\setminus\{j\}}P(r_{ij'}=\zeta) \tag{2.4}$$

Where $\zeta=\{0 \text{ or } 1\}$, $M(i)$ is the set of the i th row of $\mathbf{H_p}$ is 1, and $M(i)\setminus\{j\}$ the set of $M(i)$ excluding the j th element. Define the Log-Likelihood Ratio (LLR) of a bit as

$$LLR(a)=\log\frac{P(a=1)}{P(a=0)} \tag{2.5}$$

We can then compute $LLR(r_{ij})$, $LLR(p_i)$, and $LLR(q_{ij})$. Using them in (2.4), we have

$$LLR(q_{ij}) = LLR(p_i) + \sum_{j' \in M(i)\setminus\{j\}} LLR(r_{ij'}) \tag{2.6}$$

Thus, $LLR(q_{ij})$ is the probability message transmitted from bit node to check node.

2.3.2 Check nodes to Bit nodes

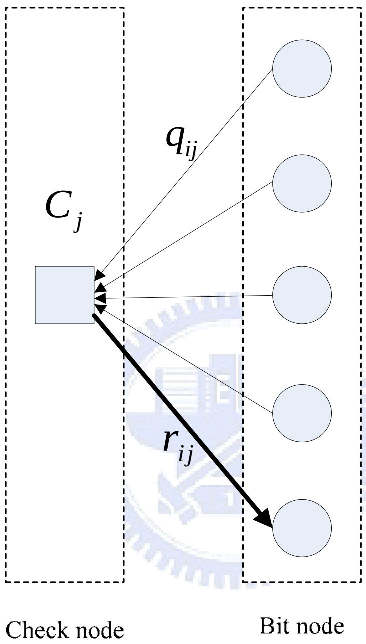


Figure 2-6: Probabilities of check nodes to bit nodes

From Figure 2-6, we see that check node C_j is independent to K of bit nodes. Since it is necessary to satisfy the equation $\mathbf{H_p}\vec{\mathbf{v}}^{\mathbf{T}} = \vec{\mathbf{0}}$, all the bit nodes connect to the same check node also need to satisfy equation shown below:

$$B_1 \oplus B_2 \oplus B_3 \oplus \cdots \oplus B_{i-1} \oplus B_i \oplus B_{i+1} \oplus \cdots \oplus B_K = 0 \tag{2.7}$$

Then from (2.7), we have

$$\begin{aligned} P(r_{ij} = 1) &= P(B_1 \oplus B_2 \oplus B_3 \oplus \cdots \oplus B_{i-1} \oplus B_i \oplus B_{i+1} \oplus \cdots \oplus B_K = 1) \\ P(r_{ij} = 0) &= P(B_1 \oplus B_2 \oplus B_3 \oplus \cdots \oplus B_{i-1} \oplus B_i \oplus B_{i+1} \oplus \cdots \oplus B_K = 0) \end{aligned} \tag{2.8}$$

Using the mathematical induction, we can the general expressions for (2.8).

(i) K=2, $P(B_1 = 1) = a_1$, $P(B_2 = 1) = a_2$

$$P(B_1 \oplus B_2 = 1) = a_1(1-a_2) + a_2(1-a_1)$$
$$P(B_1 \oplus B_2 = 0) = a_1 a_2 + (1-a_1)(1-a_2) \tag{2.9}$$

Then we can rewrite the result in (2.9) as:

$$P(B_1 \oplus B_2 = 1) = \frac{1-(1-2a_1)(1-2a_2)}{2} = \frac{1-\prod_{i=1}^{2}(1-2a_i)}{2}$$
$$P(B_1 \oplus B_2 = 0) = \frac{1+(1-2a_1)(1-2a_2)}{2} = \frac{1+\prod_{i=1}^{2}(1-2a_i)}{2} \tag{2.10}$$

(i i) If K=n-1, the equation is satisfied, then

$$P(B_1 \oplus B_2 \oplus B_3 \oplus \cdots \oplus B_{n-1} = 1) = \frac{1-\prod_{i=1}^{n-1}(1-2a_i)}{2} = M_{n-1}$$
$$P(B_1 \oplus B_2 \oplus B_3 \oplus \cdots \oplus B_{n-1} = 0) = \frac{1+\prod_{i=1}^{n-1}(1-2a_i)}{2} = 1-M_{n-1} \tag{2.11}$$

$$2M_{n-1} + 1 = \prod_{i=1}^{n-1}(1-2a_i) \tag{2.12}$$

(i i i) K=n, $P(B_n = 1) = a_n$

$$P(B_1 \oplus B_2 \oplus B_3 \oplus \cdots \oplus B_{n-1} \oplus B_n = 1) = a_n(1-M_{n-1}) + M_{n-1}(1-a_n)$$
$$P(B_1 \oplus B_2 \oplus B_3 \oplus \cdots \oplus B_{n-1} \oplus B_n = 0) = a_n M_{n-1} + (1-a_n)(1-M_{n-1}) \tag{2.13}$$

Then it can be spread as (2.14):

$$P(B_1 \oplus B_2 \oplus \cdots \oplus B_n = 1) = \frac{1-(1-2a_n)(1-2M_{n-1})}{2} = \frac{1-\prod_{i=1}^{n}(1-2a_i)}{2}$$
$$P(B_1 \oplus B_2 \oplus \cdots \oplus B_n = 0) = \frac{1+(1-2a_n)(1-2M_{n-1})}{2} = \frac{1+\prod_{i=1}^{n}(1-2a_i)}{2} \tag{2.14}$$

Thus, we can the general expressions as

$$p(r_{ij}=1)=P(B_1\oplus B_2\oplus\cdots\oplus B_K=1)=\frac{1-\prod_{i^{'}\in L(j)\backslash\{i\}}(1-2Q_{i^{'}j})}{2}$$

$$p(r_{ij}=0)=P(B_1\oplus B_2\oplus\cdots\oplus B_K=0)=\frac{1+\prod_{i^{'}\in L(j)\backslash\{i\}}(1-2Q_{i^{'}j})}{2} \qquad (2.15)$$

Where $Q_{i^{'}j}=P(q_{i^{'}j}=1)$, $L(j)$ is the set of nonzero row indexes in the j th column of $\mathbf{H_p}$, and $L(j)\backslash\{i\}$ is the set of $L(j)$ excluding the i th element. From the definition of LLR in (2.5), we then have

$$LLR(r_{ij})=\log\frac{P(r_{ij}=1)}{P(r_{ij}=0)}=\log\frac{1-\prod_{i^{'}\in L(j)\backslash\{i\}}(1-2Q_{i^{'}j})}{1+\prod_{i^{'}\in L(j)\backslash\{i\}}(1-2Q_{i^{'}j})} \qquad (2.16)$$

Consider the following two equations.

$$LLR(q_{i^{'}j})=\log\frac{Q_{i^{'}j}}{1-Q_{i^{'}j}} \;\Rightarrow\; e^{LLR(q_{i^{'}j})}=\frac{Q_{i^{'}j}}{1-Q_{i^{'}j}} \qquad (2.17)$$

$$\tanh(x)=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \;\Rightarrow\; \tanh\left(\frac{x}{2}\right)=\frac{e^{x}-1}{e^{x}+1} \qquad (2.18)$$

Now, letting $x=LLR(q_{i^{'}j})$ in (2.18) and using (2.17), we have

$$\tanh\left(\frac{LLR(q_{i^{'}j})}{2}\right)=\frac{\dfrac{Q_{i^{'}j}}{1-Q_{i^{'}j}}-1}{\dfrac{Q_{i^{'}j}}{1-Q_{i^{'}j}}+1}=2Q_{i^{'}j}-1 \qquad (2.19)$$

Substituting (2.19) into (2.16) leads to

$$LLR(r_{ij})=\log\frac{1-(-1)^{|L(j)|-1}\prod_{i^{'}\in L(j)\backslash\{i\}}\tanh\left(\dfrac{LLR(q_{i^{'}j})}{2}\right)}{1+(-1)^{|L(j)|-1}\prod_{i^{'}\in L(j)\backslash\{i\}}\tanh\left(\dfrac{LLR(q_{i^{'}j})}{2}\right)} \qquad (2.20)$$

$$\tanh^{-1}(y)=\frac{1}{2}\log\frac{1+y}{1-y} \;\Rightarrow\; -2\tanh^{-1}(y)=\log\frac{1-y}{1+y} \qquad (2.21)$$

Let $y = (-1)^{|L(j)|-1} \prod_{i' \in L(j) \setminus \{i\}} \tanh\left(\frac{LLR(q_{i'j})}{2}\right)$ in (2.21). Then, we can write $LLR(r_{ij})$ as:

$$LLR(r_{ij}) = 2 \times (-1)^{|L(j)|} \times \tanh^{-1}\left(\prod_{i' \in L(j) \setminus \{i\}} \tanh\left(\frac{LLR(q_{i'j})}{2}\right)\right) \tag{2.22}$$

So $LLR(r_{ij})$ in equation (2.22) is the probability message transmitted from check nodes to bit nodes.

2.3.3 Posteriori Probability of Bit Node

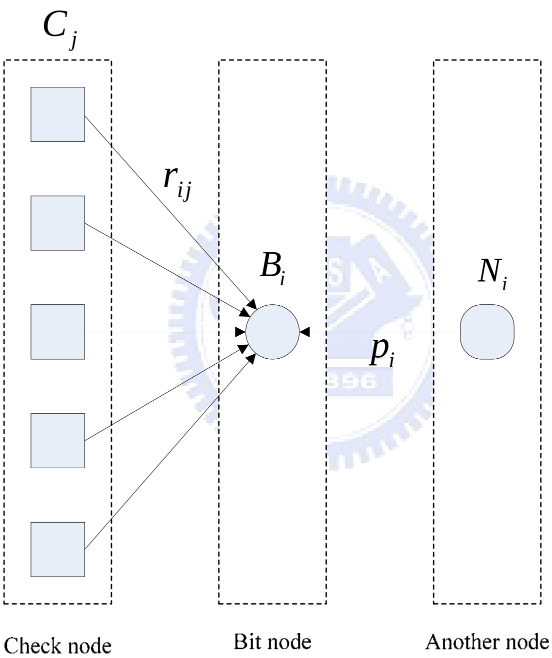


Figure 2-7: Posteriori probabilities of bit nodes

From Figure 2-7, we see that the codeword bit B_i is decided by another node N_i and all the connected check nodes. Then we know that

$$P(B_i = \zeta) = P(p_i = \zeta) \prod_{j \in M(i)} P(r_{ij} = \zeta) \tag{2.23}$$

Then,

$$LLR(B_i) = LLR(p_i) + \sum_{j \in M(i)} LLR(r_{ij}) \tag{2.24}$$

So $LLR(B_i)$ in (2.24) is the posteriori probability for bit node B_i.

Finally, we can make the decision for B_i. If $LLR(B_i) \geq 0$, we can decide bit B_i as 1. Otherwise, B_i is 0.

$$B_i = \begin{cases} 1 & , \quad LLR(B_i) \geq 0 \\ 0 & , \quad LLR(B_i) < 0 \end{cases} \tag{2.25}$$

2.3.4 Sum Product Algorithm in LDPC

We now summarize the procedure to conduct the sum product decoding algorithm for LPDC codes as follows:

Step 1: Initialization: Set the maximum iteration number k_{MAX}, and assume the initial probabilities of check nodes to bit nodes as

$$LLR^{(0)}(r_{ij}) = \log \frac{0.5}{0.5} = 0 \tag{2.26}$$

Step 2: Computation of the probability message from bit nodes to check nodes:

$$LLR^{(k)}(q_{ij}) = LLR(p_i) + \sum_{j' \in M(i) \setminus \{j\}} LLR^{(k-1)}(r_{ij'}) \tag{2.27}$$

Where the superscript indicate the result of the the k th iteration.

Step 3: Computation of the probability message from check nodes to bit nodes:

$$LLR^{(k)}(r_{ij}) = 2 \times (-1)^{|L(j)|-1} \times \tanh^{-1}\left(\prod_{i' \in L(j) \setminus \{i\}} \tanh\left(\frac{LLR^{(k)}(q_{i'j})}{2} \right) \right) \tag{2.28}$$

Step 4: Computation of the posteriori probability message for B_i and making the decision:

$$LLR^{(k)}(B_i) = LLR(p_i) + \sum_{j \in M(i)} LLR^{(k)}(r_{ij}) \tag{2.29}$$

$$B_i^{(k)} = \begin{cases} 1 & , \quad LLR^{(k)}(B_i) \geq 0 \\ 0 & , \quad LLR^{(k)}(B_i) < 0 \end{cases} \tag{2.30}$$

Step 5: Iteration until the codeword satisfy $\mathbf{H_p} \bar{\mathbf{v}}^{\mathbf{T}} = \vec{\mathbf{0}}$ or $k = k_{MAX}$.

2.4 LDPC in 802.15.3c

In this section, we will introduce the LDPC parity-check matrix in IEEE 802.15.3c systems [7]. First, we should notice about the UEP (Unequal Error Protection) property in LDPC Codes. In [8], it mentions that bit nodes with differet degrees have different UEP. The error protection capability of irregular LDPC Codes is improved with its higher degree. We can call these higher degree bit nodes the higher error protection nodes. In the other words, with the lower numbers of connected check nodes, the bit node has the lower error protection. Figure 2-8 is the LDPC parity-check matrix in IEEE 802.15.3c systems with different code rates.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	2	∞	∞	∞	4	∞	∞	∞	∞	∞	2	∞	∞	5	∞	∞	∞	∞	10	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
2	∞	∞	9	∞	∞	∞	∞	5	5	∞	∞	∞	10	∞	∞	∞	∞	∞	∞	8	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
3	∞	11	∞	∞	∞	15	∞	∞	∞	∞	∞	8	∞	∞	∞	6	∞	17	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
4	∞	∞	∞	10	∞	∞	2	∞	∞	13	∞	∞	∞	∞	10	∞	5	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
5	∞	2	∞	∞	∞	4	∞	∞	∞	∞	∞	2	∞	∞	5	∞	∞	∞	∞	10	∞	∞	7	∞	∞	∞	5	∞	∞	∞	∞	∞
6	∞	∞	∞	9	5	∞	∞	∞	∞	5	∞	∞	∞	10	∞	∞	8	∞	∞	∞	∞	∞	∞	16	∞	2	∞	∞	∞	∞	∞	∞
7	∞	∞	11	∞	∞	∞	15	∞	8	∞	∞	∞	6	∞	∞	∞	∞	∞	17	∞	∞	6	∞	∞	7	∞	∞	∞	∞	∞	∞	∞
8	10	∞	∞	∞	∞	∞	∞	2	∞	∞	13	∞	∞	∞	∞	10	∞	5	∞	∞	0	∞	∞	∞	∞	∞	∞	8	∞	∞	∞	∞
9	∞	∞	2	∞	∞	∞	4	∞	2	∞	∞	∞	∞	∞	∞	5	10	∞	∞	∞	∞	∞	∞	7	∞	∞	∞	∞	∞	∞	∞	∞
10	9	∞	∞	∞	∞	5	∞	∞	∞	∞	5	∞	∞	∞	10	∞	∞	8	∞	∞	16	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
11	∞	∞	∞	11	∞	∞	∞	15	∞	8	∞	∞	∞	6	∞	∞	∞	∞	∞	17	∞	∞	6	∞	∞	∞	∞	∞	∞	∞	∞	∞
12	∞	10	∞	∞	2	∞	∞	∞	∞	∞	∞	13	10	∞	∞	∞	∞	∞	5	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
13	∞	∞	∞	2	∞	∞	∞	4	∞	2	∞	∞	5	∞	∞	∞	∞	10	∞	∞	7	∞	∞	∞	5	∞	∞	∞	∞	∞	∞	12
14	∞	9	∞	∞	∞	∞	5	∞	∞	∞	∞	5	∞	∞	∞	10	∞	∞	8	∞	∞	16	∞	∞	∞	∞	∞	2	∞	16	∞	∞
15	11	∞	∞	∞	15	∞	∞	∞	∞	∞	8	∞	∞	∞	6	∞	17	∞	∞	∞	∞	∞	∞	6	∞	∞	7	∞	∞	∞	10	∞
16	∞	∞	10	∞	∞	2	∞	∞	13	∞	∞	∞	∞	10	∞	∞	∞	∞	∞	5	∞	∞	0	∞	∞	8	∞	∞	16	∞	∞	∞

L1 L2 L3 L4

parity check matrix of rate 1/2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	2	∞	9	∞	4	∞	∞	5	12	∞	16	∞	5	∞	2	∞	10	5	∞	∞	∞	∞	10	8	∞	7	∞	∞	∞	∞	∞	∞
2	∞	11	∞	10	∞	15	2	∞	∞	16	∞	10	∞	13	∞	8	∞	∞	10	6	5	17	∞	∞	6	∞	∞	∞	∞	∞	∞	∞
3	∞	2	∞	9	5	4	∞	∞	∞	12	∞	16	∞	5	∞	2	∞	10	5	∞	8	∞	∞	10	∞	∞	7	16	∞	2	∞	∞
4	10	∞	11	∞	∞	∞	15	2	10	∞	16	∞	8	∞	13	∞	6	∞	∞	10	∞	5	17	∞	0	6	∞	∞	7	∞	∞	∞
5	9	∞	2	∞	∞	5	4	∞	16	∞	12	∞	2	∞	5	∞	∞	∞	10	5	10	8	∞	∞	16	∞	∞	7	∞	∞	∞	∞
6	∞	10	∞	11	2	∞	∞	15	∞	10	∞	16	∞	8	∞	13	10	6	∞	∞	∞	∞	5	17	∞	0	6	∞	∞	∞	∞	∞
7	∞	9	∞	2	∞	∞	5	4	∞	16	∞	12	∞	2	∞	5	5	∞	∞	10	∞	10	8	∞	7	16	∞	∞	5	∞	∞	2
8	11	∞	10	∞	15	2	∞	∞	16	∞	10	∞	13	∞	8	∞	∞	10	6	∞	17	∞	∞	5	∞	∞	0	6	∞	8	7	∞

L1 L2 L3 L4

parity check matrix of rate 3/4

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	2	11	9	10	4	15	2	5	12	16	16	10	5	13	2	8
2	10	2	11	9	5	4	15	2	10	12	16	16	8	5	13	2
3	9	10	2	11	2	5	4	15	16	10	12	16	2	8	5	13
4	11	9	10	2	15	2	5	4	16	16	10	12	13	2	8	5

	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	10	5	10	6	5	17	10	8	6	7	16	0	2	∞	∞	∞
2	6	10	5	10	8	5	17	10	0	6	7	16	7	2	∞	∞
3	10	6	10	5	10	8	5	17	16	0	6	7	8	7	2	∞
4	5	10	6	10	17	10	8	5	7	16	0	6	5	8	7	2

parity check matrix of rate 7/8

Figure 2-8: Parity-check matrices for rate 1/2, 3/4, and 7/8 in 802.15.3c

In Figure 2-8, the integrate number in a grid indicate the shifting number of a 21x21 unit matrix. And, L1, L2, L3, and L4 indicate the UEP level in the parity-check matrix; L1 has better performance than L2, and so on. The shifting operation in an unit matrix is illustrated below. For example, for a shift number S=0,1,2 in a 8x8 unit matrx, we have

$$J^0_{8x8}=\begin{bmatrix}1&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&1\end{bmatrix},\ J^1_{8x8}=\begin{bmatrix}0&0&0&0&0&0&0&0\\1&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&0&0\\0&0&0&0&0&0&1&0\end{bmatrix},\ J^2_{8x8}=\begin{bmatrix}0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\1&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&0&0\end{bmatrix} \quad (2.31)$$

Figure 2-9 shows the density function of decoded LLR's (16QAM) in different UEP levels at SNR=25dB, and code rate=1/2. We can see that in L1, the distance between the two modes (corresponding to bit 1 or 0) is largest. In other words, the probability of decision errors will be the smallest. Figure 2-10 shows the BER for each UEP level. The result conforms the assertion we just made. In the simulation and later chapter, we will need to use the UEP property to facilitate our analysis.

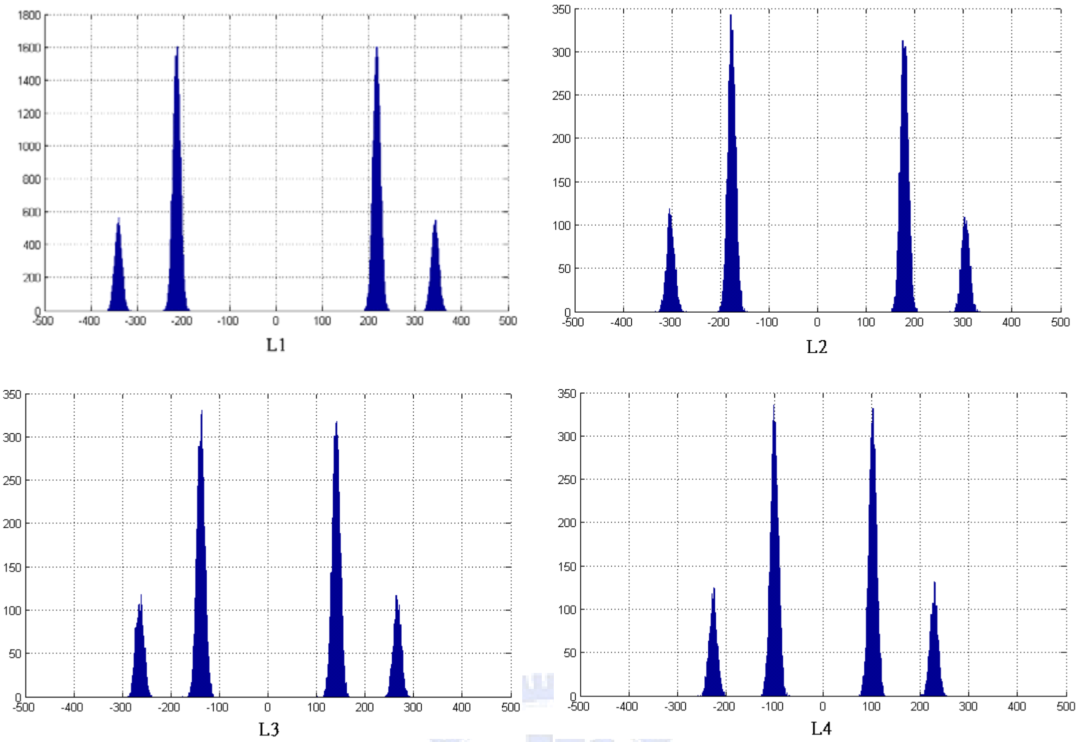


Figure 2-9: LLR density functions in different UEP level

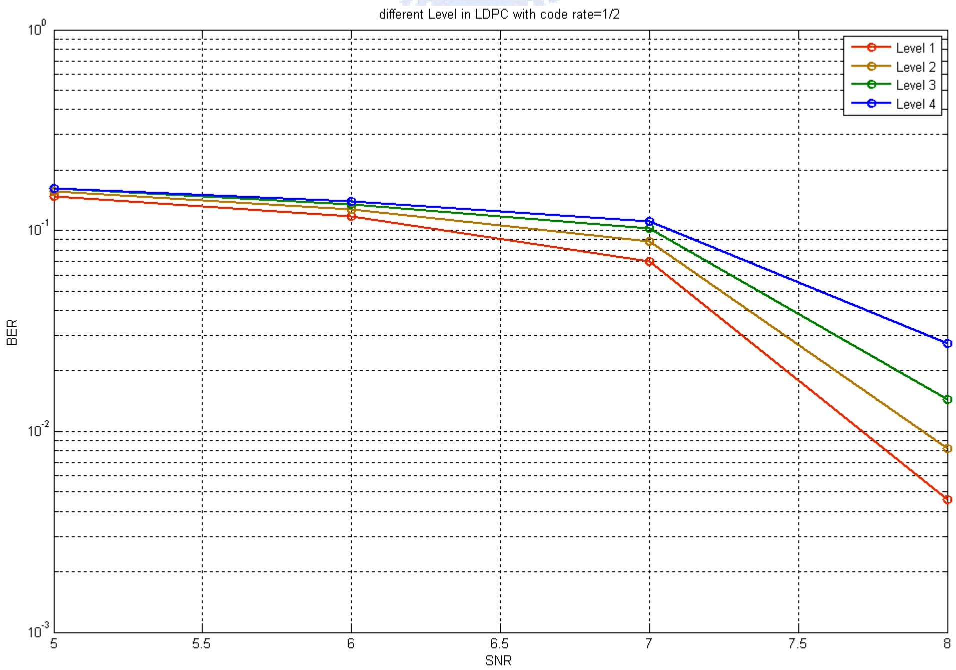


Figure 2-10: BER for each UEP level

3 Cooperative Communication Systems

3.1 Cooperative Communication

Figure 3-1 shows a simplest three-terminal network consisting of a source, a relay and a destination, showing the basic idea behind this concept. Since each of the users sees an independent fading path to the destination, diversity is obtained by transmitting the data through the relay. By using this approach, multiple virtual-antennas can be constructed in the transmitter. Many research works also show that considerable benefits result from signal relaying in fading environments especially over slow fading channels, including the reduction in outage probability, high capacity, less power consumption and wider dynamic range.

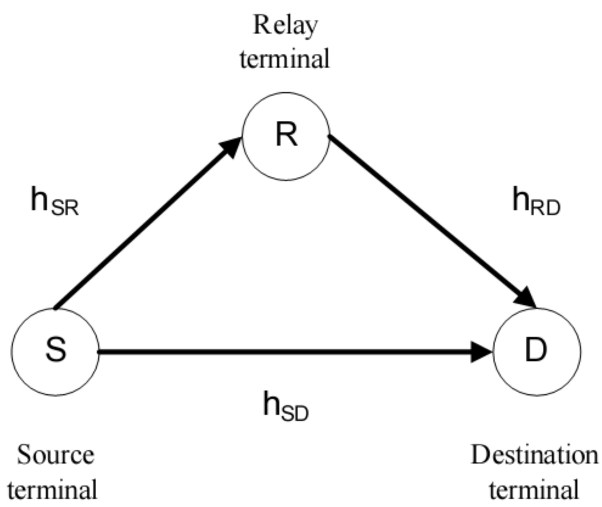


Figure 3-1: The scenario of relay channel

Despite the theoretic advances in wireless user cooperation, practical signal relaying strategies have not evolved much out of the three basic forms proposed by Cover and El Gamal in 1979, namely, amplify-and –forward (AF), decode-and-forward (DF) and compress-and-forward (CF). In section 3.3 to 3.5, we will review two basic strategies AF and DF, and also the scenarios including LDPC

Codes in these two strategies, after the system model is first given in section 3.2.

A particular powerful variation of user cooperation is coded cooperation. Coded cooperation integrates cooperation into channel coding. The codeword will experience two independent channels before it is received by the destination.

3.2 System Model [2]

Consider the basic relay system in Figure 3-1 that comprises a source node, a relay node and a destination node. We consider the half-duplex transmit mode which means that the system cannot send and receive data at the same time. The user cooperation is operated in two stages: the broadcasting stage, where the source broadcasts a packet of data to both the destination and the relay, and the relaying stage, where the relay processes and forwards part or all of the observations to the destination. The destination then combines the signals received from both stages to make a best estimation of the original data. Throughout the paper, we will use subscripts *S*, *R*, *D* and *SR*, *SD*, *RD* to denote the quantities pertaining to the source, relay, and destination nodes, and those pertaining to the source-to-relay, source-to-destination and relay-to-destination channels, respectively.

We take AWGN and block Rayleigh fading as our channel models, which are described as

$$y_{SR}(i) = h_{SR}x_S(i) + n_{SR}(i) \tag{3.1}$$

$$y_{SD}(i) = h_{SD}x_S(i) + n_{SD}(i) \tag{3.2}$$

$$y_{RD}(i) = h_{RD}x_R(i) + n_{RD}(i) \tag{3.3}$$

where x_S is the transmitted signal from the source, x_R is the transmitted signal from the relay, *y* is the received signal and *h* is the channel state information. In the case of AWGN, *h* is a constant of 1. In the case of block fading, *h* follows a Rayleigh distribution with a variance of 1, remains fixed over a block of fixed size, and changes independently between successive blocks. When we consider a binary-shift keying (BPSK) modulation, $x_S \in \{1,-1\}$ $(0 \rightarrow -1, 1 \rightarrow 1)$. The AWGNs, n_{SD}, n_{SR} and n_{RD}, have zero mean and the variances of σ_{SD}^2, σ_{SR}^2 and σ_{RD}^2, respectively. We consider spatially independent channels among the source, the relay, and the destination. We also assume that the instantaneous channel condition is known to the receivers, so that the decoder can exploit efficient soft decoding algorithms.

With LDPC Codes in the system, we consider the LDPC Code defined in IEEE 802.15.3c, i.e., r=1/2 and (N,K)=(672,336). In the cooperative system, we let the packet size for transmission is 672 bits. And, the transition protocol is TDMA which means that in the first time slot the source transmits a data packet to both the destination and the relay, and in the second time slot the relay forwards the packet to the destination. In the second time slot, the source neither transmits nor receives signal.

3.3 Amplify-and-Forward (AF)

In the AF scenario, the relay only amplifies and retransmits the analog signal waveform received from the source. The operation of AF is quite straightforward, requiring a lower implementation complexity in digital signal processing. More importantly, AF can operate at all times, even when the source-to-relay channel experiences outage.

In the first time slot, the relay receives the data packet from the source. Due to the channel h_{SR}, the packet will experience fading and be contaminated with noise n_{SR}. In the second time slot, the relay just amplifies and re-transmits the received signal to the destination. Finally, the destination uses the maximum ratio combining (MRC) detector to combine the received signals from the both time slots and recover the original transmitted data

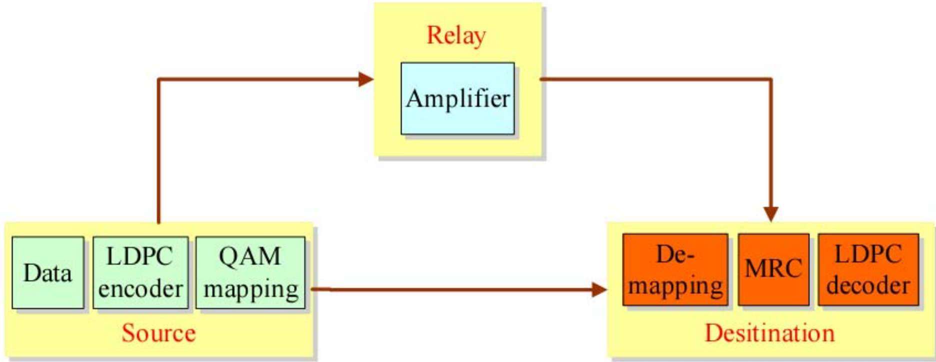


Figure 3-2: AF block diagram

Mathematically, the transmit signal at the relay is formulated as

$$x_R(i) = \frac{y_{SR}(i)}{\sqrt{\overline{P}_y}} \quad i = 1, 2, \ldots, N \tag{3.4}$$

where N is the length of the codeword (block), $x_R(i)$ is the retransmitted signal at the relay, and $\overline{P}_y$ is the average power of the received signals:

$$\overline{P}_y = \frac{\sum_{i=1}^{N} | y_{SR}(i) |^2}{N} \overset{N\to\infty}{\rightarrow} | h_{SR} |^2 + \sigma_{SR}^2 \tag{3.5}$$

The destination observes from the source-relay-destination (S-R-D) channel a noisy signal of the form:

$$\begin{aligned} y_{RD}(i) &= h_{RD}\left(\frac{h_{SR}x_S(i) + n_{SR}(i)}{\sqrt{\overline{P}_y}}\right) + n_{RD}(i) \\ &= \frac{h_{RD}h_{SR}x_S(i)}{\sqrt{\overline{P}_y}} + \frac{h_{RD}n_{SR}(i)}{\sqrt{\overline{P}_y}} + n_{RD}(i) \end{aligned} \tag{3.6}$$

Equation (3.6) makes the cascade channel behave like a single (block) fading channel with fading coefficient $\left(\frac{h_{RD}h_{SR}}{\sqrt{\overline{P}_y}}\right)$, and a complex Gaussian noise of variance $\left(\frac{| h_{RD} |^2 \sigma_{SR}^2}{\overline{P}_y} + \sigma_{RD}^2\right)$. Where h is CSI and σ^2 is the noise variance.

To conduct decoding, the LLR of the transmit bit must be calculate first. In cooperative communication, the LLR in the destination can be calculated by an efficient way. Upon receiving a symbol, we then have to calculate the LLR of a certain bit. This is referred to soft demapping and it can be derived as follows [9] :

First, if we receive a signal *r*, we can get equalized signal *y*, it will formulate as (3.7).

$$r(i) = h(i)\cdot x(i) + n(i) \rightarrow y(i) = x(i) + \frac{n(i)}{h(i)} \tag{3.7}$$

Then the conditional pdf of *y* is (3.8) and the LLR is (3.9):

$$p(r(i)\,|\,x(i)=\alpha) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left\{-\frac{|r(i)-h(i)\cdot\alpha|^2}{2\sigma^2}\right\} \tag{3.8}$$

$$\begin{aligned} LLR(b_{I,k}) &= \log\frac{p(b_{I,k}=1\,|\,r(i))}{p(b_{I,k}=0\,|\,r(i))} = \log\frac{\sum_{\alpha\in S_{I,k}^{(1)}} p(x(i)=\alpha\,|\,r(i))}{\sum_{\alpha\in S_{I,k}^{(0)}} p(x(i)=\alpha\,|\,r(i))} \\ &\overset{\text{By equal distributed}}{\underset{\text{Bayes Rule}}{=}} \log\frac{\sum_{\alpha\in S_{I,k}^{(1)}} p(r(i)\,|\,x(i)=\alpha)}{\sum_{\alpha\in S_{I,k}^{(0)}} p(r(i)\,|\,x(i)=\alpha)} \\ &\overset{\log\sum_j Z_j \approx \max_j \log Z_j}{\approx} \log\frac{\max_{\alpha\in S_{I,k}^{(1)}} p(r(i)\,|\,x(i)=\alpha)}{\max_{\alpha\in S_{I,k}^{(0)}} p(r(i)\,|\,x(i)=\alpha)} \end{aligned} \tag{3.9}$$

Where $S_{I.k}^{(0)}$ is the in-phase part region of 0 in the *k*th bit,α is {-1,1,-3,3} in the $S_{I.k}^{(0)}$and $S_{I.k}^{(1)}$, and $b_{I,k}$ is the *k*th in-phase part bit.

Using (3.7) and (3.8) in (3.9), we have

$$\begin{aligned} LLR(b_{I,k}) &= \log\frac{\max_{\alpha\in S_{I,k}^{(1)}}\exp\left(-\frac{|r(i)-h(i)\cdot\alpha|^2}{2\sigma^2}\right)}{\max_{\alpha\in S_{I,k}^{(0)}}\exp\left(-\frac{|r(i)-h(i)\cdot\alpha|^2}{2\sigma^2}\right)} \\ &= \log\max_{\alpha\in S_{I,k}^{(1)}}\exp\left(-\frac{|r(i)-h(i)\cdot\alpha|^2}{2\sigma^2}\right) - \log\max_{\alpha\in S_{I,k}^{(0)}}\exp\left(-\frac{|r(i)-h(i)\cdot\alpha|^2}{2\sigma^2}\right) \\ &= \frac{-1}{2\sigma^2}\min_{\alpha\in S_{I,k}^{(1)}}|h(i)|^2\cdot|y(i)-\alpha|^2 + \frac{1}{2\sigma^2}\min_{\alpha\in S_{I,k}^{(0)}}|h(i)|^2\cdot|y(i)-\alpha|^2 \end{aligned}$$

$$= \frac{|h(i)|^2}{2\sigma^2}\left\{ \min_{\alpha \in S_{I,k}^{(0)}} |y(i)-\alpha|^2 - \min_{\alpha \in S_{I,k}^{(1)}} |y(i)-\alpha|^2 \right\}$$

$$\stackrel{\Delta}{=} \frac{2|h(i)|^2}{\sigma^2} D_{I,k} \tag{3.10}$$

Where $D_{I,k} = \frac{1}{4}\left\{ \min_{\alpha \in S_{I,k}^{(0)}} |y(i)-\alpha|^2 - \min_{\alpha \in S_{I,k}^{(1)}} |y(i)-\alpha|^2 \right\}$

For a 16 QAM symbol mapping, the function of $D_{I,K}$ is plotted in Figure 3-3 and the 16QAM constellation is in Figure 3-4. If we only see the area where y(i)>0, we can find that the function is nonlinear for two mapped bits. For simplicity, we can approximate it as a linear function.

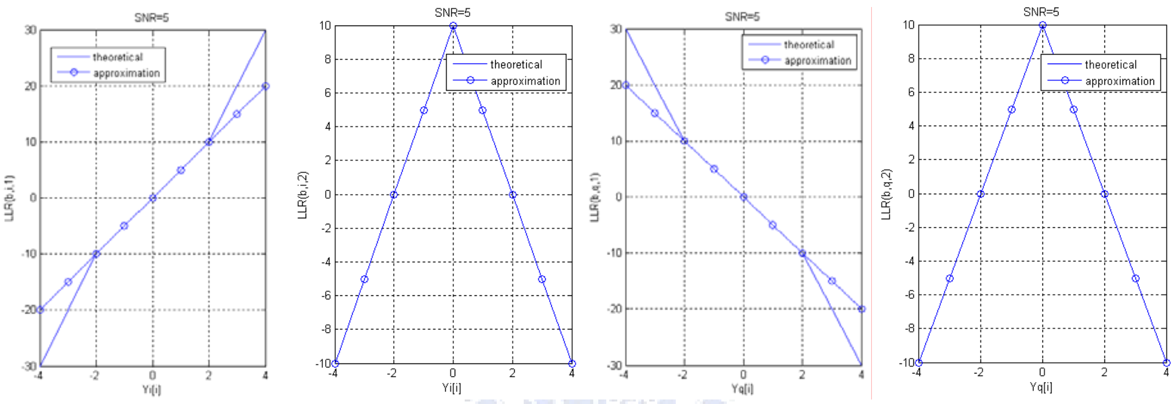
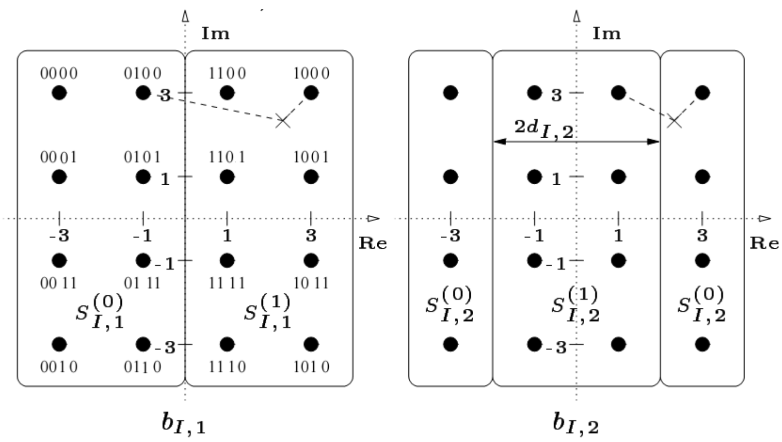


Figure 3-3: Approximate versus exact LLR functions for the in-phase and quad-phase of the 16QAM constellation



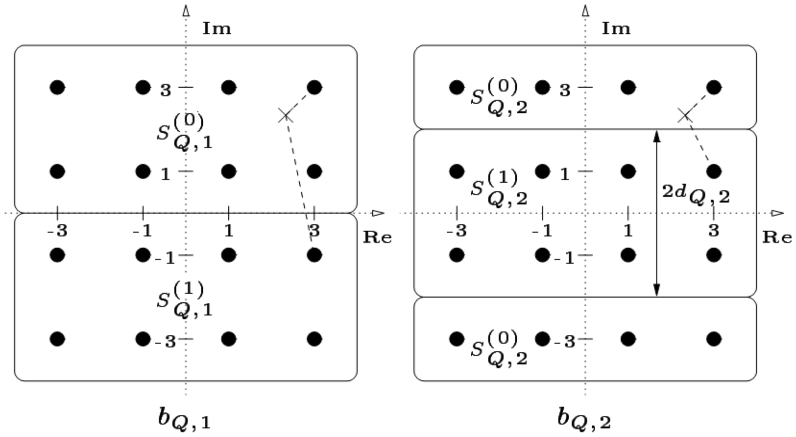


Figure 3-4: Partition of the 16QAM constellation

We then have

$$D_{I,k} \approx \begin{cases} y_I(i) & , \quad k=1 \\ -|y_I(i)|+2 & , \quad k=2 \end{cases} \tag{3.11}$$

The destination then gathers the signals received from both the cascade channel and the direct source-to-destination channel using maximal ratio combining (MRC), which in effect is to extract and combine the log-likelihood ratios (LLR) from the channels, i.e.,

$$LLR_{AF}(i) = LLR_{SD}(i) + LLR_{RD}(i) = \frac{2|h_{SD}|^2}{\sigma_{SD}^2} D_{I,k} + \frac{2\sqrt{\bar{P}_y}\,|h_{SR}|^2 |h_{RD}|^2}{|h_{RD}|^2\,\sigma_{SR}^2 + \bar{P}_y \sigma_{RD}^2} D_{I,k} \tag{3.12}$$

Where $LLR_{\mathrm{SD}}^{\mathrm{ch}}$ and $LLR_{\mathrm{RD}}^{\mathrm{ch}}$ are the LLR of channel S-D and R-D.

Here, we prove that the LLRs of BPSK signals obtained by the summation of LLRs caculated from the direct and the relay link is equivalent to that caculated from the received signal after MRC :

Assume the destination receives y_{SD} from the source and y_{RD} from the relay:

$$y_{SD} = h_{SD} x_s + n_{SD}$$
$$y_{RD} = \frac{h_{SR} h_{RD}}{\sqrt{\overline{p}_y}} x_s + \frac{h_{RD}}{\sqrt{\overline{p}_y}} n_{SR} + n_{RD} \tag{3.13}$$

where $n_{SD}, n_{SR}, n_{RD} \in CN(0, N_0)$ and $\overline{p}_y$ is the average power of the received signals.

Then we sum two paths LLR in (3.14).

$$\begin{aligned}
LLR_{AF} &= LLR_{SD} + LLR_{RD} = \frac{2h_{SD}}{N_0} y_{SD} + \frac{2 \cdot \frac{h_{SR} h_{RD}}{\sqrt{\overline{p}_y}}}{N_0 \left(1 + \frac{h_{RD}^2}{\overline{p}_y}\right)} y_{RD} \\
&= \frac{2h_{SD}}{N_0}(h_{SD} x_s + n_{SD}) + \frac{2 \cdot \frac{h_{SR} h_{RD}}{\sqrt{\overline{p}_y}}}{N_0 \left(1 + \frac{h_{RD}^2}{\overline{p}_y}\right)} \left(\frac{h_{SR} h_{RD}}{\sqrt{\overline{p}_y}} x_s + \frac{h_{RD}}{\sqrt{\overline{p}_y}} n_{SR} + n_{RD}\right) \\
&= \frac{2 \cdot \left[h_{SD}^2 x_s + h_{SD} n_{SD} + \frac{\frac{h_{SR}^2 h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} x_s + \frac{\frac{h_{SR} h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} n_{SR} + \frac{\frac{h_{SR} h_{RD}}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} n_{RD} \right]}{N_0}
\end{aligned} \tag{3.14}$$

Then we arrange (3.14) to get (3.15)

$$\begin{aligned}
&\frac{2 \cdot \left[\left(h_{SD}^2 + \frac{\frac{h_{SR}^2 h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} \right) x_s + h_{SD} n_{SD} + \frac{\frac{h_{SR} h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} n_{SR} + \frac{\frac{h_{SR} h_{RD}}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} n_{RD} \right]}{N_0} \\
&= \frac{2 \cdot \left[x_s + \frac{\left(h_{SD} n_{SD} + \frac{\frac{h_{SR} h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} n_{SR} + \frac{\frac{h_{SR} h_{RD}}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} n_{RD} \right)}{\left(h_{SD}^2 + \frac{\frac{h_{SR}^2 h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} \right)} \times \left(h_{SD}^2 + \frac{\frac{h_{SR}^2 h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} \right) \right]}{N_0}
\end{aligned}$$

$$
= \frac{2 \cdot (x_{MRC}) \left(h_{SD}^2 + \frac{\frac{h_{SR}^2 h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} \right)}{N_0} = LLR_{MRC} \tag{3.15}
$$

Where $x_{MRC} = x_s + \frac{\left(h_{SD} n_{SD} + \frac{\frac{h_{SR} h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} n_{SR} + \frac{\frac{h_{SR} h_{RD}}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} n_{RD} \right)}{\left(h_{SD}^2 + \frac{\frac{h_{SR}^2 h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} \right)} = x_s + w,$

and $Var[w] = \frac{N_0}{2 \cdot \left(h_{SD}^2 + \frac{\frac{h_{SR}^2 h_{RD}^2}{\overline{p}_y}}{(1 + \frac{h_{RD}^2}{\overline{p}_y})} \right)}$

Thus, the LLR obtained by the summation of LLRs calculated from the two paths is equivalent to that calculated from the received signal after MRC.

3.4 MRC and Demapping in MQAM

In the last section, we discuss the LLRs of BPSK signals obtained by the summation of LLRs calculated from the two paths is equivalent to that calculated from the received signal after MRC. In the following subsection, we try to analysis in higher level constellation, and we show that the results for these approaches are different.

3.4.1 Demapping and Combining

From the reference [9], the soft bit value can express as (3.10):

$$LLR(b_{I,k}) = \frac{|h(i)|^2}{2\sigma^2}[\min_{\alpha \in S_{I,k}^{(0)}} |y[i]-\alpha|^2 - \min_{\alpha \in S_{I,k}^{(1)}} |y[i]-\alpha|^2] = \frac{2|h(i)|^2}{\sigma^2} D_{I,k}$$

For the in-phase bits of a 16QAM symbol, we have

$$D_{I,k} \approx \begin{cases} y_I(i) & , \quad k=1 \\ -|y_I(i)|+2 & , \quad k=2 \end{cases} \tag{3.17}$$

Then we sum two paths LLR:

$$LLR_{AF} = LLR_{SD} + LLR_{RD} = \frac{2 \cdot h_{SD}^2}{\sigma_{SD}^2} D_{I,k} + \frac{2 \cdot \dfrac{h_{SR}^2 h_{RD}^2}{\overline{p}_y}}{\dfrac{h_{RD}^2}{\overline{p}_y}\sigma_{SR}^2 + \sigma_{RD}^2} D_{I,k} \tag{3.18}$$

The definition of every symbol is as before, and we call this DC.

3.4.2 MRC and Demapping

From the last section 3.3, we find the y_{MRC} ,

$$y_{MRC} = \left(h_{SD}^2 + \frac{h_{SR}^2 h_{RD}^2}{\overline{p}_y + h_{RD}^2} \right) x_s + h_{SD} n_{SD} + \frac{h_{SR} h_{RD}^2}{\overline{p}_y + h_{RD}^2} n_{SR} + \frac{\sqrt{\overline{p}_y} h_{SR} h_{RD}}{\overline{p}_y + h_{RD}^2} n_{RD} = G_c \cdot x_s + w \ (3.19)$$

$$G_c = \left(h_{SD}^2 + \frac{h_{SR}^2 h_{RD}^2}{\overline{p}_y + h_{RD}^2} \right) \quad , \quad \text{variance of noise } w \text{ is } \sigma_n^2 = N_0 \left(h_{SD}^2 + \frac{h_{SR}^2 h_{RD}^4}{\left(\overline{p}_y + h_{RD}^2\right)^2} + \frac{\overline{p}_y h_{SR}^2 h_{RD}^2}{\left(\overline{p}_y + h_{RD}^2\right)^2} \right)$$

Where G_c is considered as equivalent channel coefficient, and w is three noise part combination. Then we can demap the y_{MRC} :

$$LLR(b_{I,k}) = \log \max_{x \in S_{I,k}^{(1)}} \exp\left\{ -\frac{1}{2} \cdot \frac{|y_{MRC} - G_c \cdot x_s|^2}{\sigma_n^2} \right\} - \log \max_{x \in S_{I,k}^{(0)}} \exp\left\{ -\frac{1}{2} \cdot \frac{|y_{MRC} - G_c \cdot x_s|^2}{\sigma_n^2} \right\} \tag{3.20}$$

We know that y_{MRC} can arrange to the form of (3.21):

$$r_{MRC} = x_s + \frac{w}{G_c} \tag{3.21}$$

Put this relation into (3.20), and then we can find the result and here we call it MD:

$$LLR(b_{I,k}) = \frac{|G_c|^2}{2\sigma_n^2} \left\{ \min_{x_I \in S_{I,k}^{(0)}} |r_{MRC,I} - x_{s,I}|^2 - \min_{x_I \in S_{I,k}^{(1)}} |r_{MRC,I} - x_{s,I}|^2 \right\} \tag{3.22}$$

Assume $D_{I,k} = \frac{1}{4} \left\{ \min_{x_I \in S_{I,k}^{(0)}} (r_{MRC,I} - x_{s,I}) - \min_{x_I \in S_{I,k}^{(1)}} (r_{MRC,I} - x_{s,I}) \right\}$

so the LLR can be formulate as : $LLR(b_{I,k}) = \frac{2|G_c|^2}{\sigma_n^2} D_{I,k}$

and in $16-QAM$, $D_{I,k} \approx \begin{cases} r_{MRC,I} \quad , & k=1 \\ -|r_{MRC,I}|+2 \quad , & k=2 \end{cases}$

3.4.3 Performance comparison

From Figure 3-4, if we choose a symbol (-1+j) , then the first bit is 0 and second is 1. If we consider the fist bit :

DC

$$LLR(b_{I,1}) = \frac{2}{N_0}\left\{(-1)\cdot\left(\frac{\sigma_{SR}^2+3}{\sigma_{SR}^2+2}\right) + \mathrm{Re}\left\{\frac{(\sigma_{SR}^2+2)n_{SD}+n_{SR}+\sqrt{\sigma_{SR}^2+1}n_{RD}}{\sigma_{SR}^2+2}\right\}\right\} \tag{3.23}$$

MD

$$LLR(b_{I,1}) = \frac{2}{N_0}\left\{(-1)\cdot\left(\frac{\sigma_{SR}^2+3}{\sigma_{SR}^2+2}\right) + \mathrm{Re}\left\{\frac{n_{SD}+n_{SR}+\sqrt{(\sigma_{SR}^2+1)}n_{RD}}{\sigma_{SR}^2+2}\right\}\right\} \tag{3.24}$$

From the (3.23) and (3.24), we can find that the means are both zero in the noise part. But the variance in DC is bigger than the variance in MD. Figure 3-5 shows the pdf of the first bit LLR in both approaches.

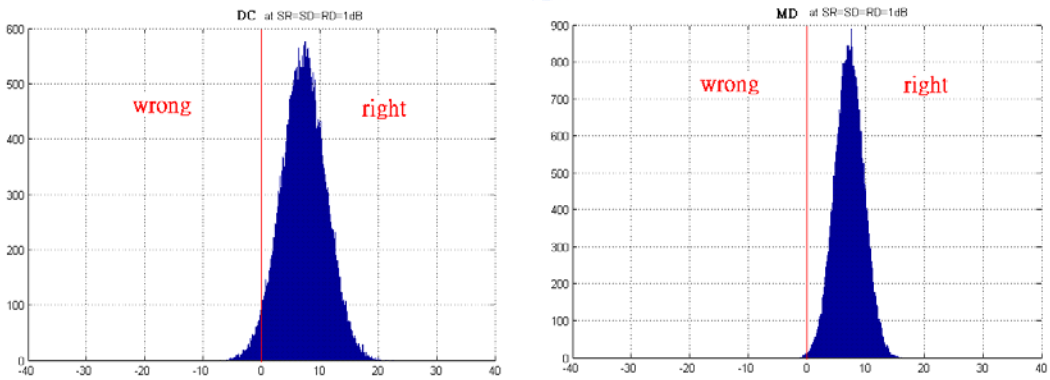


Figure 3-5: Pdf of the LLR in DC and MD

The result is MD is better than DC. In the latter simulations in Chapter 6, we will show the performance of these two different methods at the receiver.

3.5 Decode-and-Forward (DF)

In DF, when the relay has successfully decoded all the bits in the received packet, it re-encode a set of bits and re-transmit to the destination. DF typically includes an option to switch to the non-cooperation mode when the relay fails to decode the packet correctly. This is to prevent error propagation and improve overall system performance. In repetition-DF, the destination will combine the signals received from the source and the relay, i.e.

$$LLR_{DF}(i) = LLR_{SD}(i) + LLR_{RD}(i) = \frac{2|h_{SD}|^2}{\sigma_{SD}^2} D_{I,k} + \frac{2|h_{RD}|^2}{\sigma_{RD}^2} D_{I,k} \tag{3.25}$$

The definition of received signals and symbols in (3.25) is the same in AF. Figure 3-6 is the block diagram of DF.

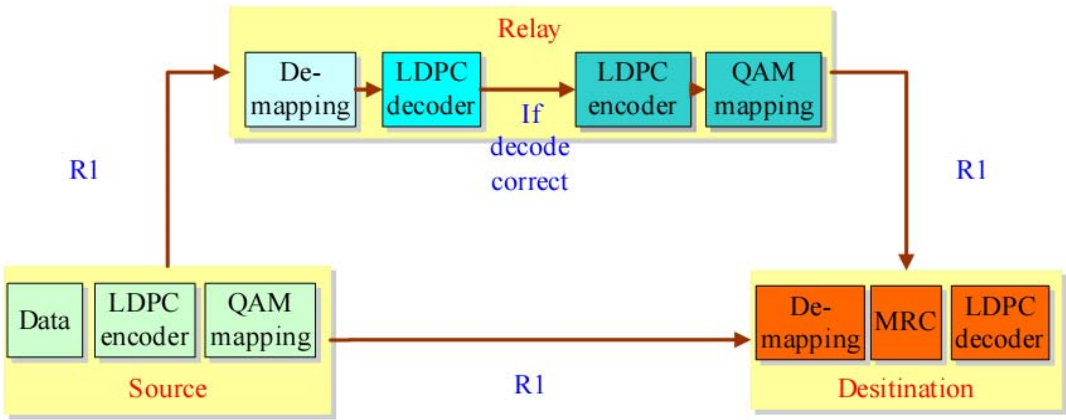


Figure 3-6 : DF block diagram

4 Gaussian Mixture Identification With EM Algorithm

We specify the maximum-likelihood parameter estimation problem and introduce the Expectation-Maximization (EM) algorithm to solve the parameter estimation problem in this chapter[10]. First we define the maximum-likelihood parameter estimation problem, then describe the EM algorithm, and finally use the EM algorithm to identify a Gaussian mixture.

4.1 Maximum-likelihood Estimation

The maximum-likelihood estimation problem can be defined as follows: We have a density function $p(x|\Theta)$ that is governed by the set of parameters Θ, and also have a data set of size N which drawn from this distribution, $\mathrm{X}=\{x_1,x_2,\ldots,x_N\}$. We assume that these data are independently and identically distributed (i.i.d.) with the density function p, so the density for the data set is

$$p(\mathrm{X}|\Theta)=\prod_{i=1}^{N}p(x_i|\Theta) \qquad (4.1)$$

Equation 4.1 is called the likelihood of the parameters given the data, which is thought of as a function of parameters Θ. Here, the data X is considered as fixed. Then in the maximum-likelihood estimation problem, the main goal is to find Θ_{opt} which maximizes (4.1).

$$\Theta_{opt}=\arg\max_{\Theta}\, p(\mathrm{X}|\Theta) \qquad (4.2)$$

To make the analysis easier, we often maximize $\log(p(\mathrm{X}|\Theta))$ instead of $p(x|\Theta)$.

Depending on the form of $p(x|\Theta)$, the problem could be easy or difficult. If $p(x|\Theta)$ is simply a single Gaussian distribution with parameter $\Theta=(\mu,\sigma^2)$, then we can just set the derivative of $\log(p(\mathrm{X} \mid \Theta))$ to zero, then directly find μ and σ^2. However, for many problems, it is difficult to find such analytical expressions, so we have to find more elaborate techniques to solve the problem.

4.2 Basic Expectation-maximization Estimation

EM algorithm is one such elaborate technique. The EM algorithm is generally used in statistics for finding maximum-likelihood estimates of parameters in probabilistic models, which is an underlying distribution from a given data set when the data is incomplete or has missing values. There are two main applications of the EM algorithm. The first occurs when the data indeed has missing values, due to problems with or limitations of the observed process. The second occurs when optimizing the likelihood function is analytically intractable but when the likelihood function can be simplified by assuming the existence of additional but missing (or hidden) parameters. The second application is the solution what we are concerned later.

We assume that a data set X is observed and is generated by some distribution, and we call it incomplete data. We also assume a complete data set Z=(X,Y), and a jointly density function arises from the marginal density function $p(x \mid \Theta)$ and the assumption of hidden variable and parameter guesses:

$$p(z \mid \Theta) = p(x, y \mid \Theta) = p(y \mid x, \Theta)\, p(x \mid \Theta) \tag{4.3}$$

Now we can define a complete-data likelihood function $p(\mathrm{X}, \mathrm{Y} \mid \Theta)$. Note that this function is a random variable since the missing information Y is unknown, random, and presumably governed by an underlying distribution. Since X and Θ can be seen as constants and Y is a random variable, the original complete-data likelihood function can be thought of some function as $p(\mathrm{X}, \mathrm{Y} \mid \Theta) = h_{\mathrm{X},\Theta}(\mathrm{Y})$.

The first step of the EM algorithm is finding the expected value of the complete-data log-likelihood $\log(p(\mathrm{X}, \mathrm{Y} \mid \Theta))$ with respect to the unknown data Y given the observed data X and the current parameter estimation.

$$Q(\Theta, \Theta^{(i-1)}) = E[\log(p(\mathrm{X}, \mathrm{Y}|\Theta)) \mid \mathrm{X}, \Theta^{(i-1)}] \qquad (4.4)$$

Where $\Theta^{(i-1)}$ are the current parameters, Θ are the new parameter. We use $\Theta^{(i-1)}$ to evaluate the expectation and optimize Θ to increase Q. Note that in (4.4), X and $\Theta^{(i-1)}$ are constants, Θ is a normal variable that we want to adjust and Y is a random variable governed by the distribution $f(y \mid \mathrm{X}, \Theta^{(i-1)})$. Then the right-hand side of (4.4) can be rewritten as (4.5):

$$E[\log(p(\mathrm{X}, \mathrm{Y}|\Theta)) \mid \mathrm{X}, \Theta^{(i-1)}] = \int_{y \in \gamma} \log(p(\mathrm{X}, \mathrm{Y}|\Theta)) f(y \mid \mathrm{X}, \Theta^{(i-1)}) dy \qquad (4.5)$$

Where $f(y \mid \mathrm{X}, \Theta^{(i-1)})$ is the marginal distribution of the unobserved data which is dependent on both the observed data X and the current parameters $\Theta^{(i-1)}$, and γ is the region where the Y can take on. If this marginal distribution is a simple analytical expression of the assumed parameters $\Theta^{(i-1)}$ and perhaps the data, the problem will be easier to solve. However, sometimes this density function might be difficult to find. The evaluation of this expectation is called the E-step in the EM algorithm.

The second step of the EM algorithm, M-step, has a goal to maximize the expectation we computed in the E-step. Mathematically, it can be expressed as,

$$\Theta^{(i)} = \arg\max_{\Theta} Q(\Theta, \Theta^{(i-1)}) \qquad (4.6)$$

Then, the two steps are iterated, and it has been shown that each iteration is guaranteed to increase the log-likelihood. The EM algorithm will converge to a local maximum of the likelihood function. There are many works discussing the convergence problem, but we will not pursue that here. From the description we give, it is not very clear how to exactly conduct the EM algorithm, this is because the details of the steps need to compute the given quantities which are strongly dependent on the particular application considered.

4.3 Gaussian Mixture Identification via EM Algorithm

The mixture-density parameter estimation problem is one of the most widely used applications of the EM algorithm. For this case, we have the density to be identified as

$$p(x \mid \Theta) = \sum_{i=1}^{M} \alpha_i p_i(x \mid \theta_i) \tag{4.7}$$

Where the parameters are $\Theta = \{\alpha_1, \alpha_2, \ldots, \alpha_M, \theta_1, \theta_2, \ldots, \theta_M\}$, $\sum_{i=1}^{M} \alpha_i = 1$, and p_i is a density function parameterized by θ_i. It can be also considered as we have M component densities mixed together with mixing coefficients α_i.

$$\log(p(\mathrm{X} \mid \Theta)) = \log \prod_{i=1}^{N} p(x_i \mid \Theta) = \sum_{i=1}^{N} \log(\sum_{j=1}^{M} \alpha_j p_j(x_i \mid \theta_j)) \tag{4.8}$$

Even with the Gaussian assumption of $p_j(.)$, (4.8) is difficult to maximize since it is highly nonlinear. If we consider X as incomplete, and posit the existence of unobserved data $\mathrm{Y=\{y_i\}_{i=1}^{N}}$ whose values indicating which the component density generated each data item, the likelihood expression can be significantly simplified and the solution is easier to obtain. Assuming that $y_i \in \{1, 2, \ldots, M\}$ for each i, and $y_i = k$ if the i_{th} sample is generated by the k_{th} mixture component. If we know the data Y, the likelihood can be expressed as:

$$\log(p(\mathrm{X}, \mathrm{Y} \mid \Theta)) = \sum_{i=1}^{N} \log(p(x_i \mid y_i) p(y_i)) = \sum_{i=1}^{N} \log(\alpha_{y_i} p_{y_i}(x_i \mid \theta_{y_i})) \tag{4.9}$$

which is a particular form of the component densities, and it can be easily optimized using a variety of techniques. However, we do not know the values of Y. If we assume Y is a random vector, we can proceed.

First, we must derive an expression for the distribution of the unobserved data. In the beginning, we guess that $\Theta^g = \{\alpha_1^g, \alpha_2^g, \ldots, \alpha_M^g, \theta_1^g, \theta_2^g, \ldots, \theta_M^g\}$. Given Θ^g, we can easily compute $p_j(x_i \mid \theta_j^g)$ for each i and j. Besides, the mixing parameters α_j can be thought of as the priori probability of each mixture component, that is $\alpha_j = p(component\ j)$. Using Bayes' rule, we can know:

$$p(y_i \mid x_i, \Theta^g) = \frac{\alpha_{y_i}^g p_{y_i}(x_i \mid \theta_{y_i}^g)}{p(x_i \mid \theta^g)} = \frac{\alpha_{y_i}^g p_{y_i}(x_i \mid \theta_{y_i}^g)}{\sum_{k=1}^{M} \alpha_k^g p_k(x_i \mid \theta_k^g)} \tag{4.10}$$

and

$$p(y \mid \mathrm{X}, \Theta^g) = \prod_{i=1}^{N} p(y_i \mid x_i, \Theta^g) \tag{4.11}$$

Where $y = (y_1, y_2, \ldots, y_N)$ is the independent unobserved data.

For the Gaussian mixture case, (4.4) can be expressed as:

$$\begin{aligned}
Q(\Theta, \Theta^g) &= \sum_{y \in \gamma} \log(p(X, y \mid \Theta)) p(y \mid X, \Theta^g) \\
&\overset{\substack{with\,(4.9) \\ and\,(4.11)}}{=} \sum_{y \in \gamma} \sum_{i=1}^{N} \log(\alpha_{y_i} p_{y_i}(x_i \mid \theta_{y_i})) \prod_{j=1}^{N} p(y_j \mid x_j, \Theta^g) \\
&= \sum_{y_1=1}^{M} \sum_{y_2=1}^{M} \cdots \sum_{y_N=1}^{M} \sum_{i=1}^{N} \log(\alpha_{y_i} p_{y_i}(x_i \mid \theta_{y_i})) \prod_{j=1}^{N} p(y_j \mid x_j, \Theta^g) \\
&= \sum_{y_1=1}^{M} \sum_{y_2=1}^{M} \cdots \sum_{y_N=1}^{M} \sum_{i=1}^{N} \sum_{l=1}^{M} \delta_{l, y_i} \log(\alpha_l p_l(x_i \mid \theta_l)) \prod_{j=1}^{N} p(y_j \mid x_j, \Theta^g) \\
&= \sum_{l=1}^{M} \sum_{i=1}^{N} \log(\alpha_l p_l(x_i \mid \theta_l)) \sum_{y_1=1}^{M} \sum_{y_2=1}^{M} \cdots \sum_{y_N=1}^{M} \delta_{l, y_i} \prod_{j=1}^{N} p(y_j \mid x_j, \Theta^g)
\end{aligned} \tag{4.12}$$

Then, we can simplify (4.12) as

$$\begin{aligned}
&\sum_{y_1=1}^{M} \sum_{y_2=1}^{M} \cdots \sum_{y_N=1}^{M} \delta_{l, y_i} \prod_{j=1}^{N} p(y_j \mid x_j, \Theta^g) \\
&= \left(\sum_{y_1=1}^{M} \cdots \sum_{y_{i-1}=1}^{M} \sum_{y_{i+1}=1}^{M} \sum_{y_N=1}^{M} \prod_{j=1, j \neq i}^{N} p(y_j \mid x_j, \Theta^g) \right) p(l \mid x_i, \Theta^g) \\
&= \prod_{j=1, j \neq i}^{N} \left(\sum_{y_j=1}^{M} p(y_j \mid x_j, \Theta^g) \right) p(l \mid x_i, \Theta^g) = p(l \mid x_i, \Theta^g)
\end{aligned} \tag{4.13}$$

Note that $l \in 1, 2, \ldots, M$, and that $\sum_{i=1}^{M} p(i \mid x_j, \Theta^g) = 1$. Using (4.13), we can write

(4.12) as (4.14):

$$Q(\Theta,\Theta^g)=\sum_{l=1}^{M}\sum_{i=1}^{N}\log(\alpha_l p_l(x_i\,|\,\theta_l))p(l\,|\,x_i,\Theta^g)$$
$$=\sum_{l=1}^{M}\sum_{i=1}^{N}\log(\alpha_l)p(l\,|\,x_i,\Theta^g)+\sum_{l=1}^{M}\sum_{i=1}^{N}\log(p_l(x_i\,|\,\theta_l))p(l\,|\,x_i,\Theta^g) \quad (4.14)$$

In order to maximize (4.14), we can maximize the term containing α_l and the term containing θ_l independently since they are not uncorrelated.

To find α_l, we introduce the Lagrange multiplier λ with the constraint that $\sum_l \alpha_l = 1$. The Lagrange multipliers provide a strategy for finding the maximum/minimum of a function subject to constraints. For example, if we want to maximize $f(x,y)$, and the constraint is $g(x,y)=c$, then the cost function can be re-defined with the Lagrange multiplier λ as follows:

$$\Lambda(x,y,\lambda)=f(x,y)+\lambda(g(x,y)-c) \quad (4.15)$$

Now, we can solve the following equation for α_l:

$$\frac{\partial}{\partial\alpha_l}[\sum_{l=1}^{M}\sum_{i=1}^{N}\log(\alpha_l)p(l\,|\,x_i,\Theta^g)+\lambda(\sum_l \alpha_l-1)]=0 \quad (4.16)$$

or

$$\sum_{i=1}^{N}\frac{1}{\alpha_l}p(l\,|\,x_i,\Theta^g)+\lambda=0 \quad (4.17)$$

Summing both sides over l, we find that $\lambda=-N$. Then we can get the expression for α_l as

$$\alpha_l=\frac{1}{N}\sum_{i=1}^{N}p(l\,|\,x_i,\Theta^g) \quad (4.18)$$

Note that in our scenario, we do not need to consider the parameters α_l since we assume that all component densities mixed together with the same mixing coefficient α_l.

We then want to find θ_l. For some distributions, it is possible to get an analytical expression for θ_l. In our scenario, the distribution is a mixture of two one-dimensional

Gaussian distributions with mean $\mu_1 = -\mu_2$ and variance $\sigma_1^2 = \sigma_2^2$, which is shown below:

$$p_l(x \mid \mu_l, \sigma_l^2) = \frac{1}{\sqrt{2\pi\sigma_l^2}} e^{-\frac{(x-\mu_l)^2}{2\sigma_l^2}} \quad , \quad l \in \{1,2\} \tag{4.19}$$

Taking the logarithm of (4.19), ignoring constant terms, and substituting the result into the right side last term of (4.14), we can obtain

$$\begin{aligned} &\sum_{l=1}^{M}\sum_{i=1}^{N} \log(p_l(x_i \mid \theta_l)) p(l \mid x_i, \Theta^g) \\ &= \sum_{l=1}^{M}\sum_{i=1}^{N} \left(-\frac{1}{2}\log(\sigma_l^2) - \frac{(x-\mu_l)^2}{2\sigma_l^2} \right) p(l \mid x_i, \Theta^g) && (4.20) \\ &= \sum_{l=1}^{M} \left(\sum_{i=1}^{N} \frac{1}{2}\log(\sigma_l^2)^{-1} p(l \mid x_i, \Theta^g) - \sum_{i=1}^{N} \frac{(x_i-\mu_l)^2}{2\sigma_l^2} p(l \mid x_i, \Theta^g) \right) && (4.21) \end{aligned}$$

Taking the derivative of (4.20) with respect to μ_l and setting it equal to zero,

$$\sum_{i=1}^{N} \frac{(x-\mu_l)}{\sigma_l^2} p(l \mid x_i, \Theta^g) = 0 \tag{4.22}$$

we can then easily solve for μ_l,

$$\mu_l = \frac{\sum_{i=1}^{N} x_i p(l \mid x_i, \Theta^g)}{\sum_{i=1}^{N} p(l \mid x_i, \Theta^g)} \tag{4.23}$$

We can use the same method to derive the estimation of the variance. Using the (4.21), we take derivative with respect to $(\sigma_l^2)^{-1}$ and set the result to zero,

$$\sum_{i=1}^{N} \frac{1}{2}\log \sigma_l^2 \, p(l \mid x_i, \Theta^g) - \sum_{i=1}^{N} \frac{(x_i-\mu_l)^2}{2\sigma_l^2} p(l \mid x_i, \Theta^g) = 0 \tag{4.24}$$

Finally, we can obtain the estimation for σ_l^2:

$$\sigma_l^2 = \frac{\sum_{i=1}^{N} (x_i-\mu_l)^2 p(l \mid x_i, \Theta^g)}{\sum_{i=1}^{N} p(l \mid x_i, \Theta^g)} \tag{4.25}$$

Now, a complete EM algorithm for the Gaussian mixture identification is derived. To sum up, the E-step finds the expected value of the complete-data log-likelihood. The M-step obtains a new estimation by maximizing the expectation computed in the E-step. The estimation of the new parameters in terms of the old parameters is summarized as below:

$$\alpha_l^{new} = \frac{1}{N}\sum_{i=1}^{N} p(l \mid x_i, \Theta^g)\ ; \quad \mu_l^{new} = \frac{\sum_{i=1}^{N} x_i\, p(l \mid x_i, \Theta^g)}{\sum_{i=1}^{N} p(l \mid x_i, \Theta^g)}\ ; \quad \sigma_l^{2new} = \frac{\sum_{i=1}^{N} (x_i - \mu_l^{new})^2\, p(l \mid x_i, \Theta^g)}{\sum_{i=1}^{N} p(l \mid x_i, \Theta^g)} \tag{4.26}$$

With the EM algorithm, it is guaranteed to increase the log-likelihood and converge to a local maximum of the likelihood function. Since the EM algorithm is not guaranteed to find the global maximum, we need to choose the initial values carefully.

5 Compress and Forward in User Cooperation

In Chapter 3, we have described two cooperative protocols, i.e., AF and DF. In this chapter, we investigate another cooperative scheme, called compress-and-forward (CF). In CF, the relay forwards the quantized/observed/estimated version of its observations. In [11], the CF for turbo decoder was introduced. In this chapter, we will extend its use to the LDPC decoder.

Since the SR channel may have deep fades with a high probability, the DF scheme cannot operate in the cooperative mode all the time and this will cause performance degradation. The AF does not have the problem. However, the SR channel may be noisy and retransmission will further amplify the noise. The CF can alleviate the problems mentioned above. Under the CF, the relay, whether the decoding is successful or not, retransmits the information from the source to the destination. Then the destination can combine both the LLR from the source and the relay for data detection. We will have more details in the later section.

5.1 Compress-and-forward (CF) Cooperation Strategy

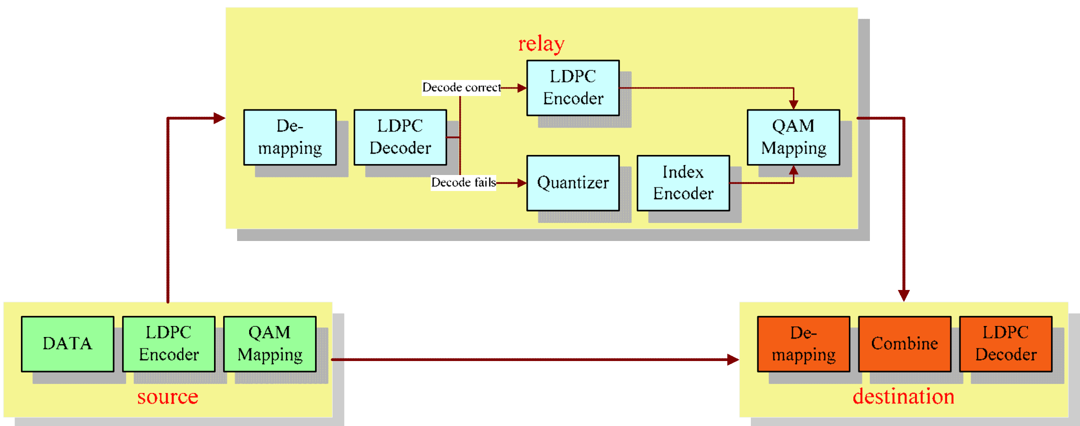


Figure 5-1: Block diagram of hybrid compress-and-forward (CF)

Figure 5-1 shows the block diagram of the hybrid CF cooperative strategy, but for simplicity, we just call it CF. In DF, when the decoding in the relay fails, the relay switches to the non-cooperation mode. As a result, the destination only have the information from the source; However, in the CF scheme, when the decoding fails, the relay will switches to a mode which will retransmit the quantized LLR information to the destination. Figure 5-2 shows the CF cooperative protocol.

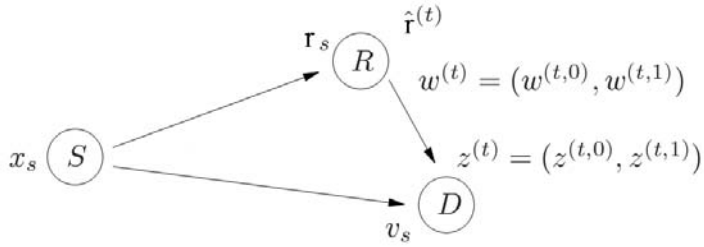


Figure 5-2: CF for BPSK modulation

In Figure 5-2, x_s denotes the transmit signal at the source, v_s its received signal at the destination, $\hat{r}^{(t)}$ the LLR of a decoded bit at the relay, and $w^{(t)}$ an modulated index for the quantized LLR (here only one bit quantization), and $z^{(t)}$ the received index at the destination. Then the destination collects v_s and $z^{(t)}$ to recover the transmit data. Here, we assume that the LDPC code is used at the source. The overall approach can be summarized as follows:

1) In the first time slot, the source broadcasts signal x_s to the relay and the destination simultaneously.
2) The relay performs LDPC decoding to estimate x_s. If x_s is decoded successfully, the relay use traditional DF scheme. If the decoding fails, the relay quantizes the LDPC-decoder LLR, encodes the index, conduct symbol mapping with BPSK, QPSK, or M-QAM, and transmits the resultant signal to the destination.
3) The destination combines the information received form the source and relay to recovery the information bits.

Note that in the second time slot, an indicating bit may be piggybacked on the relay packet, so the destination knows if information bits or LLR indices are re-transmitted. Besides, if the error rate at the relay is too high, we may also switch to the non- cooperative mode. This is because if the SR channel is very poor, not much information can be explored in the relay.

5.2 System Model

Here we consider the same scenario as that in Chapter 3, a typical three-node relay system. Let the channels be block Rayleigh fading, the variances of all channels be one, and all noises are AWGNs, The received signal at the relay and the destination can be expressed as follow:

$$r_{SR}(i) = h_{SR}x_S(i) + n_{SR}(i) \tag{5.1}$$

$$r_{SD}(i) = h_{SD}x_S(i) + n_{SD}(i) \tag{5.2}$$

$$r_{RD}(i) = h_{RD}x_R(i) + n_{RD}(i) \tag{5.3}$$

where x_S is the transmitted signal from the source, x_R is the transmitted signal from the relay, r_{SR} and n_{SR} are the received signal and noise at the relay, r_{SD},, r_{RD} and n_{SD}, n_{RD} are the received signals and noise at the destination (first and second time slot). All noises have zero mean and the variance of, n_{SD}, n_{SR} , n_{RD} are σ_{SD}^2, σ_{SR}^2 and σ_{RD}^2, respectively.

For decoding, we have to calculate the channel LLR upon receiving the signal at the relay and destination. For retransmission at the relay, we have to calculate the decoder LLR. For a received signal r , the LLR can be written as:

$$LLR_r = \log\frac{p(r\,|\,x=+1)}{p(r\,|\,x=-1)} = \frac{2h^2}{\sigma^2}x + n_r \quad , \quad n_r \sim N(0,\sigma_r^2) \tag{5.4}$$

Where LLR_r is the demapping value for each bit in a LDPC code block, and x is BPSK signal. Note that we have assumed $p(x=+1) = p(x=-1) = 0.5$. For the decoder

LLR $\hat{r}$, we can model it as a binary signal corrupted by a Gaussian noise, i.e,:

$$LLR_{\hat{r}} = \mu_{\hat{r}} x + n_{\hat{r}} \tag{5.5}$$

Where $LLR_{\hat{r}}$ is the decoder LLR, $\mu_{\hat{r}}$ is the mean of LLRs pdf, and the noise variance $n_{\hat{r}} \sim N(0,\sigma_{\hat{r}}^2)$. The probability density function (PDF) of the decoder LLR is then

$$\begin{aligned} p_{x=+1}(\hat{r}) = p_{+1}(\hat{r}) &= \frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}} \exp\left\{\frac{-(\hat{r}-\mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2}\right\} \\ p_{x=-1}(\hat{r}) = p_{-1}(\hat{r}) &= \frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}} \exp\left\{\frac{-(\hat{r}+\mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2}\right\} \end{aligned} \tag{5.6}$$

In the next section, we will discuss the quantizer and index encoder at the relay.

5.3 Quantizer Optimization [11]

In this section, we address how to quantize $\hat{r}^{(t)}$ at the relay, where the superscript t is the index of a signal in a packet. In CF, an index encoder (IE) typically succeeds the quantizer to compress the indices of the quantization bins for further rate reduction. Design of a CF quantizer needs to consider the index encoder type. Here for simplicity, we consider a fixed-rate index encoding. We discuss the design procedure by considering a four-level scalar quantizer. However, one can extend the method to a higher-level quantiztier.

Let $\{u_i, i = 0,1,2,3,4\}$ be the bin-boundaries where u_0 and u_4 are set to be $-\infty$ and ∞ respectively. We assume that the value of $\hat{r}$ has a symmetric PDF with respect to the origin. Due to the symmetric property, it is reasonable to let $u_1 = -u_d$, $u_2 = 0$, $u_3 = u_d$, where u_d (> 0) is to be determined. Using the scheme, each soft output of the LDPC decoder is mapped to a two-bit bin-index $w^{(t)} = (w^{(t,0)}, w^{(t,1)})$. Let $u_l^{(t)} \in \{u_0,u_1,u_2,u_3,u_4\}$ and $u_h^{(t)} \in \{u_0,u_1,u_2,u_3,u_4\}$ denote the low-end bin-boundary and the high-end bin-boundary for $\hat{r}^{(t)}$. Figure 5-3 shows the bin boundaries and an example of the index-encoder. For a specific bin-index $w^{(t)}$, we have

$$p(w^{(t)} \mid x_s^{(t)}) = \int_{u_l^{(t)}}^{u_h^{(t)}} p_{x_s^{(t)}}(\hat{r})d\hat{r} = \int_{u_l^{(t)}}^{u_h^{(t)}} p_{\pm 1}(\hat{r})d\hat{r} \tag{5.7}$$

and

$$p(w^{(t)}) = \int_{u_l^{(t)}}^{u_h^{(t)}} p(\hat{r})d\hat{r} \tag{5.8}$$

Where $x_s^{(t)}$ is the transmitted signal from the source, and the superscript *t* is the index of a signal in a packet.

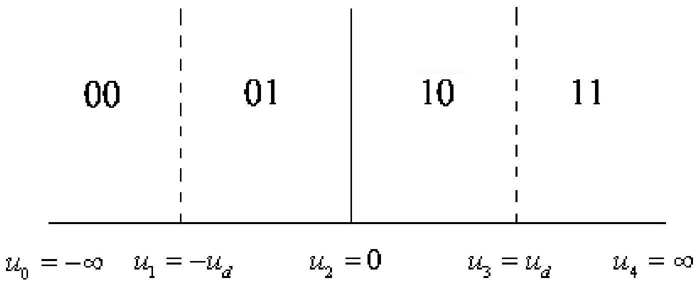


Figure 5-3: Bin boundaries and index-encoder

The general design goal for a CF scheme, as well as other cooperative schemes, is for the relay to maximize the amount of "new" information about signal x_s. Where the new, we mean non-overlap information that complements the information conveyed directly to the destination by the source. Mathematically, this criterion to maximize can be expressed as,

$$\arg\min_{u_d} H(x_s^{(t)} \mid w^{(t)}) \tag{5.9}$$

Where $H(x_s^{(t)} \mid w^{(t)})$ is the conditional entropy defined as:

$$H(x_s^{(t)} \mid w^{(t)}) = -\sum_{x_s^{(t)}, w^{(t)}} p(x_s^{(t)}, w^{(t)}) \log p(x_s^{(t)} \mid w^{(t)}) \tag{5.10}$$

and where

$$p(x_s^{(t)}, w^{(t)}) = p(w^{(t)} \mid x_s^{(t)}) p(x_s^{(t)}) \tag{5.11}$$

$$p(x_s^{(t)} \mid w^{(t)}) = p(x_s^{(t)}, w^{(t)}) / p(w^{(t)}) \tag{5.12}$$

Substituting (5.11) and (5.12) into (5.10), we can have

$$
\begin{aligned}
H(x_s^{(t)} \mid w^{(t)}) &= -\sum_{x_s^{(t)}, w^{(t)}} p(w^{(t)} \mid x_s^{(t)}) p(x_s^{(t)}) \cdot \log \frac{p(x_s^{(t)}, w^{(t)})}{p(w^{(t)})} \\
&= -\sum_{x_s^{(t)}, w^{(t)}} \int_{u_l^{(t)}}^{u_h^{(t)}} p_{x_s^{(t)}}(\hat{r})\, d\hat{r} \cdot p(x_s^{(t)}) \cdot \log \frac{p(w^{(t)} \mid x_s^{(t)}) p(x_s^{(t)})}{\int_{u_l^{(t)}}^{u_h^{(t)}} p(\hat{r})\, d\hat{r}} \\
&= -\sum_{x_s^{(t)}, w^{(t)}} \int_{u_l^{(t)}}^{u_h^{(t)}} p_{x_s^{(t)}}(\hat{r})\, d\hat{r} \cdot p(x_s^{(t)}) \cdot \log \frac{\int_{u_l^{(t)}}^{u_h^{(t)}} p_{x_s^{(t)}}(\hat{r})\, d\hat{r} \cdot p(x_s^{(t)})}{\int_{u_l^{(t)}}^{u_h^{(t)}} p(\hat{r})\, d\hat{r}} \\
&= -\sum_{x_s^{(t)}, w^{(t)}} \int_{u_l^{(t)}}^{u_h^{(t)}} \frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}} \exp\left\{ \frac{-(\hat{r} \mp \mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2} \right\} d\hat{r} \times \frac{1}{2} \\
&\quad \times \left[\log \int_{u_l^{(t)}}^{u_h^{(t)}} \frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}} \exp\left\{ \frac{-(\hat{r} \mp \mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2} \right\} d\hat{r} \times \frac{1}{2} \right. \\
&\quad \left. - \log \int_{u_l^{(t)}}^{u_h^{(t)}} \frac{1}{2} \times \left(\exp\left\{ \frac{-(\hat{r} - \mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2} \right\} + \exp\left\{ \frac{-(\hat{r} + \mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2} \right\} \right) d\hat{r} \right]
\end{aligned} \tag{5.13}
$$

We observe from (5.13) that $H(x_s^{(t)} \mid w^{(t)})$ is a function of u_d, the mean of LLR $\mu_{\hat{r}}$ and the variance $\sigma_{\hat{r}}^2$. With the LLRs of a packet available, $\mu_{\hat{r}}$ and $\sigma_{\hat{r}}^2$ can be computed by the EM algorithm described in Chapter 4.

5.4 LLR Computation at Destination

Another important issue in CF is how the destination exploits the information received from the relay. As mentioned, the LLR is what we need for LDPC decoding. In this section, we will derive the formula to compute the LLR at the destination using the observations. In Figure 5-2 , $z^{(t)}$ and $v_s^{(t)}$ Define $z^{(t)} = (z^{(t,0)}, z^{(t,1)})$, $d^k = \{d^{(1)}, d^{(2)}, \ldots, d^{(k)}\}$, and $d^{(t)} = \{v_s^{(t)}, z^{(t)}\} = \{v_s^{(t)}, z^{(t,0)}, z^{(t,1)}\}$, where k is the length of the signal in one packet.

5.4.1 BPSK Modulation at the Relay

First, we consider a simple case where BPSK is used at the relay for the modulation of the quantized LLR. Let

$$
\begin{aligned}
p(z^{(t)} \mid x^{(t)}=1) &= \sum_{w^{(t)}} p(z^{(t)} \mid w^{(t)})\, p(w^{(t)} \mid x^{(t)}=1) \\
p(z^{(t)} \mid x^{(t)}=0) &= \sum_{w^{(t)}} p(z^{(t)} \mid w^{(t)})\, p(w^{(t)} \mid x^{(t)}=0)
\end{aligned}
\tag{5.14}
$$

and

$$p(z^{(t)} \mid w^{(t)}) = p(z^{(t,0)} \mid w^{(t,0)})\, p(z^{(t,1)} \mid w^{(t,1)}) \tag{5.15}$$

$$p(w^{(t)} \mid x_s^{(t)}) = \int_{u_l^{(t)}}^{u_h^{(t)}} p_{x_s}^{(t)}(\hat{r})\,d\hat{r} = \int_{u_l^{(t)}}^{u_h^{(t)}} p_{\pm 1}^{(t)}(\hat{r})\,d\hat{r} \tag{5.16}$$

$$z'^{(t)} = \frac{z^{(t)}}{h_{RD}} \tag{5.17}$$

Where $z^{(t,0)}$, $z^{(t,1)}$ are the first bit and the second bit quantized index received from the relay, and h_{RD} is the CSI of R-D.

The LLR we want to calculate is

$$LLR(x^{(t)}) = \log\frac{p(x^{(t)}=1 \mid d^{(t)})}{p(x^{(t)}=0 \mid d^{(t)})} = \log\frac{p(x^{(t)}=1 \mid v^{(t)}, z^{(t)})}{p(x^{(t)}=0 \mid v^{(t)}, z^{(t)})} \tag{5.18}$$

Note that $v^{(t)}$ is the channel LLR of $v_s^{(t)}$ and which is computed by the destination. Assuming that $p(x^{(t)}=1) = p(x^{(t)}=-1) = 0.5$, we can combine (5.14), (5.15), (5.16), (5.17), and (5.18) to obtain the LLR which combined the information from the source and the relay at the destination.

$$
\begin{aligned}
LLR(x^{(t)}) &= \log\frac{p(x^{(t)}=1 \mid d^{(t)})}{p(x^{(t)}=-1 \mid d^{(t)})} = \log\frac{p(x^{(t)}=1 \mid v^{(t)}, z^{(t)})}{p(x^{(t)}=-1 \mid v^{(t)}, z^{(t)})} \\
&= \log\frac{p(v^{(t)}, z^{(t)} \mid x^{(t)}=1)\, p(x^{(t)}=1)}{p(v^{(t)}, z^{(t)} \mid x^{(t)}=-1)\, p(x^{(t)}=-1)}
\end{aligned}
$$

Due to $v^{(t)}$ and $z^{(t)}$ are independent, we can arrange the equation as:

$$\log\frac{p(v^{(t)} \mid x^{(t)}=1)\, p(z^{(t)} \mid x^{(t)}=1)}{p(v^{(t)} \mid x^{(t)}=-1)\, p(z^{(t)} \mid x^{(t)}=-1)} \tag{5.19}$$

Then use the (5.15) and (5.16),

$$\log\frac{\sum_{w^{(t)}} p(v^{(t)}\mid x^{(t)}=1)\,p(z^{(t,0)}\mid w^{(t,0)})\,p(z^{(t,1)}\mid w^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} p_{+1}^{(t)}(\hat{r})d\hat{r}}{\sum_{w^{(t)}} p(v^{(t)}\mid x^{(t)}=-1)\,p(z^{(t,0)}\mid w^{(t,0)})\,p(z^{(t,1)}\mid w^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} p_{-1}^{(t)}(\hat{r})d\hat{r}}$$

$$=\log\frac{p(v^{(t)}\mid x^{(t)}=1)}{p(v^{(t)}\mid x^{(t)}=-1)}+\log\frac{\sum_{w^{(t)}} p(z^{(t,0)}\mid w^{(t,0)})\,p(z^{(t,1)}\mid w^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} p_{+1}^{(t)}(\hat{r})d\hat{r}}{\sum_{w^{(t)}} p(z^{(t,0)}\mid w^{(t,0)})\,p(z^{(t,1)}\mid w^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} p_{-1}^{(t)}(\hat{r})d\hat{r}} \qquad (5.20)$$

Here we can find the first part in (5.20) is the same as the LLR we consider in Chapter 3, so we can rewrite as:

$$LLR(x^{(t)})=\frac{2\,|h_{SD}|^2}{\sigma_{SD}^2}v^{'(t)}+\log\frac{\sum_{w^{(t)}} p(z^{(t,0)}\mid w^{(t,0)})\,p(z^{(t,1)}\mid w^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} p_{+1}^{(t)}(\hat{r})d\hat{r}}{\sum_{w^{(t)}} p(z^{(t,0)}\mid w^{(t,0)})\,p(z^{(t,1)}\mid w^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} p_{-1}^{(t)}(\hat{r})d\hat{r}} \qquad (5.21)$$

Where $v^{'(t)}$ is the value through channel equalizer from $v^{(t)}$. Then we can get (5.22):

$$LLR(x^{(t)})=\frac{2\,|h_{SD}|^2}{\sigma_{SD}^2}v^{'(t)}+\log\frac{\sum_{w^{(t)}} p(z^{(t,0)}\mid w^{(t,0)})\,p(z^{(t,1)}\mid w^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} \frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}}\exp\left\{-\frac{(\hat{r}-\mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2}\right\}d\hat{r}}{\sum_{w^{(t)}} p(z^{(t,0)}\mid w^{(t,0)})\,p(z^{(t,1)}\mid w^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} \frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}}\exp\left\{-\frac{(\hat{r}+\mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2}\right\}d\hat{r}}$$

$$=\frac{2\,|h_{SD}|^2}{\sigma_{SD}^2}v^{'(t)}$$

$$+\log\frac{\sum_{w^{(t)}}\frac{1}{\sqrt{2\pi\sigma_{RD}^2}}\exp\left\{-\frac{(z^{(t,0)}-w^{(t,0)})^2}{2\sigma_{RD}^2}\right\}\frac{1}{\sqrt{2\pi\sigma_{RD}^2}}\exp\left\{-\frac{(z^{(t,1)}-w^{(t,1)})^2}{2\sigma_{RD}^2}\right\}\int_{u_l^{(t)}}^{u_h^{(t)}}\frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}}\exp\left\{-\frac{(\hat{r}-\mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2}\right\}d\hat{r}}{\sum_{w^{(t)}}\frac{1}{\sqrt{2\pi\sigma_{RD}^2}}\exp\left\{-\frac{(z^{(t,0)}-w^{(t,0)})^2}{2\sigma_{RD}^2}\right\}\frac{1}{\sqrt{2\pi\sigma_{RD}^2}}\exp\left\{-\frac{(z^{(t,1)}-w^{(t,1)})^2}{2\sigma_{RD}^2}\right\}\int_{u_l^{(t)}}^{u_h^{(t)}}\frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}}\exp\left\{-\frac{(\hat{r}+\mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2}\right\}d\hat{r}}$$

$$=\frac{2\,|h_{SD}|^2}{\sigma_{SD}^2}v^{'(t)}$$

$$+\log\frac{\sum_{w^{(t)}}\exp\left\{-\frac{(z^{(t,0)}-w^{(t,0)})^2+(z^{(t,1)}-w^{(t,1)})^2}{2\sigma_{RD}^2}\right\}\int_{u_l^{(t)}}^{u_h^{(t)}}\frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}}\exp\left\{-\frac{(\hat{r}-\mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2}\right\}d\hat{r}}{\sum_{w^{(t)}}\exp\left\{-\frac{(z^{(t,0)}-w^{(t,0)})^2+(z^{(t,1)}-w^{(t,1)})^2}{2\sigma_{RD}^2}\right\}\int_{u_l^{(t)}}^{u_h^{(t)}}\frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}}\exp\left\{-\frac{(\hat{r}+\mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2}\right\}d\hat{r}}$$

$$=\frac{2\,|h_{SD}|^2}{\sigma_{SD}^2}v^{'(t)}$$

$$+\log\sum_{w^{(t)}}\left[\exp\left\{-\frac{|h_{RD}|^2\left((z^{'(t,0)}-w^{(t,0)})^2+(z^{'(t,1)}-w^{(t,1)})^2\right)}{2\sigma_{RD}^2}\right\}\cdot P_{+1}^{w^{(t)}}\right] \qquad (5.22)$$

$$-\log\sum_{w^{(t)}}\left[\exp\left\{-\frac{|h_{RD}|^2\left((z^{'(t,0)}-w^{(t,0)})^2+(z^{'(t,1)}-w^{(t,1)})^2\right)}{2\sigma_{RD}^2}\right\}\cdot P_{-1}^{w^{(t)}}\right]$$

Where we assume

$$\int_{u_l^{(t)}}^{u_h^{(t)}} \frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}} \exp\left\{-\frac{(\hat{r}-\mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2}\right\} d\hat{r} = P_{+1}^{w^{(t)}} \text{ and } \int_{u_l^{(t)}}^{u_h^{(t)}} \frac{1}{\sqrt{2\pi\sigma_{\hat{r}}^2}} \exp\left\{-\frac{(\hat{r}+\mu_{\hat{r}})^2}{2\sigma_{\hat{r}}^2}\right\} d\hat{r} = P_{-1}^{w^{(t)}}$$

We also assume that the destination will receive the value of $\int_{u_l^{(t)}}^{u_h^{(t)}} (\cdot)\, d\hat{r}$ from the relay by a side information channel.

5.4.2 QPSK Modulation at the Relay

Note that in Figure 5-2, $w^{(t)} = \{w^{(t,0)}, w^{(t,1)}\}$ has two bits. Thus, for BPSK, it needs two symbols to transmit. For QPSK, we only need one symbol. We can let $w^{(t)}$ be a complex number, i.e., $w^{(t)} = \{w^{(t,0)} + w^{(t,1)} j\}$. Now, we can re-plot the Figure 5-2 to 5-4.

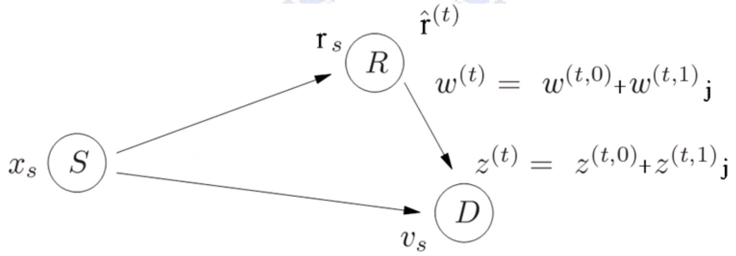


Figure 5-4: CF for QPSK modulation

The LLR can then be expressed as:

$$\begin{aligned} LLR(x^{(t)}) &= \log\frac{p(x^{(t)}=1\,|\,d^{(t)})}{p(x^{(t)}=-1\,|\,d^{(t)})} \\ &= \log\frac{p(v^{(t)}\,|\,x^{(t)}=1)}{p(v^{(t)}\,|\,x^{(t)}=-1)} + \log\frac{\sum_{w^{(t)}} p(z_I{}^{(t)}\,|\,w^{(t,0)})\,p(z_Q{}^{(t)}\,|\,w^{(t,1)}) \int_{u_l^{(t)}}^{u_h^{(t)}} p_{+1}^{(t)}(\hat{r})d\hat{r}}{\sum_{w^{(t)}} p(z_I{}^{(t)}\,|\,w^{(t,0)})\,p(z_Q{}^{(t)}\,|\,w^{(t,1)}) \int_{u_l^{(t)}}^{u_h^{(t)}} p_{-1}^{(t)}(\hat{r})d\hat{r}} \\ &= \frac{2\,|\,h_{SD}\,|^2}{\sigma_{SD}^2} v'^{(t)} \\ &\quad + \log\sum_{w^{(t)}}\left[\exp\left\{-\frac{|\,h_{RD}\,|^2\,((z'_I{}^{(t)} - w^{(t,0)})^2 + (z'_Q{}^{(t)} - w^{(t,1)})^2)}{2\sigma_{RD}^2}\right\}\cdot P_{+1}^{w^{(t)}}\right] \\ &\quad - \log\sum_{w^{(t)}}\left[\exp\left\{-\frac{|\,h_{RD}\,|^2\,((z'_I{}^{(t)} - w^{(t,0)})^2 + (z'_Q{}^{(t)} - w^{(t,1)})^2)}{2\sigma_{RD}^2}\right\}\cdot P_{-1}^{w^{(t)}}\right] \end{aligned} \tag{5.23}$$

Where $z_I^{'(t)}$ and $z_Q^{'(t)}$ is the real part and the image part of $z^{'(t)}$, and $z^{'(t)}$ is the same definition as in Section 5.4.1. For each bit of the QPSK signal, we can find that the region for $w^{(t,0)}$ and $w^{(t,1)}$ to demap. With the Gray coding, the regions are shown in Figure 5-5. Bit 1 is from $w^{(t,0)}$, and bit 2 is from $w^{(t,1)}$.

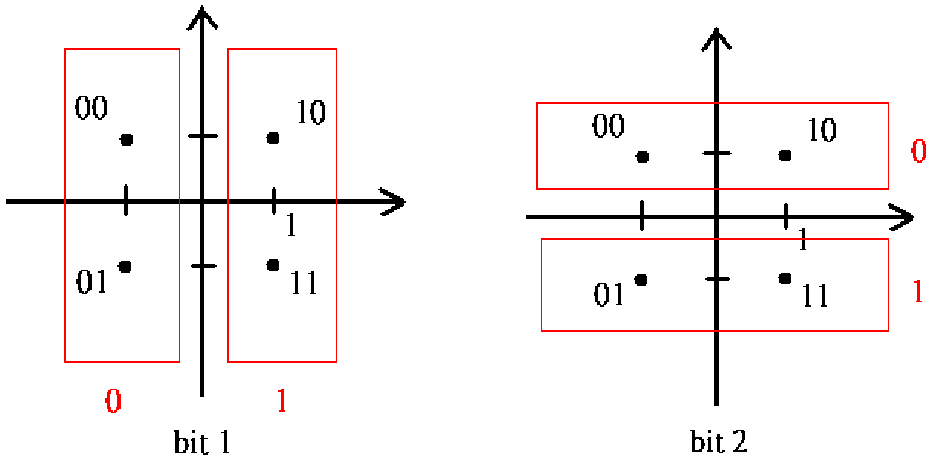


Figure 5-5 : The region for $w^{(t,0)}$ and $w^{(t,1)}$

5.4.3 16QAM Modulation at the Relay

The idea is similar to QPSK, and the system model is also similar to Figure 5-4. However, there are 4 bits carried in a 16QAM signal, so we need to modulate two bin-indices $\{w_n^{(t,0)}, w_n^{(t,1)}\}$ and $\{w_{n+1}^{(t,0)}, w_{n+1}^{(t,1)}\}$ to a 16QAM symbol. Therefore, the symbol we send from the relay is $w^{16QAM} = (w_n^{(t,0)}, w_n^{(t,1)}, w_{n+1}^{(t,0)}, w_{n+1}^{(t,1)}) = (w_n^{(t)} + w_{n+1}^{(t)} j)$, where $w_n^{(t)}, w_{n+1}^{(t)} = \{\pm 1, \pm 3\}$. Figure 5-6 is the corresponding system model.

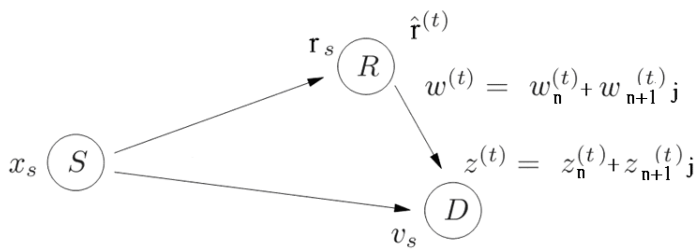


Figure 5-6 : CF for 16QAM modulation

The two LLR can be expressed as:

$$
\begin{aligned}
LLR(x_n^{(t)}) &= \log \frac{p(x_n^{(t)}=1\,|\,d^{(t)})}{p(x_n^{(t)}=-1\,|\,d^{(t)})} \\
&= \log \frac{p(v_n^{(t)}\,|\,x_n^{(t)}=1)}{p(v_n^{(t)}\,|\,x_n^{(t)}=-1)} + \log \frac{\sum_{w_n^{(t)}} p(z_I{}^{(t)}\,|\,w_n^{(t,0)})\,p(z_I{}^{(t)}\,|\,w_n^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} p_{+1}^{(t)}(\hat{r})d\hat{r}}{\sum_{w_n^{(t)}} p(z_I{}^{(t)}\,|\,w_n^{(t,0)})\,p(z_I{}^{(t)}\,|\,w_n^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} p_{-1}^{(t)}(\hat{r})d\hat{r}} \\
&= \log \frac{p(v_n^{(t)}\,|\,x_n^{(t)}=1)}{p(v_n^{(t)}\,|\,x_n^{(t)}=-1)} + \log \frac{\sum_{w_n^{(t)}} p(z_I{}^{(t)}\,|\,w_n^{(t)})\cdot P_{+1}^{w_n{}^{(t)}}}{\sum_{w_n^{(t)}} p(z_I{}^{(t)}\,|\,w_n^{(t)})\cdot P_{-1}^{w_n{}^{(t)}}} \\
&= \frac{2\,|h_{SD}|^2}{\sigma_{SD}^2} v_n'^{(t)} + \log \sum_{w_n{}^{(t)}} \left[\exp\left\{ -\frac{|h_{RD}|^2 (z'_I{}^{(t)} - w_n{}^{(t)})^2}{2\sigma_{RD}^2} \right\} \cdot P_{+1}^{w_n{}^{(t)}} \right] \\
&\quad - \log \sum_{w_n{}^{(t)}} \left[\exp\left\{ -\frac{|h_{RD}|^2 (z'_I{}^{(t)} - w_n{}^{(t)})^2}{2\sigma_{RD}^2} \right\} \cdot P_{-1}^{w_n{}^{(t)}} \right]
\end{aligned} \tag{5.24}
$$

$$
\begin{aligned}
LLR(x_{n+1}^{(t)}) &= \log \frac{p(x_{n+1}^{(t)}=1\,|\,d^{(t)})}{p(x_{n+1}^{(t)}=-1\,|\,d^{(t)})} \\
&= \log \frac{p(v_{n+1}^{(t)}\,|\,x_{n+1}^{(t)}=1)}{p(v_{n+1}^{(t)}\,|\,x_{n+1}^{(t)}=-1)} + \log \frac{\sum_{w_{n+1}^{(t)}} p(z_Q{}^{(t)}\,|\,w_{n+1}^{(t,0)})\,p(z_Q{}^{(t)}\,|\,w_{n+1}^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} p_{+1}^{(t)}(\hat{r})d\hat{r}}{\sum_{w_{n+1}^{(t)}} p(z_Q{}^{(t)}\,|\,w_{n+1}^{(t,0)})\,p(z_Q{}^{(t)}\,|\,w_{n+1}^{(t,1)})\int_{u_l^{(t)}}^{u_h^{(t)}} p_{-1}^{(t)}(\hat{r})d\hat{r}} \\
&= \log \frac{p(v_{n+1}^{(t)}\,|\,x_{n+1}^{(t)}=1)}{p(v_{n+1}^{(t)}\,|\,x_{n+1}^{(t)}=-1)} + \log \frac{\sum_{w_{n+1}^{(t)}} p(z_Q{}^{(t)}\,|\,w_{n+1}^{(t)})\cdot P_{+1}^{w_{n+1}{}^{(t)}}}{\sum_{w_{n+1}^{(t)}} p(z_Q{}^{(t)}\,|\,w_{n+1}^{(t)})\cdot P_{-1}^{w_{n+1}{}^{(t)}}} \\
&= \frac{2\,|h_{SD}|^2}{\sigma_{SD}^2} v_{n+1}'^{(t)} + \log \sum_{w_{n+1}{}^{(t)}} \left[\exp\left\{ -\frac{|h_{RD}|^2 (z'_Q{}^{(t)} - w_{n+1}{}^{(t)})^2}{2\sigma_{RD}^2} \right\} \cdot P_{+1}^{w_{n+1}{}^{(t)}} \right] \\
&\quad - \log \sum_{w_{n+1}{}^{(t)}} \left[\exp\left\{ -\frac{|h_{RD}|^2 (z'_Q{}^{(t)} - w_{n+1}{}^{(t)})^2}{2\sigma_{RD}^2} \right\} \cdot P_{-1}^{w_{n+1}{}^{(t)}} \right]
\end{aligned} \tag{5.25}
$$

Where $z_I^{'(t)}$ and $z_Q^{'(t)}$ is the real part and the image part of $z^{'(t)}$. For the each bit of the 16QAM symbol, we can find that the region for $w_n^{(t)}$ and $w_{n+1}^{(t)}$ to demap. With Gray coding, the regions are shown in Figure 5-7. Bit 1 and bit 2 are from $w_n^{(t)}$, and bit 3 and bit 4 are from $w_{n+1}^{(t)}$.

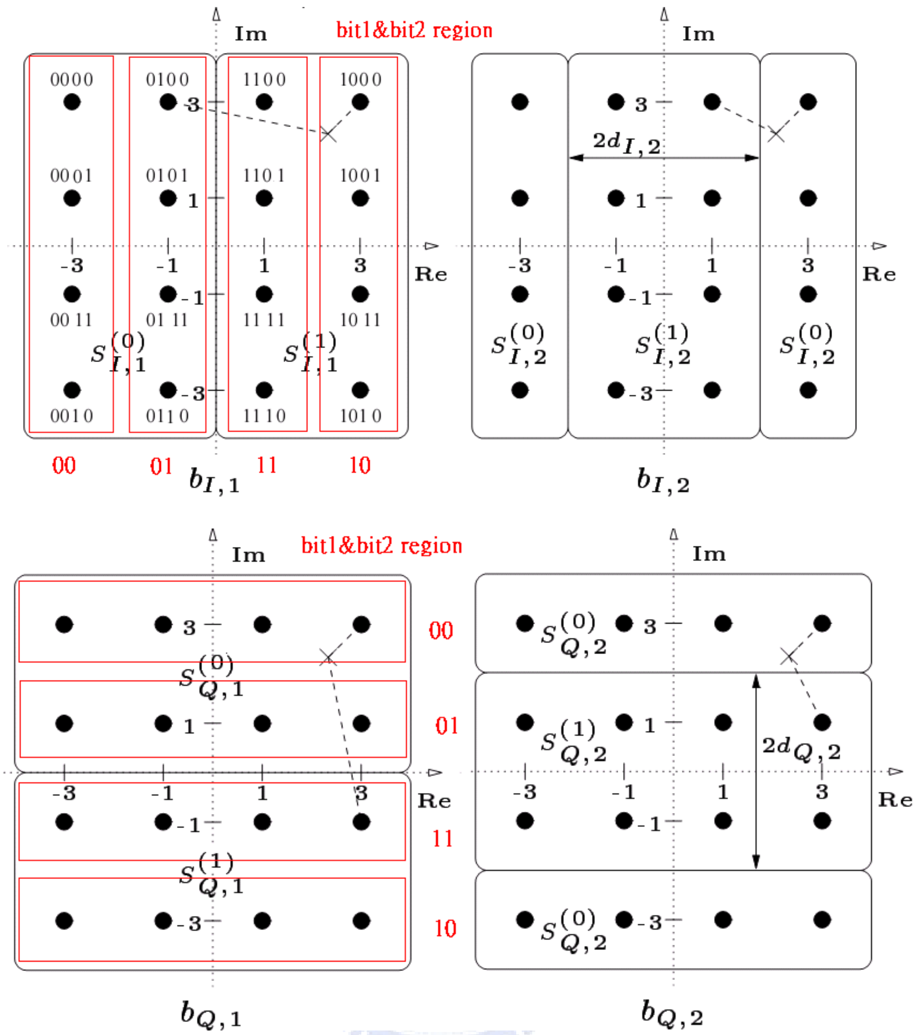


Figure 5-7 : The region for $w_n^{(t)}$ and $w_{n+1}^{(t)}$

Similarly, we assume that the destination will receive the value of $\int_{u_l^{(t)}}^{u_h^{(t)}} (\cdot)\, d\hat{r}$ from the relay by a side information channel. After computing the LLR, we can use it as the input to the LDPC decoder to find the soft decoder LLR. Finally, we make data decisions as that in (2.30).

6 Simulations

In this chapter, we will report simulate results evaluating the performance of different cooperative schemes in different scenarios. In the simulations, we assume that the instantaneous CSI h_{SD}, h_{SR}, and h_{RD} are known.to the receivers, and BPSK, QPSK, and 16QAM are used as the modulation schemes. The bit error rate (BER) and packet error rate (PER) are used as the performance measures.

We also assume that h_{SD}, h_{SR}, and h_{RD} are spatially independent and experience slow Rayleigh fading. The variance of each channel is one. We also consider the line-of-side (LOS) channel in which each channel has an unit gain. As to the noise, we consider the AWGN. The means of n_{SD}, n_{SR}, and n_{RD} are zeros and the variances are σ_{SD}^2, σ_{SR}^2, and σ_{RD}^2, respectively. Given the SNR and the average power of a signal P_k, we can compute the noise variance easily. For reference simplicity, we let the SNR of the SR channel be denoted as SNR_{SR}' that of the SD as SNR_{SD}, and that of RD as SNR_{RD}.

At the source, we encode the original information bits with the LDPC encoder defined in IEEE 802.15.3c with code rate=1/2. We let the packet size be equal to the coding-block size. In other words, there is one LDPC codeword (672 bits) in one packet. Five scenarios are considered. We use the DC scheme for Scenario 1 to scenario 4, and the MD scheme for Scenario 5.

6.1 Scenario 1

In this scenario, we consider the LOS channel, i.e, the gain of each channel is always one. We evaluate the performance of the non-cooperative (NC), the cooperative, and the cooperative LDPC-coded schemes. Here let SNR_{SD}=SNR_{SR}=SNR_{RD}. Figure 6-1 shows the simulation results. As we can see, at BER=10^{-3} the cooperative scheme outperforms the NC about 2 dB. Also, AF with LDPC coding outperforms AF without coding about 4 dB.

We then conduct more simulations for AF without LDPC coding. Let SNR_{SD}=5dB, SNR_{RD}=1, 5, 9 dB, and SNR_{SR} be varied. Figure 6-2 shows the performance comparison. From the figure, we see that the higher the SNR_{SR}, the better the performance we can have. Then, we let SNR_{SD} =5dB, SNR_{SR}=5, 9dB, and SNR_{RD} be varied. Figure 6-3 shows the performance comparison. From the figure, we see that the higher channel SNR_{RD}, the better the performance we can have.

Then we conduct simulations for AF with LDPC coding. Let SNR_{SD}=1dB, SNR_{RD}=1, 5 dB, and SNR_{SR} be varied. Figure 6-4 shows the performance comparison. From the figure, we see that the higher the SNR_{SR}, the better the performance we can have. Because of with LDPC coding, the performance is much better than the situation without LDPC coding. Then, we let SNR_{SD} =1dB, SNR_{SR}=1, 5dB, and SNR_{RD} be varied. Figure 6-5 shows the performance comparison. From the figure, we also see that the higher channel SNR_{RD}, the better the performance we can have. Sum up, cooperative systems with LDPC codes can work better than without LDPC codes.

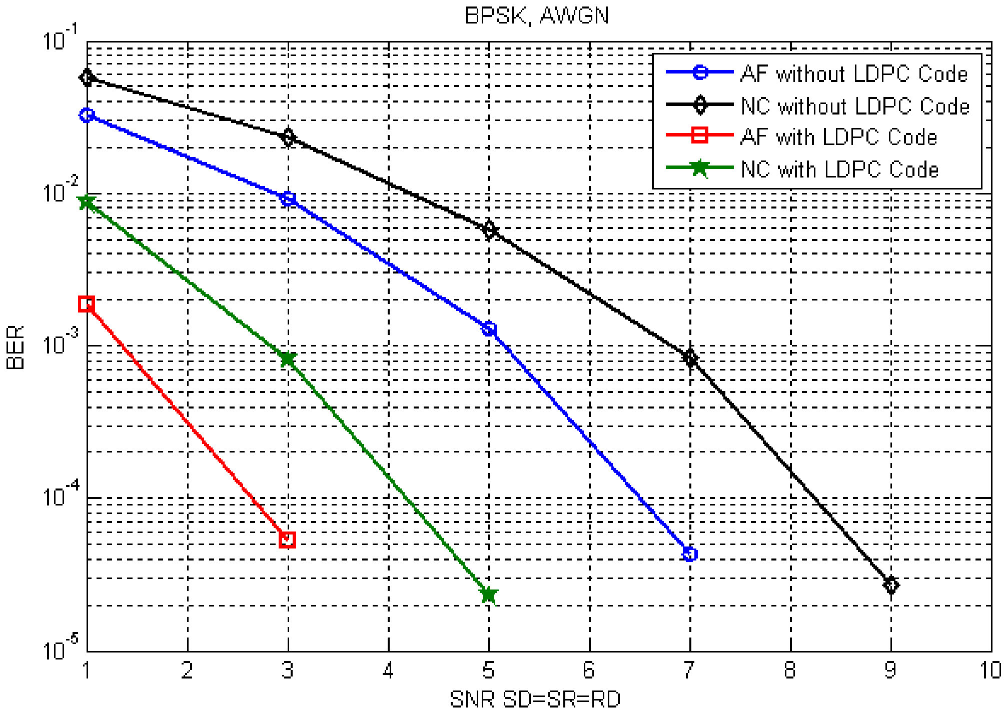


Figure 6-1 : BER comparison for AF cooperative/non cooperative systems with LDPC codes and without LDPC codes

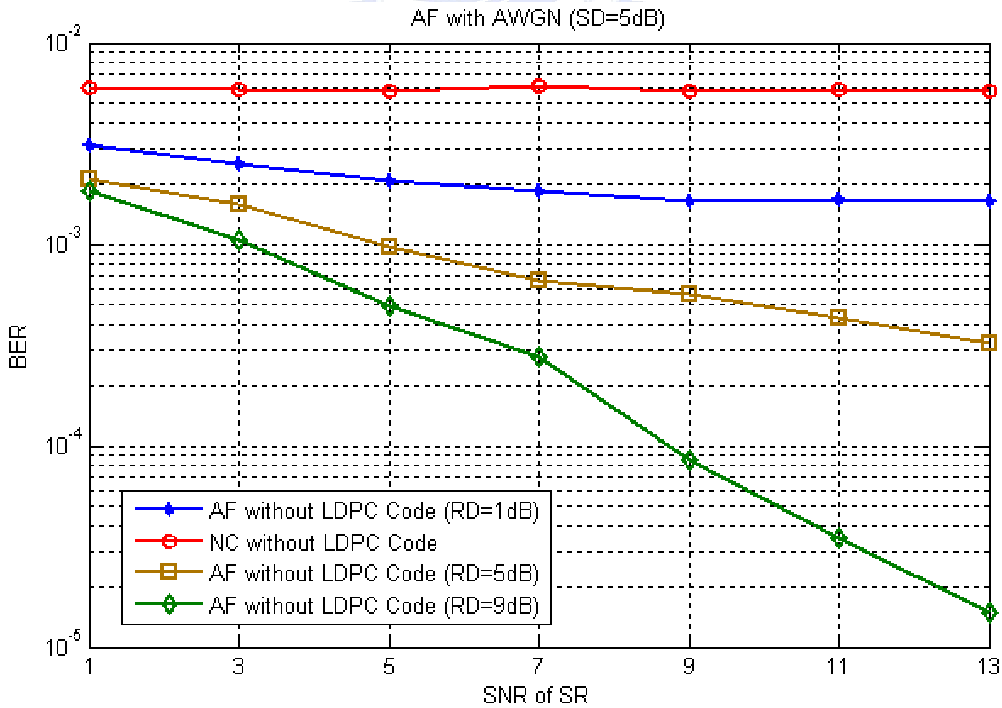


Figure 6-2 : BER comparison for various SNR_{SR} in AF without LDPC codes

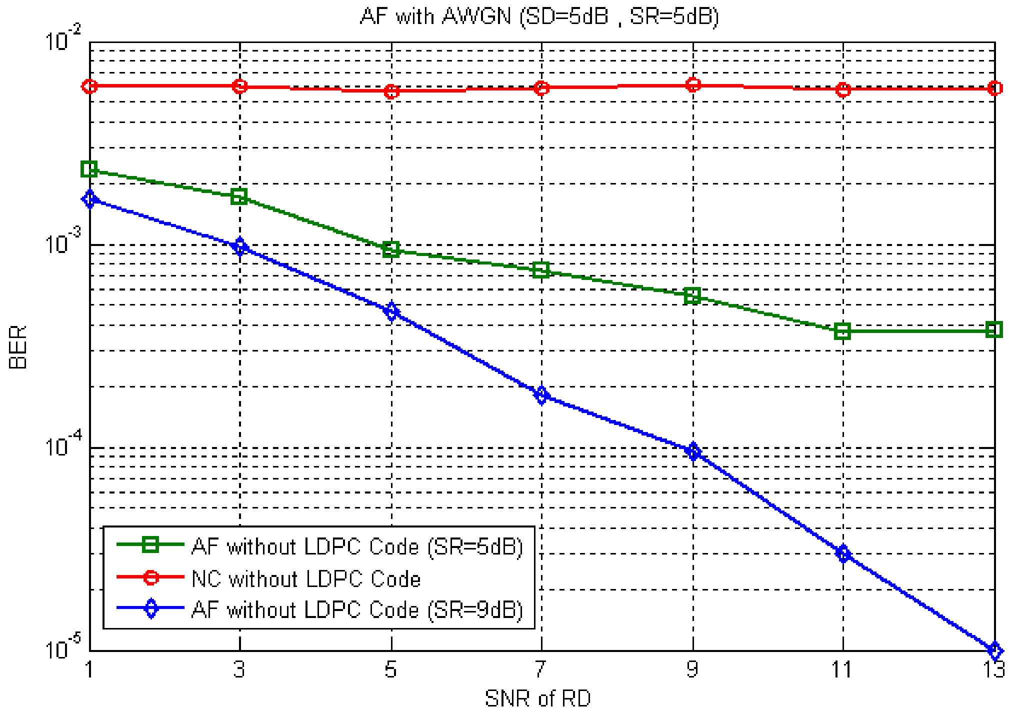


Figure 6-3 : BER comparison for various SNR_{RD} in AF without LDPC codes

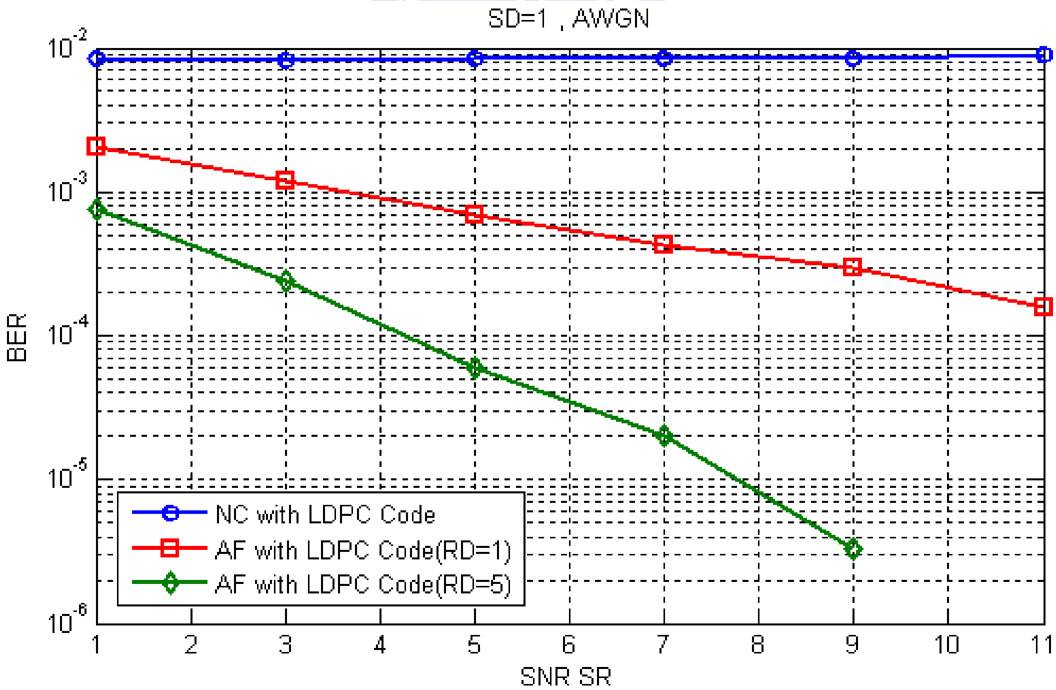


Figure 6-4 : BER comparison for various SNR_{SR} in AF with LDPC codes

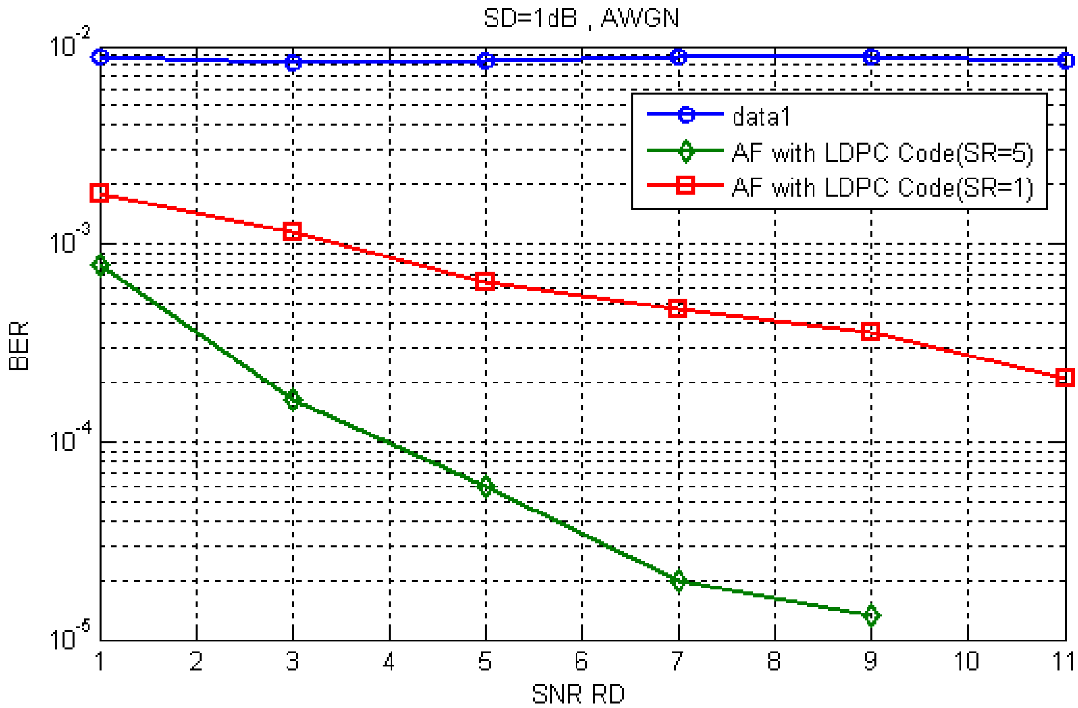


Figure 6-5 : BER comparison for various SNR_{RD} in AF with LDPC codes

6.2 Scenario 2

In this scenario, we consider the system with Rayleigh fading channels. We compare the PER performance between the NC, AF, and DF systems with BPSK modulation. The channel SNRs are set as SNR_{SD}=SNR_{SR}=SNR_{RD}. In AF, the relay just amplifies the signals and transmits to the destination, so the relay will propagate the noise. However, it does not have the decision errors. In DF, it is degenerated to the NC mode when decision error occurs at relay. Despite of that, DF has 1~2.5dB gain over AF.

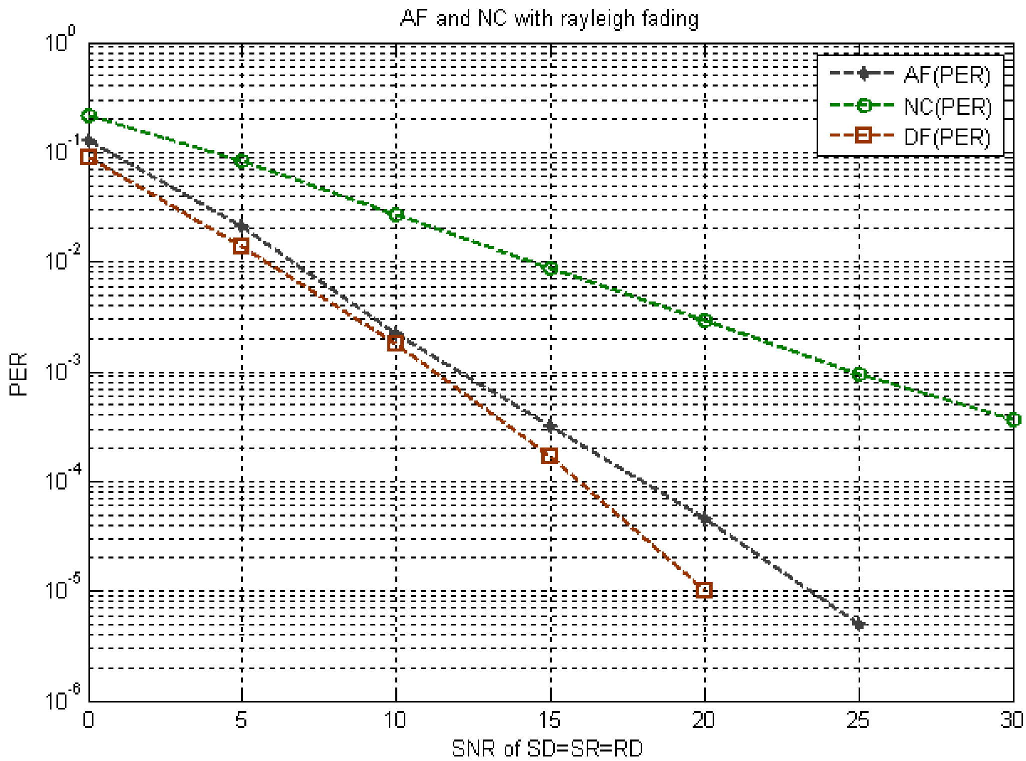


Figure 6-6 : BER comparison for NC, AF and DF,

(SNR_{SR}= SNR_{SD}= SNR_{RD})

6.3 Scenario 3

In this scenario, we assume the LOS channel in the system. That means in every packet, the SNR is always fixed. We include the CF scheme in our simulations. In CF, if the relay decodes the information bits correctly, it will choose the DF mode to re-encode and re-transmits the information bits to the destination. If it decodes the bits incorrectly, the relay will have two modes to choose, the CF or NC modes. Here, we set a threshold for the mode selection. If the BER is higher than the threshold at the relay, the relay will choose the CF mode; otherwise, the relay will switch to NC mode. The threshold we set is 0.5.

For cooperative systems, the source uses the 16QAM modulation scheme. At the first time slot, it transmits the modulated signal to the relay and the destination. At the second slot, the relay uses DF or CF to transmit the processed signal to the destination. In DF, the 16QAM scheme is used, while for CF, BPSK, QPSK, and 16QAM modulation schemes are used. We use CF (BPSK), CF (QPSK) , and CF(16QAM) to denote the various CF schemes we consider. Note that the data rates in the RD channel are different for different modulation/cooperative schemes. In general, the CF scheme requires a higher data rate. However, as the typical case, the PER is small, the overhead introduced by the CF scheme will be slightly higher than the DF scheme.

6.3.1 Case 1

We set the channel SNRs as $SNR_{SR}=SNR_{RD} - 8$ and $SNR_{SD}=SNR_{RD} - 10$. Figure 6-7 shows the simulation results and we can find that the performance of CF is much better than DF. Also, the performance of CF(BPSK), CF(QPSK), and CF(16QAM) is very close. Below SNR_{RD}=15 dB, the performance of DF is almost the same as NC. This is because SNR_{SR} is low and the DF always switches to the NC mode most of the time.

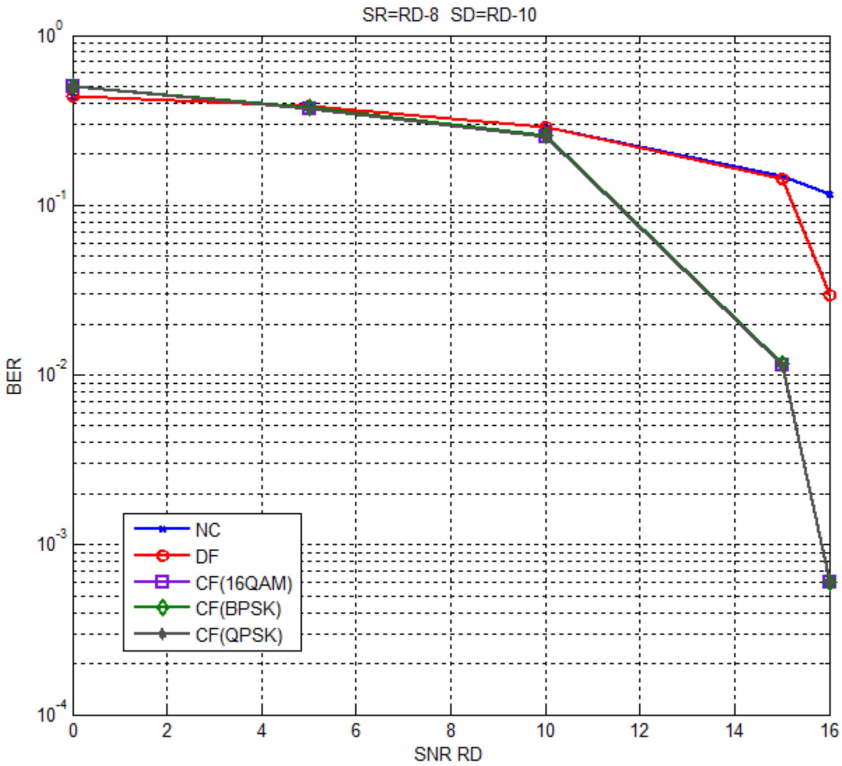


Figure 6-7 : BER comparison for NC, DF, and CF in LOS channel,

(SNR_{SR}=SNR_{RD} - 8 and SNR_{SD}=SNR_{RD} – 10)

6.3.2 Case 2

We set the channel SNRs as SNR_{SR}=7dB and SNR_{SD}=5dB, and vary SNR_{RD}. Figure 6-8 shows the results and we can find that in this case the DF is still close to NC. The performance of the CF scheme improves very quickly as SNR_{RD} is getting higher. Finally, the BER will saturate around 10^{-2}.

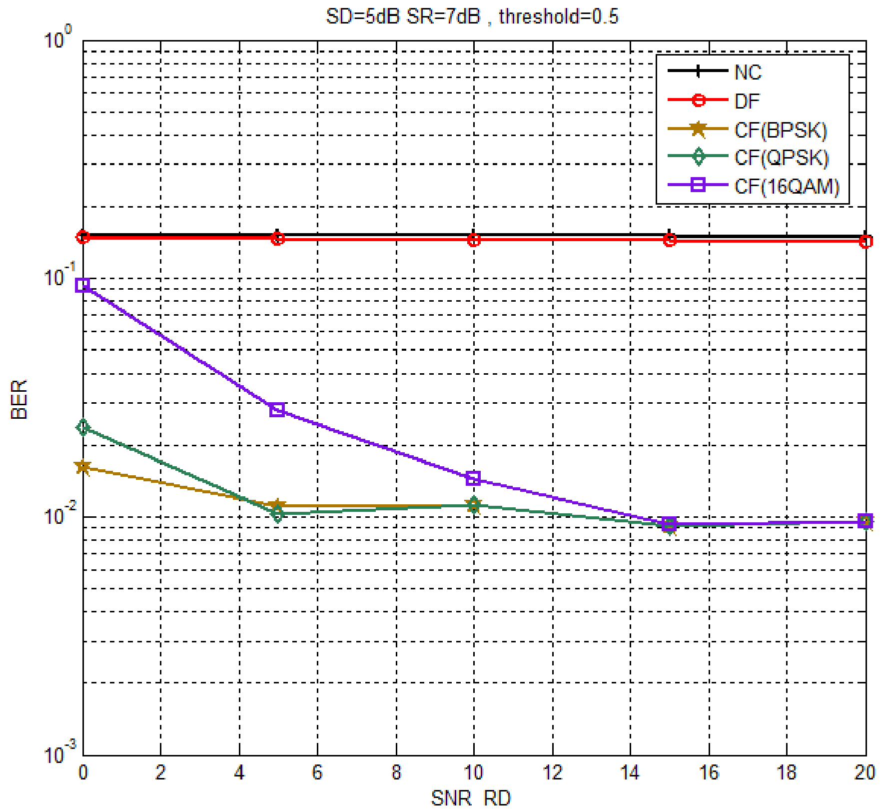


Figure 6-8 : BER comparison for NC, DF, and CF in LOS channel

(SNR_{SR}=7dB and SNR_{SD}=5dB)

6.3.3 Case 3

We set the channel SNRs as SNR_{SR}=8dB and SNR_{SD} = SNR_{RD} - 10dB. Figure 6-9 shows the simulation results and we can find that the performance of CF is much better than that of DF while SNR_{RD} is higher than 5dB. The higher the SNR_{RD}, the larger gain we can obtain with CF.

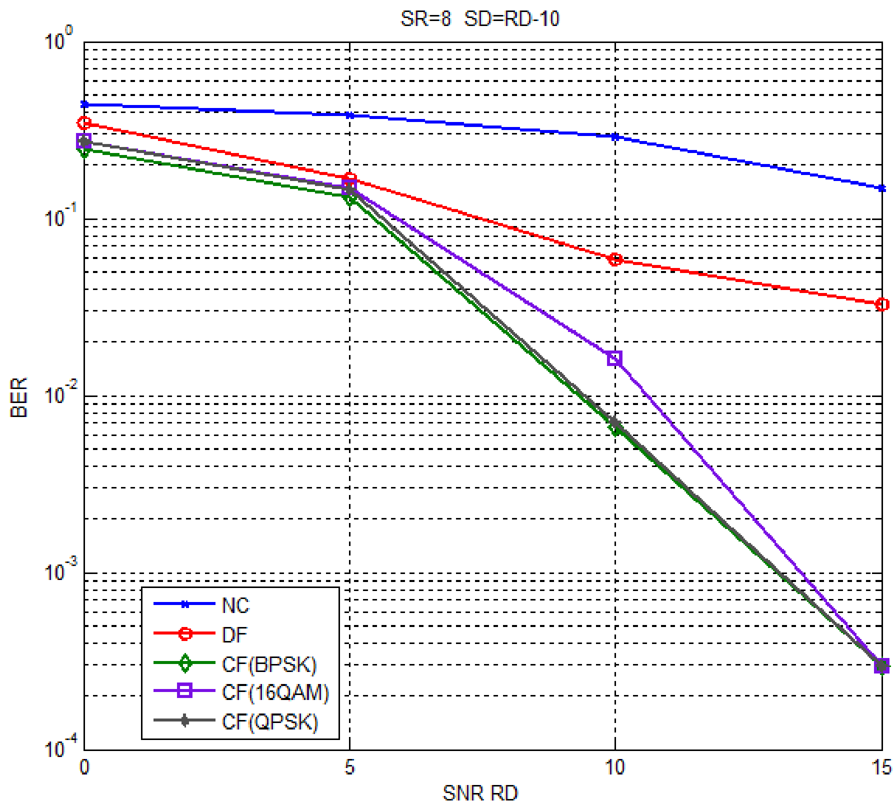


Figure 6-9 : BER comparison for NC, DF, and CF in LOS channel

(SNR_{SR}=8dB and SNR_{SD} = SNR_{RD} - 10dB)

6.3.4 Case 4

We set the channel SNRs as SNR_{SR}=SNR_{RD}, and SNR_{SD}=SNR_{RD}-10dB. Figure 6-10 shows the result. From the figure, we can see that the performance of NC and DF curve is almost the same when SNR_{RD} is less than 7dB. Also, and the DF is worse than AF in this SNR region. The reason, as mentioned, DF switches to the NC mode most of the time. When SNR_{RD} is higher than 7dB, the performance of DF starts to improve and becomes better than that of CF. As we can see, CF always gives the best performance among all the cooperative schemes.

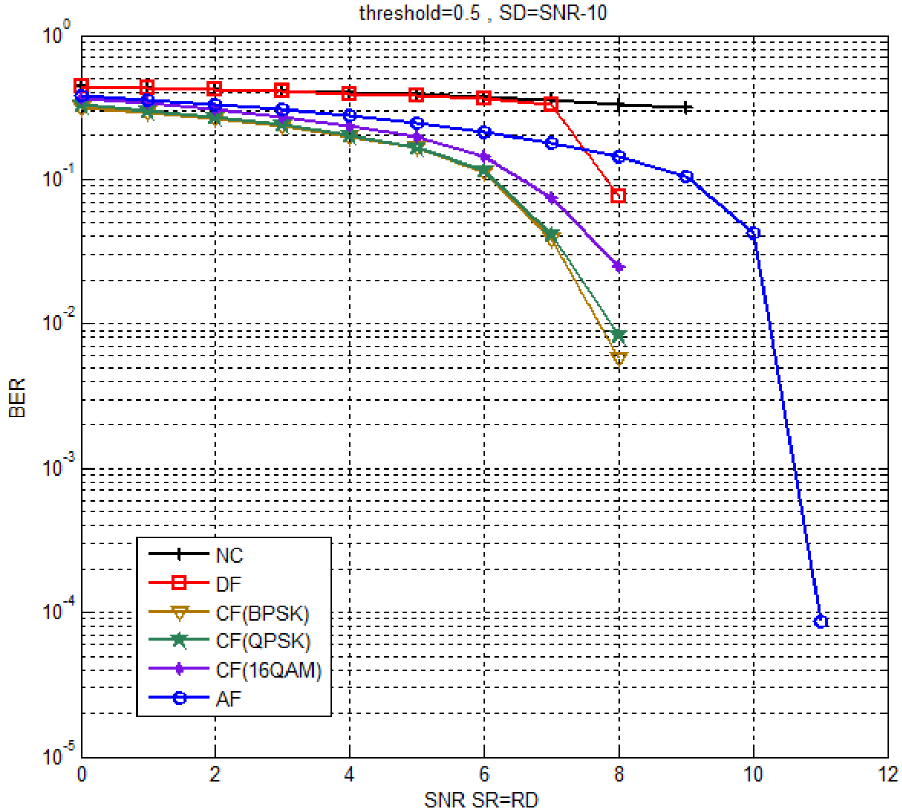


Figure 6-10 : BER comparison for NC, DF, and CF in LOS channel

($SNR_{SR}=SNR_{RD}$, and $SNR_{SD}=SNR_{RD}$-10dB)

6.4 Scenario 4

In this scenario, we assume Rayleigh fading channels in our system. At the source, the transmitter uses QPSK as the modulation scheme. At the relay, DF uses QPSK as the modulation scheme, while CF uses BPSK, QPSK, or 16QAM. Since a two-bit quantizer is used in CF, the transmit bits at the relay is doubled. We use DC at the destination. As a result, if BPSK or QPSK is used the data rate of CF is higher than that of DF. However, if 16QAM is used, the data rate for CF is then the same as that of DF.

6.4.1 Case 1

We set SNR_{SR}=SNR_{RD} and SNR_{SD}=SNR_{RD}-10dB. Figure 6-11 shows the simulation result. From the figure, we can find that the performance of CF is slightly better than that of DF. This indicates that the relay either uses DF or NC most of the time. Also, the performance of the AF scheme is about 2dB worse than the DF and CF schemes.

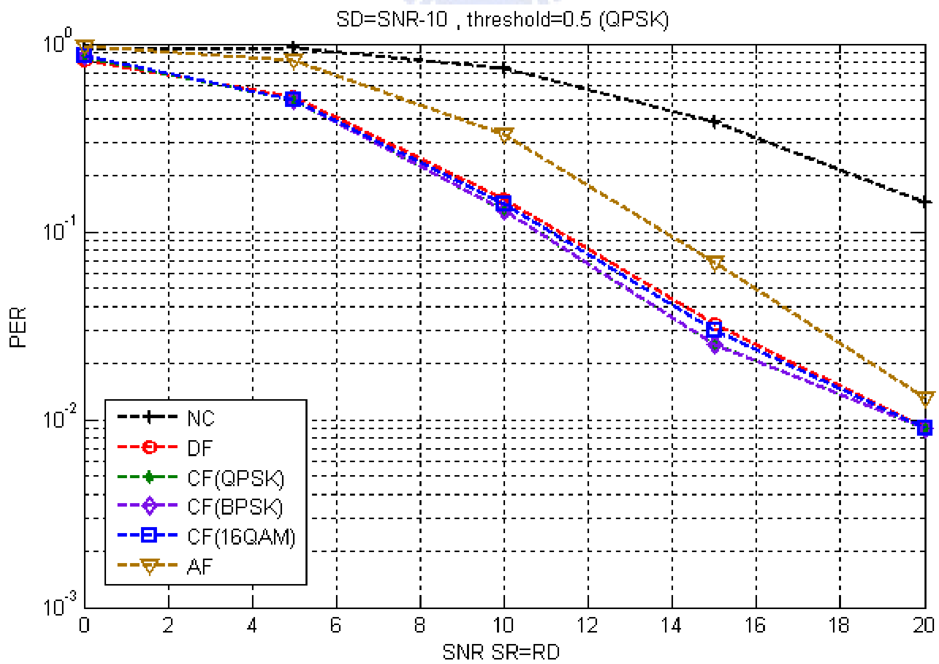


Figure 6-11 : PER comparison for NC, DF, and CF in Rayleigh channel

(SNR_{SR}=SNR_{RD} and SNR_{SD}=SNR_{RD}-10dB)

6.4.2 Case 2

We consider two scenarios that SNR_{SR}=7dB and SNR_{SD}=SNR_{RD} - 10dB and SNR_{SR}=15dB and SNR_{SD}=SNR_{RD} - 10dB. The results are shown in Figures 6-12 and 6-13. From the figures, we see that the performance trend is the same. When SNR_{SR} is higher, CF and DF can have more gains over NC and AF. Also, CF outperform DF by 1.5~2dB when SNR_{RD} is 15dB. This indicates that the BER at the relay is main factor influencing the CF performance. When the BER is low at the relay, the CF can then forward useful LLR information to the destination. When the BER is high, the relay will either switch to the NC model or forward insignificant LLR information.

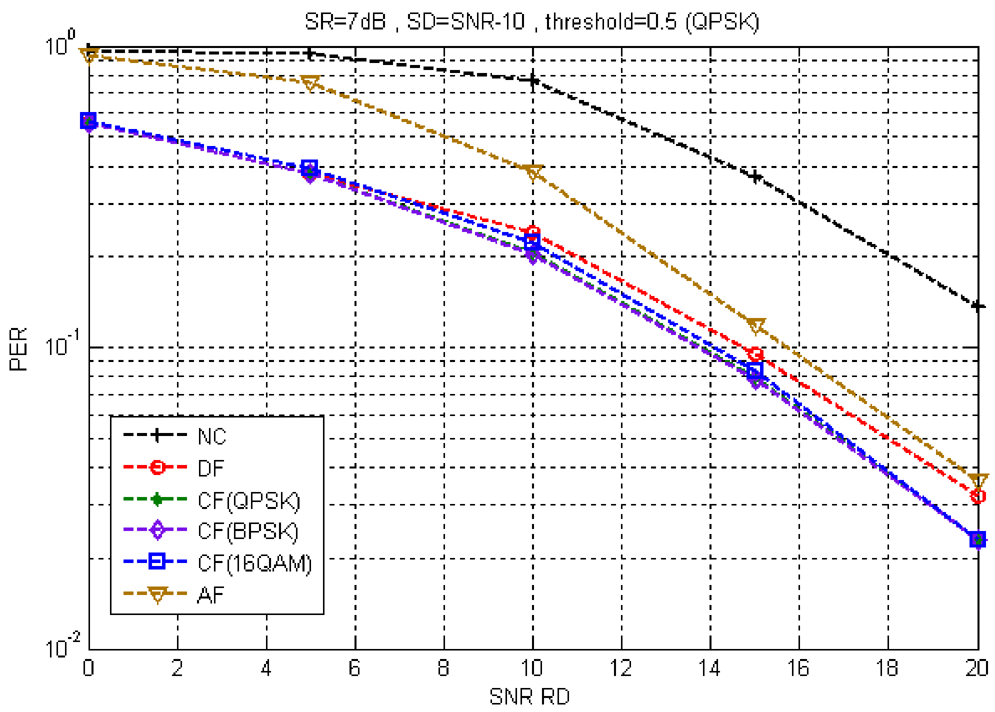


Figure 6-12 : PER comparison for NC, DF, and CF in Rayleigh channel

(SNR_{SR}=7dB and SNR_{SD}=SNR_{RD} - 10dB)

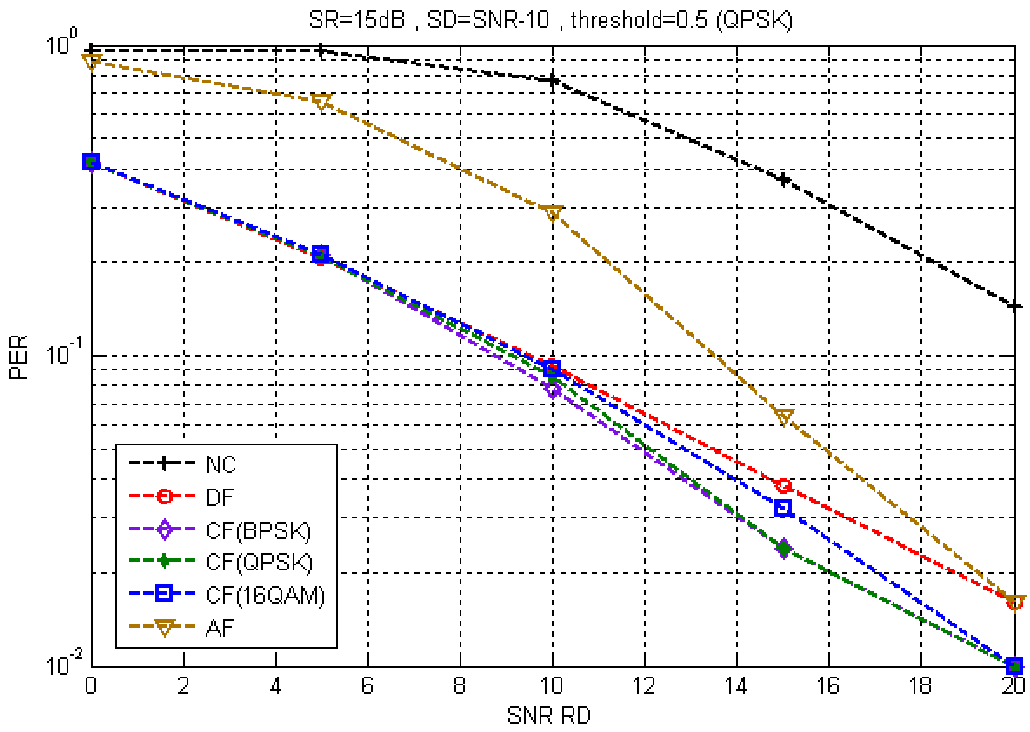


Figure 6-13 : PER comparison for NC, DF, and CF in Rayleigh channel

(SNR_{SR}=7dB and $SNR_{SD}=SNR_{RD}$ - 10dB)

6.4.3 Case 3

We set the channel SNRs as $SNR_{SR}=SNR_{SD}=SNR_{RD}$. Figure 6-14 shows the result. From the figure, we see that AF has more than 2dB gains over NC, but performs worse than DF and CF. The performance of CF(16QAM) is slightly better than DF. As mentioned, since the PER is usually small at the relay, we can actually use CF(QPSK) or CF(BPSK) instead of CF(16QAM) at the relay. In case, we can obtain 0.5~1dB gain.

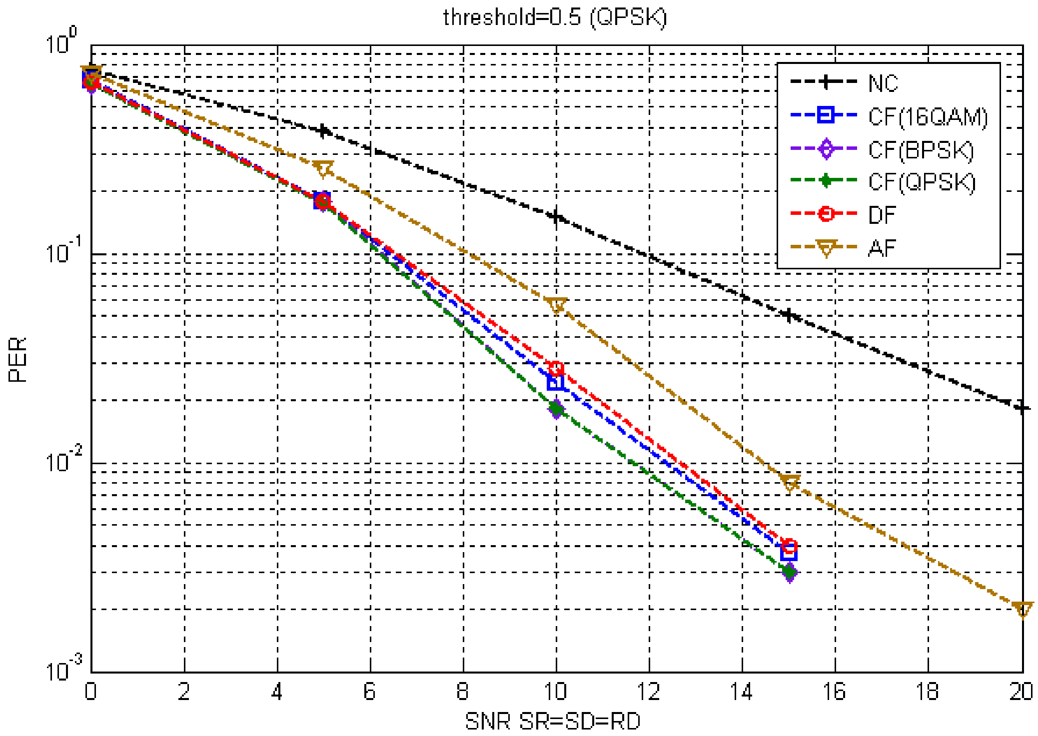


Figure 6-14 : PER comparison for NC, DF, and CF in Rayleigh channel

(SNR_{SR}=SNR_{SD}=SNR_{RD})

6.5 Scenario 5

In this scenario, we assume Rayleigh fading channels in our system. The source uses QPSK as the modulation scheme while the relay with DF uses QPSK and that with CF uses BPSK, QPSK, or 16QAM. Since a two-bit quantizer is used in CF, the transmit bits at the relay is doubled. Thus, if BPSK or QPSK is used, the data rate of CF is higher than that of DF. However, if 16QAM is used, the data rate for CF is the same as that of DF. At the destination, we use the MD scheme to recover the transmitted data.

6.5.1 Case 1

We consider two scenarios that SNR_{SR}=7dB and $SNR_{SD}=SNR_{RD}$ - 10dB, and SNR_{SR}=15dB and $SNR_{SD}=SNR_{RD}$ - 10dB. The results are shown in Figures 6-15 and 6-16. From the figures, we see that CF outperform DF by 0.5~1dB when PER is 10^{-1}.

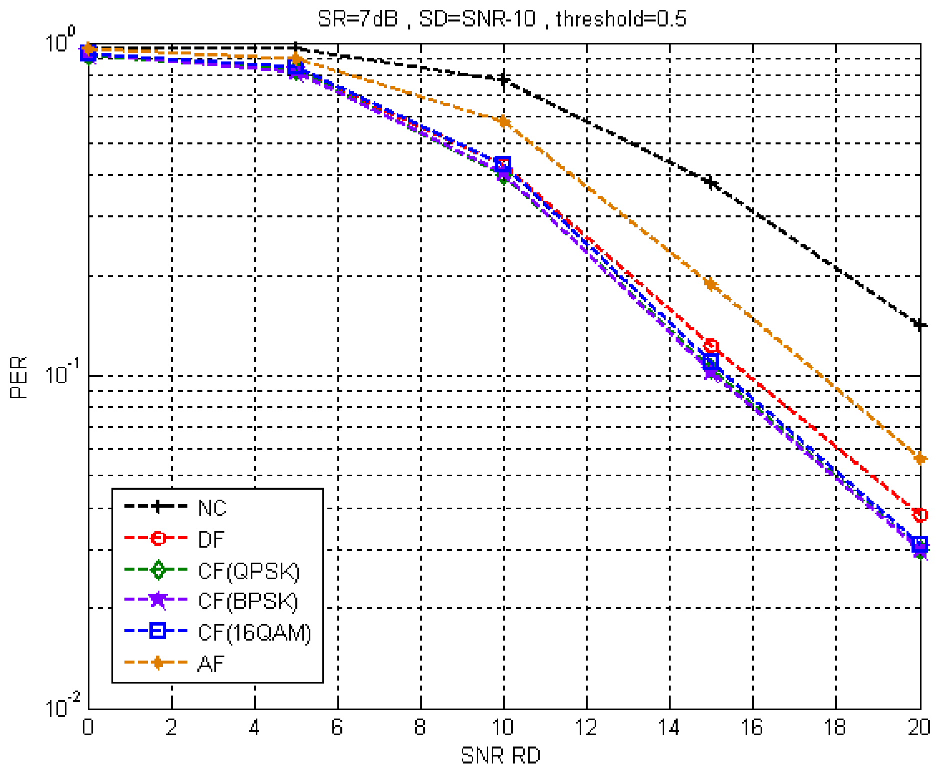


Figure 6-15 : PER comparison for NC, DF, and CF in Rayleigh channel

(SNR_{SR}=7dB and $SNR_{SD}=SNR_{RD}$ - 10dB)

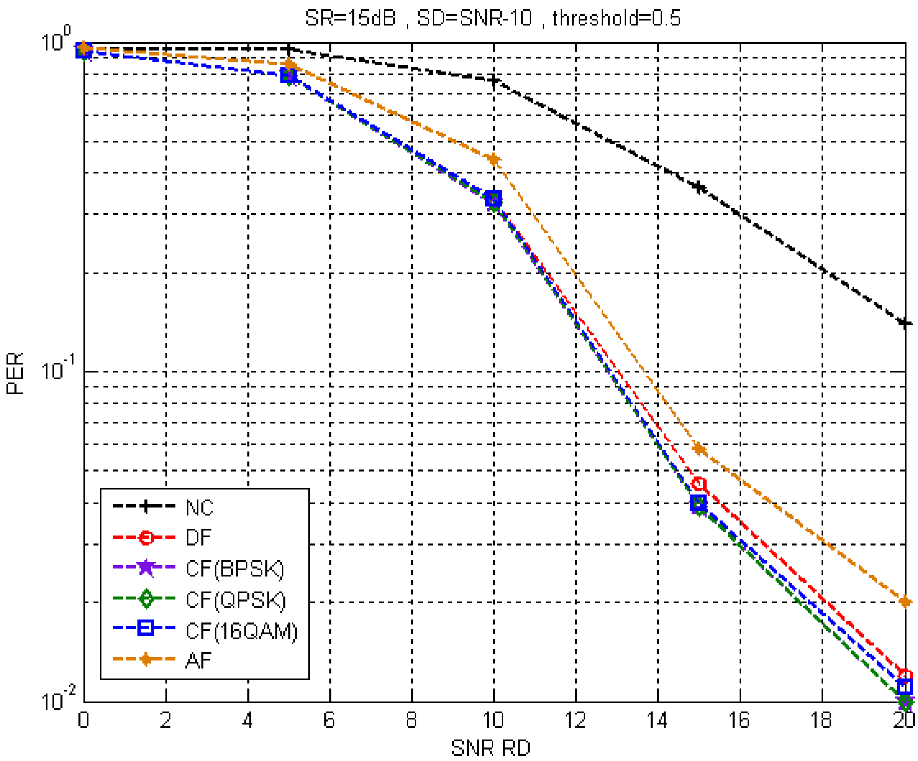


Figure 6-16 : PER comparison for NC, DF, and CF in Rayleigh channel

(SNR_{SR}=15dB and SNR_{SD}=SNR_{RD} - 10dB)

6.5.2 Case 2

We let the channel SNRs as $SNR_{SR}=SNR_{SD}=SNR_{RD}$. Figure 6-17 shows the result. From the figure, we see that AF can have more than 5dB gains over NC, but performs worse than DF and CF. The performance of CF(16QAM) is better than DF. In this case, we can obtain 0.5~1dB gain when CF(16QAM) is used.

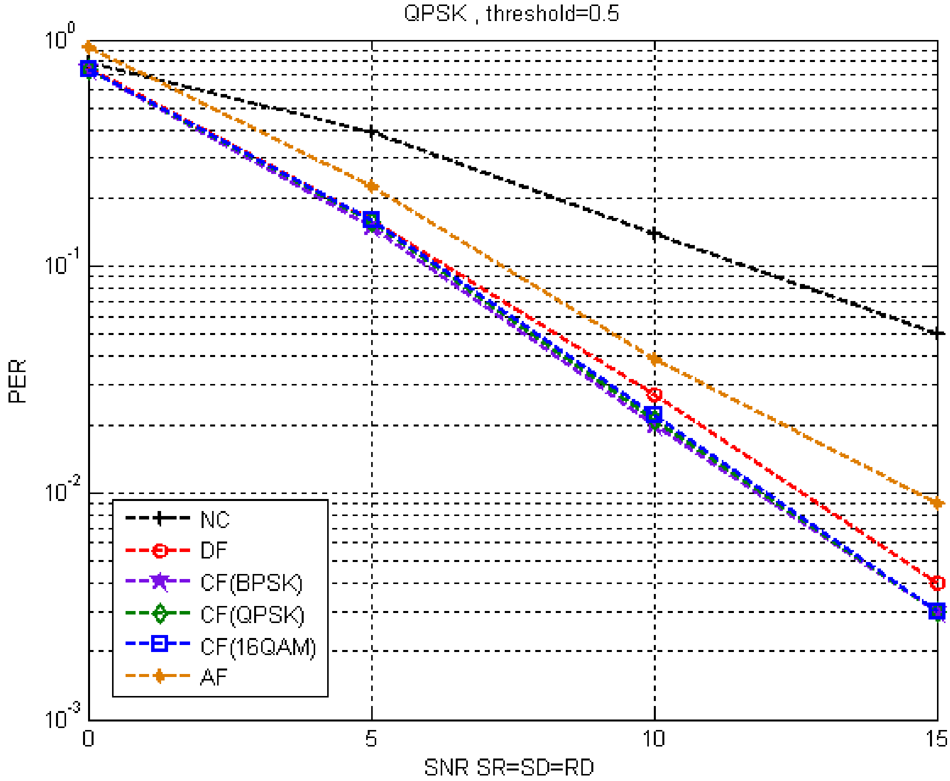


Figure 6-17 : PER comparison for NC, DF, and CF in Rayleigh channel

(SNR_{SR}=SNR_{SD}=SNR_{RD})

7 Conclusions

Diversity is known to be an effective technique to combat fading. With multiple transmit/receive antennas, spatial diversity, which does not scarify the spectrum efficiency, can be realized. Cooperative communication has been recently proposed to achieve virtual spatial diversity. AF and DF are two commonly used cooperative strategies. Although they are simple to apply, the performance may not be always satisfactory. In this thesis, we focus on the CF strategy. We extend the method in [11] and develop a practical CF scheme with the LDPC code. Specifically, we consider the QAM modulation scheme in the relay link. To estimate the LLRs of information bits, we model the distributions as Gaussian mixtures, and use the EM algorithm to identify the unknown parameters. Using the method, we can then have a higher spectral efficiency for the relay link. Simulation shows that the proposed CF scheme can outperform AF and DF.

In concluding the work, we outline some possible topics for further research. First, in this thesis, we assume that the BER at the relay is known and this may be not realistic. We may use the LLR distribution to obtain an estimate of the BER to solve the problem. Also, the LDPC coding conducted at the relay is a repetition of that at the source, and all the coded bits are transmitted. Instead, we may use a different LDPC code or just transmit some of the coded bits at the relay. This may also serve a potential topic for further research.

Reference

[1] T. M. Cover and A. A. El Gamal, “ Capacity Theorems for the Relay Channel, ” *IEEE Trans. Inform. Theory*, Vol. IT-25, pp. 572-584, Sept. 1979.

[2] Xingkai Bao and Jing Li, “ Efficient Message Relaying for Wireless User Cooperation: Decode-Ampligy-Forward (DAF) and Hybrid DAF and Coded-Cooperation, ” *IEEE Trans. Wireless Commun.*, Vol. 6, NO. 11, Nov. 2007.

[3] Robert G. Gallager, “ Low Density Parity Check Codes, ” *IRE Trans. Inform. Theory*, 1962.

[4] C.E. Shannon, “ A Mathematical Theory of Communication, ” *Bell System Technical Journal*, Vol. 27, pp. 379-423, 623-656, July, Oct. 1948

[5] R. M. Tanner, “ A Recursive Approach to Low Complexity Codes, ” *IEEE Trans. Inform. Theory*, Vol. IT-27, NO. 5, pp. 533–547, Sep. 1981.

[6] D. J. C. Mackay, “Good Error-Correcting Codes Based on Very Sparse Matrices,” *IEEE Trans. Inform. Theory*, 45 (2): 399-432, March 1999.

[7] *IEEE* Std 802.15.3-2003, Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPANs), Nov, 2008.

[8] Wang Ruihua, Yin Liuguo, Yu Quan, Lu Jianhua, “ Design of Irregular LDPC Code with Unequal Error Protection Property and its Performance Analysis on Image Transmission, ” *Journal of Electronics (China)*, Vol. 22, NO. 6, Nov. 2005.

[9] Filippo Tosato and Paola Bisaglia, “Simplified Soft Output Demapper for Binary Interleaved COFDM with Application to HIPERLAN/2, ” HP Laboratories Bristol, Oct. 2001.

[10] Jeff A. Bilmes, “ A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models, ” International Computer Science Institute Berkerley CA., April 1998.

[11] Ruiyuan Hu and Jing Li, “ Practical Compress-Forward in User Cooperation: Wyner-Ziv Cooperation, ” *ISIT* 2006, Seattle, USA, July 2006.

[12] Nazanin Rahnavard and Faramarz Fekri, “ Unequal Error Protection Using Low-Density Parity-Check Codes, ” *ISIT* 2004, Chicago, USA, 2004.

[13] Arnab Chakrabarti, Elza Erkip, Ashutosh Sabharwal, and Behnaam Aazhang, “ Code Designs for Cooperative Communication, ” IEEE Signal Processing Magazine, pp. 16-25, Sept. 2007.

[14] R. Hu and J. Li, “ Exploiting Slepian-Wolf Coding in Wireless User Cooperation, ” *IEEE SPAWC*, New York, NY, June 2005.

[15] Z. Tu, J. Li, and R. Blum, “ On Scalar Quantizer Design with Decoder Side Information, ” submitted to *IEEE Trans. Inform. Theory*, 2005.

[16] R. Hoshyar, R. Tafazolli, “Soft Decode and Forward of MQAM Modulations for Cooperative Relay Channels,” *IEEE VTC Spring 2008*, pp. 639 – 643, May 2008.