A Practical Coding Scheme For Broadcast Channel

A PRACTICAL CODING SCHEME FOR BROADCAST CHANNEL

BY

WENBO SUN, B.Sc.

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

Master of Applied Science

Master of Applied Science (2012) McMaster University

(Electrical & Computer Engineering) Hamilton, Ontario, Canada

TITLE:	A Practical Coding Scheme For Broadcast Channel
AUTHOR:	Wenbo Sun B.Sc., (Electrical Engineering) Tianjin University, Tianjin, China
SUPERVISOR:	Dr. Jun Chen
NUMBER OF PAGES:	xi, 49

To my beloved parents

Abstract

In this thesis, a practical superposition coding scheme based on multilevel low-density parity-check (LDPC) codes is proposed for discrete memoryless broadcast channels. The simulation results show that the performance of the proposed scheme approaches the information-theoretic limits. We also propose a method for optimizing the degree distribution of multilevel LDPC codes based on the analysis of EXIT functions.

Acknowledgements

Firstly, I would like to show my deepest gratitude to my supervisor Dr Jun Chen, for providing me the opportunity to work on this interesting topic and offering continuous guidance, advice and support throughout the research.

Secondly, I would also like to thank Cheryl and all other Electrical and Computer Engineering administrative staff for their friendly assistance in the past two years. Further thanks go to my fellow graduate students in the ECE department, especially all my friends in the Communication Lab, for their encouragement and support.

Last, but not least, I would like to extend my gratitude to my family members in China, for their love, and for their words of kind encouragement, sent to me consistently regardless of the distance between us.

Abbreviations

DM-BC	Discrete Memoryless Broadcast Channel
BS-BC	Binary Symmetric Broadcast Channel
BSC	Binary Symmetric Channel
LDPC	Low-Density Parity-Check
DMC	Discrete Memoryless Channel
BM-SC	Binary Memoryless Symmetric Channel
EXIT	Extrinsic Information Transfer
BER	Bit Error Rate

Contents

List of Figures

Chapter 1

Introduction and Problem Statement

1.1 Background

In this section, the problem setup and some coding schemes for discrete memoryless broadcast channel (DM-BC) are reviewed.

1.1.1 Broadcast Channel Problem Setup

A two-receiver DM-BC $(\mathcal{X}, p(y_1, y_2|x), \mathcal{Y}_1 \times \mathcal{Y}_2)$ consists of three finite sets $\mathcal{X}, \mathcal{Y}_1, \mathcal{Y}_2$, and a conditional probability mass function (pmf) on $\mathcal{Y}_1 \times \mathcal{Y}_2$. The transmitter X wishes to send messages M_1 and M_2 to receivers Y_1 and Y_2 respectively.

As illustrated in Figure 1.1, a $(2^{nR_1}, 2^{nR_2}, n)$ code for DM-BC consists of two message sets $[1 : 2^{nR_1}]$ and $[1 : 2^{nR_2}]$. At the transmitter side, the encoder assigns a codeword $x^n(m_1, m_2)$ to each message pair (m_1, m_2). Meanwhile, at the receiver side, decoder

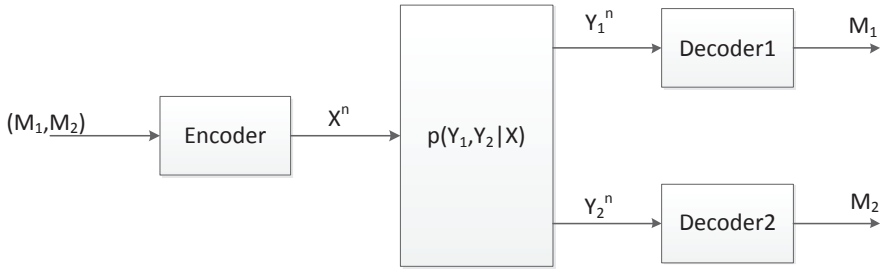

Figure 1.1: Channel model of broadcast channel

1 assigns an estimate $y_1^n(\hat{m}_1) \in [1 : 2^{nR_1}]$ or an error e to each received sequence y_1^n, and decoder 2 assigns an estimate $y_2^n(\hat{m}_2) \in [1 : 2^{nR_2}]$ or an error e to each received sequence y_2^n.

We assume that message pair (M_1, M_2) is uniformly distributed and that the average probability of error is defined as $P_e^{(n)} = P(\hat{M}_1 \neq M_1 \text{ or } \hat{M}_2 \neq M_2)$. A rate pair (R_1, R_2) is said to be achievable for DM-BC if there exists a sequence of $(2^{nR_1}, 2^{nR_2}, n)$ codes with $P_e^{(n)} \to 0$ as $n \to \infty$.

1.1.2 Time Sharing

It is well known [1] that, for a point-to-point discrete memoryless channel (DMC) with input X and output Y, any rate R below $\max_{p(x)} I(X;Y)$ is achievable. Note that the two-receiver DM-BC $(\mathcal{X}, p(y_1, y_2|x), \mathcal{Y}_1 \times \mathcal{Y}_2)$ introduced in Section 1.1.1 can be viewed as two separated point-to-point DMCs, as shown in Figure 1.2. As a consequence, the following rate region (see Figure 1.3) is achievable via time-sharing:

$$R_1 \leq t \max_{p(x)} I(X;Y_1), \quad R_2 \leq (1-t) \max_{p(x)} I(X;Y_2), \tag{1.1}$$

where $t \in [0, 1]$ is the time-sharing parameter.

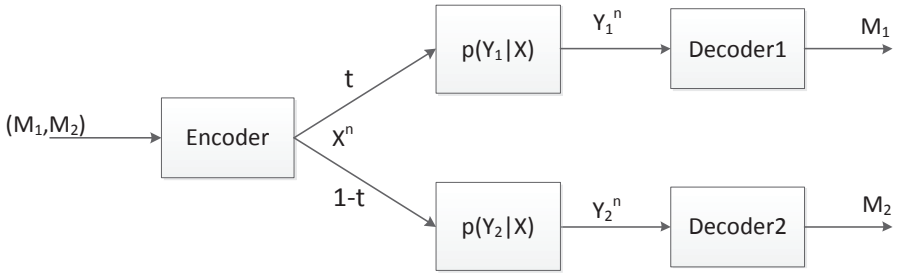

Figure 1.2: A broadcast channel viewed as two separate point-to-point channels

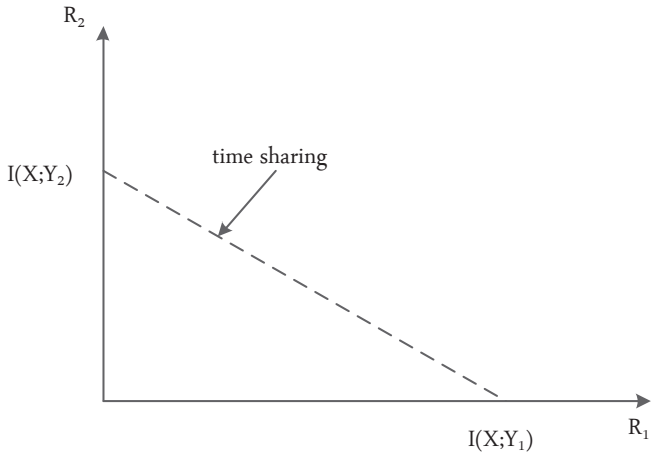

Figure 1.3: Rate region of time sharing

1.1.3 Superposition Coding

The superposition coding scheme was introduced by Cover in [2]. For DM-BC $(\mathcal{X}, p(y_1, y_2|x), \mathcal{Y}_1 \times \mathcal{Y}_2)$, the superposition coding scheme is described as follows:

- Codebook generation: Fix $p(u)p(v)$

  - Randomly and independently generate 2^{nR_2} sequences $v^n(m_2)$, $m_2 \in [1 : 2^{nR_2}]$, each according to $\prod_{i=1}^{n} p(v_i)$.
  - Randomly and independently generate 2^{nR_1} sequences $u^n(m_1)$, $m_1 \in [1 : 2^{nR_1}]$, each according to $\prod_{i=1}^{n} p(u_i)$.
  - Generate $2^{n(R_1+R_2)}$ sequences of $x^n(m_1, m_2)$ by function $f(u^n(m_1), v^n(m_2))$, where $u^n(m_1)$ and $v^n(m_2)$ are from two codebooks separately.

- Encode: Given the message pair (m_1, m_2), $x^n(m_1, m_2)$ is transmitted.

- Decode: Decoder 2 declares that a message $\hat{m_2}$ is sent if it is the unique message such that $(v^n(\hat{m_2}), y_2^n) \in T_\epsilon^{(n)}$, where $T_\epsilon^{(n)}$ is the joint typical set [7]; otherwise it declares an error.

  Decoder 1 declares that a message $\hat{m_1}$ is sent if it is the unique message such that $(x^n(\hat{m_1}, m_2), v^n(m_2), y_1^n) \in T_\epsilon^{(n)}$ for some m_2; otherwise it declares an error.

A rate pair (R_1, R_2) is achievable if it satisfies the conditions [2, 4, 5]:

$$\begin{aligned} R_1 &\le I(X; Y_1|V), \\ R_2 &\le I(V; Y_2), \\ R_1 + R_2 &\le I(X; Y_1) \end{aligned} \tag{1.2}$$

for some $p(u)p(v)$, where $X = f(U, V)$.

It can be seen from Figure 1.4 that the achievable rate region of superposition coding is larger than that of time-sharing.

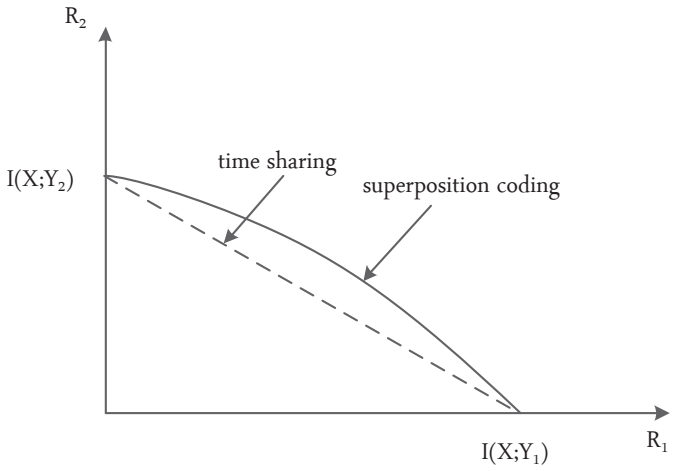

Figure 1.4: Rates regions of time-sharing and superposition coding

1.1.4 Binary Symmetric Broadcast Channel

In this thesis, we mainly focus on the binary symmetric broadcast channel (BS-BC), which is a special case of DM-BC. As shown in Figure 1.5, BS-BC consists of two binary symmetric channels (BSCs), where the crossover probabilities for the first BSC and the second BSC are p_1 and p_2 respectively. With no loss of generality, we assume $p_1 < p_2 < \frac{1}{2}$.

For BS-BC, the rate region of time-sharing is

$$R_1 \leq t(1 - H_b(p_1)), \quad R_2 \leq (1-t)(1 - H_b(p_2)) \tag{1.3}$$

while the rate region of superposition coding is given by

$$\begin{aligned} R_1 &\leq H_b(\alpha * p_1) - H_b(p_1), \\ R_2 &\leq 1 - H_b(\alpha * p_2) \end{aligned} \tag{1.4}$$

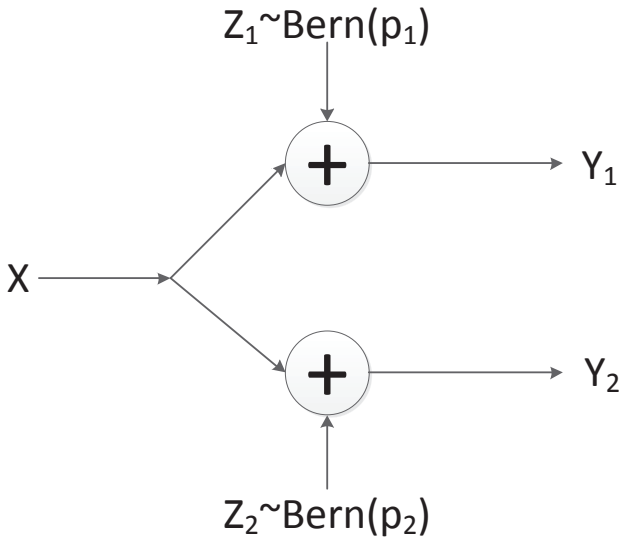

Figure 1.5: Channel model of BS-BC

for some $\alpha \in [0, 1/2]$. Here $H_b(p) = -p\log_2 p - (1-p)\log_2(1-p)$ and $\alpha * p = \alpha(1-p) + (1-\alpha)p$.

These two rate regions are depicted in Figure 1.6.

1.2 Motivation and Contribution of the Thesis

Recently the applications of low-density parity-check (LDPC) codes in conjunction with various message passing algorithms to channel coding problems have shown extremely good performance (see, e.g., [6]). It should be noted that these codes and message passing algorithms are mainly developed for binary-input symmetric-output channels for which the capacity-achieving distribution is the uniform distribution. However, in some scenarios one might be interested in the case where input distribution is not uniform. One notable example is the superposition coding scheme introduced in Section 1.1.3. In this case, the direct application of binary linear codes

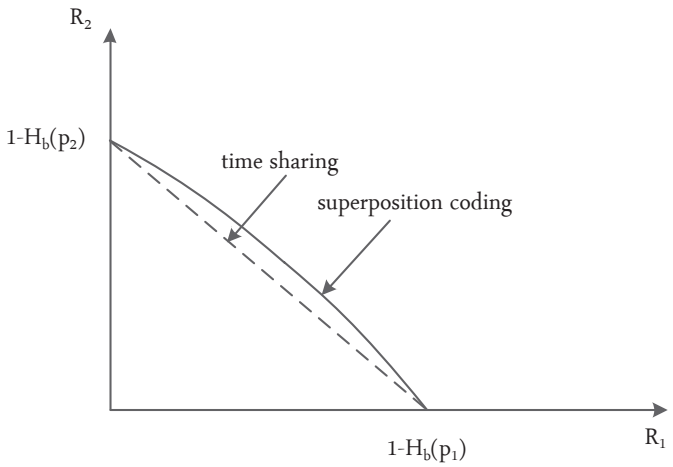

Figure 1.6: Rates regions of time-sharing and superposition coding for BS-BC

is suboptimal since they can only induce uniform input distribution. In order to generate the desired non-uniform input distribution, one can use multilevel linear codes constructed through deterministic mapping. The main contribution of this work is the implementation of multilevel codes in superposition coding as well as the associated code design method.

1.3 Organization of the Thesis

The thesis is structured as follows:

1. In Chapter 2, we show that the rate region of superposition coding can be achieved by multilevel codes, which serves as the theoretical foundation for the subsequent chapters.

2. LDPC codes are introduced in Chapter 3, where the belief propagation algorithm and the associated analysis methods, like density evolution, EXIT chart, *etc.* are also discussed.

3. In Chapter 4, we focus on superposition coding for BS-BC. A multilevel coding scheme based on LDPC codes as well as the associated degree distribution optimization method is proposed. Simulation results are also presented.

4. In Chapter 5, we conclude this thesis and suggest the directions for future work.

Chapter 2

Proof of Theoretical Bounds

2.1 Review of Linear Codes

For error correction codes used in communication, it is important that they are amendable to fast encoding and decoding and efficient storage representation.

Let us think about how well we can represent a code. In general, we need to write all of the codewords down into a codebook describing which sequence of k bits gets mapped to which codeword. A code $C : \{0,1\}^k \to \{0,1\}^n$ can be stored using $n2^k$ bits, which is exponential in space. In contrast, linear codes have succinct representation, specifically, only nk bits space is needed, for each linear code.

2.1.1 Achievable Rates of Linear Codes

We shall show that, for an arbitrary DMC $p(y|x)$, rate R is achievable by binary linear codes if $R < I(X;Y)$, where $I(X;Y)$ is evaluated using the binary uniform input.

- Random linear codebook generation: Let $k = \lceil nR \rceil$ and $(u_1, u_2, \ldots, u_k) \in \{0,1\}^k$ be the binary expansion of the message $m \in [0 : 2^k - 1]$. Generate a random codebook such that each codeword $x^n(u^k)$ is a linear function of u^k. Specifically, let

$$[\ x_1 \quad x_2 \quad \cdots \quad x_n\ ] = [\ u_1 \quad u_2 \quad \cdots \quad u_k\ ] \cdot \begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ g_{k1} & g_{k2} & \cdots & g_{kn} \end{pmatrix}$$

  where $g_{ij} \in \{0,1\}, i \in [1 : k], j \in [1 : n]$ are independent and identically distributed (i.i.d.) variables according to $Bern(1/2)$.

- Two properties:

  1. $X_1(m), \ldots, X_n(m)$ are i.i.d. $Bern(1/2)$ for each $m \neq 0$.
  2. $X^n(m)$ and $X^n(\tilde{m})$ are independent for each $m \neq \tilde{m}$.

  The proof for the two properties is in Appendix A.

- For the DMC, the transition probability is given by $p(y^n|x^n(m)) = \prod_{i=n}^{n} p(y_i|x_i(m))$.

- Encoding: Given message m, we transmit $x^n(m)$.

- Decoding: Let us denote received sequence as y^n. The receiver declares that a message $\hat{m}$ is sent if it is the unique message that $(x^n(\hat{m}), y^n) \in T_\epsilon^{(n)}$; otherwise, the receiver declares an error.

- Analysis of error probability: We assume that message m is sent.

  If $m = 0$, we assume it will cause a decoding error : $\varepsilon_1 :\ m = 0$. If $m \neq 0$, the

possible error cases are as below:

$$\varepsilon_2 := \{(X^n(0), Y^n) \notin T_\epsilon^{(n)}\},$$
$$\varepsilon_3 := \{(X^n(m), Y^n) \in T_\epsilon^{(n)} \text{ for some } m \neq 0\}.$$

By the union bound, $P(\varepsilon) = P(\varepsilon_1 \cup \varepsilon_2 \cup \varepsilon_3) \leq P(\varepsilon_1) + P(\varepsilon_2) + P(\varepsilon_3)$.

1. For the first term, $P(\varepsilon_1) = \frac{1}{2^{nR}}$ approaches to zero as $n \to \infty$.
2. For the second term, $P(\varepsilon_2)$ tends to zero as $n \to \infty$ by the law of large numbers.
3. For the third term, $(X^n(m), Y^n) \sim \prod_{i=1}^n p_X(x_i)p_Y(y_i)$, which means $X^n(m)$ and Y^n are pairwise independent. Thus $P\{(X^n(m), Y^n) \in T_\epsilon^{(n)}$ for some $m \neq 0\} \to 0$ as $n \to \infty$, if $R \leq I(X;Y) - \delta(\epsilon)$ by the packing lemma [7].

Note that since the probability of error averaged over codebooks goes to zero, It follows that there must exist a good linear code (n, k) with diminishing error probability as $n \to \infty$. This proves that $R < I(X;Y)$ is achievable for binary linear code, where $I(X;Y)$ is evaluated using the binary uniform input.

2.1.2 Converse

In this subsection we shall show that it is impossible to achieve rates above $I(X;Y)$ (evaluated using the binary uniform input) with binary linear codes. With no essential

loss of generality, we assume the decoding error probability is zero. Note that

$$\begin{aligned}
nR &= H(M) \\
&= I(M; Y^n) \\
&= I(M, X^n(M); Y^n) \\
&= H(Y^n) - H(Y^n|X^n) \\
&= \sum_{i=1}^{n} [H(Y_i|Y_1, Y_2, \ldots, Y_{i-1}) - H(Y_i|X^n, Y_1, Y_2, \ldots, Y_{i-1})] \\
&\leq \sum_{i=1}^{n} [H(Y_i) - H(Y_i|X^n, Y_1, Y_2, \ldots, Y_{i-1})] \\
&= \sum_{i=1}^{n} [H(Y_i) - H(Y_i|X_i)] \\
&= \sum_{i=1}^{n} I(X_i; Y_i).
\end{aligned}$$

Where M is the set of messages. It is shown in Appendix A that the distribution of X_i is $Bern(1/2)$. This completes the proof.

2.2 Proof of the Rate Region of Superposition Coding Achieved by Linear Codes

2.2.1 Introduction to Deterministic Mapping

It is shown in Section 2.1.1 that the input distribution induced by linear codes follows $Bern(1/2)$. We shall show that by using deterministic mapping, it is possible achieve arbitrary channel input distribution by linear codes.

A deterministic mapper is defined as $f : W \to X$, where $W \in \{0,1\}^m$, $X \in \mathcal{X}$ ($\mathcal{X}$ is the channel input alphabet) and m is the length of W. A simple example is illustrated in Figure 2.1 where W has a uniform distribution, $P(X = 0) = 1/4$ and $P(X = 1) = 3/4$. Note we can only achieve probabilities of form $k/2^m$. However, by increasing the value of m, we can approximate an arbitrary channel input distribution.

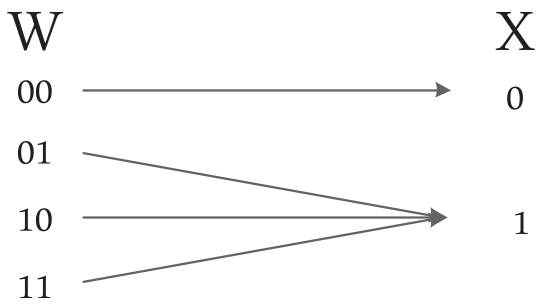

Figure 2.1: An example of deterministic mapping

2.2.2 Proof of Achievability by Linear Codes

Here we will prove the achievability for the superposition coding scheme with linear codes.

- Codebook generation:

    1. Generate 2^{nR_1} sequences of linear codeword $u^n(m_1)$ with deterministic mapping, where $m_1 \in [0 : 2^{nR_1} - 1]$ and U is according to $p(u)$.
    2. Generate 2^{nR_2} sequences of linear codeword $v^n(m_2)$ with deterministic mapping, where $m_2 \in [0 : 2^{nR_2} - 1]$ and V is according to $p(v)$.

3. Generate $2^{n(R_1+R_2)}$ $x^n(m_1, m_2)$ by function $f^n(u^n(m_1), v^n(m_2))$, where $u^n(m_1)$ and $v^n(m_2)$ are from two codebooks separately.

- Encode: Given the message pair (m_1, m_2), $x^n(m_1, m_2)$ is transmitted.

- Decode: Decoder 2 declares that a message $\hat{m_2}$ is sent if it is the unique message such that $(v^n(\hat{m_2}), y_2^n) \in T_\epsilon^{(n)}$; otherwise it declares an error.
  Decoder 1 declares that a message $\hat{m_1}$ is sent if it is the unique message such that $(x^n(\hat{m_1}, m_2), v^n(m_2), y_1^n) \in T_\epsilon^{(n)}$ for some m_2; otherwise it declares an error.

- Error probability analysis: We assume that message pair $(M_1, M_2) = (m_1, m_2)$ is sent.

  – Average probability of error for decoder 2

  If we assume it will cause a decoding error event ε_{21}: $m_1 \neq 0$ and $m_2 \neq 0$.
  If $m_1 = 0$ or $m_2 = 0$, we define the error cases as:

$$\varepsilon_{22} := \{(V^n(m_2), Y_2^n) \notin T_\epsilon^{(n)}\},$$
$$\varepsilon_{23} := \{(V^n(m_2'), Y_2^n) \in T_\epsilon^{(n)} \text{ for some } m_2' \neq m_2\}.$$

  1. For the first term, $P(\varepsilon_{21}) \leq \frac{1}{2^{nR_1}} + \frac{1}{2^{nR_2}}$ approaches to 0 as $n \to \infty$.
  2. For the second term, $P(\varepsilon_{22})$ tends to 0 as $n \to \infty$ by the law of large numbers.
  3. For the third term, $P(\varepsilon_{23})$ goes to 0 as $n \to \infty$, if $R_2 < I(V; Y_2)$ by the packing lemma.

  – Average probability of error for decoder 1

We assume $m_1 \neq 0$ and $m_2 \neq 0$. We divide the error events into 3 cases:

$$\varepsilon_{12} := \{(X^n(m_1, m_2), V^n(m_2), Y_1^n) \notin T_\epsilon^{(n)}\},$$
$$\varepsilon_{13} := \{(X^n(m_1', m_2), V^n(m_2), Y_1^n) \in T_\epsilon^{(n)} \text{ for some } m_1' \neq m_1\},$$
$$\varepsilon_{14} := \{(X^n(m_1', m_2'), V^n(m_2')Y_1^n) \in T_\epsilon^{(n)} \text{ for some } m_1' \neq m_1,\ m_2' \neq m_2\}.$$

1. By the law of large numbers, the term $P(\varepsilon_{12})$ tends to 0 as $n \to \infty$.
2. For $m_1' \neq m_1$, $X^n(m_1', m_2)$ is conditionally independent of $(X^n(m_1, m_2), Y_1^n)$ given $V^n(m+2)$ and is distributed according to $\prod_{i=1}^n p(x_i|v_i)$, where $X = f(U, V)$. By Packing Lemma, $P(\varepsilon_{13})$ tends to 0 as $n \to \infty$, if $R_1 < I(X(V, U); Y_1|V) = I(V, U; Y_1|V) = I(U; Y_1|V)$.
3. For $m_1' \neq m_1$ and $m_2' \neq m_2$, $X^n(m_1', m_2')$ is independent of $(X^n(m_1, m_2), Y_1^n)$. Hence, by Packing Lemma, $P(\varepsilon_{14})$ goes to 0 as $n \to \infty$, if $R_1 + R_2 < I(X; Y_1) = I(U, V; Y_1)$.

This proves that the rate region

$$\begin{aligned} R_1 &< I(U; Y_1|V), \\ R_2 &< I(V; Y_2), \\ R_1 + R_2 &< I(U, V; Y_1) \end{aligned} \tag{2.1}$$

is achievable by superposition coding scheme with linear codes. It will be seen in Chapter 4 that we propose a practical coding scheme with more structured multilevel codes.

Chapter 3

Low-Density Graph Codes

3.1 Introduction

Low-density parity-chek (LDPC) codes were invented by Gallager in [8]. LDPC codes are linear codes obtained from sparse bipartite graphs. Assume that $\mathcal{G}$ is a graph with n variable nodes and r check nodes. The sum of the neighboring positions of each check node is zero (see Figure 3.1).

Let H be the adjacency matrix of graph $\mathcal{G}$. Specifically, $\mathbf{H}$ is a binary $r \times n$ matrix such that the element (i, j) is 1 if and only if the i-th check node and the j-th variable node are connected in graph $\mathcal{G}$. The set of the vector $x = (x_1, \ldots, x_n)$ satisfying $\mathbf{H} \cdot x^T = 0$ forms a linear code. The matrix $\mathbf{H}$ is called the parity-check matrix for the code.

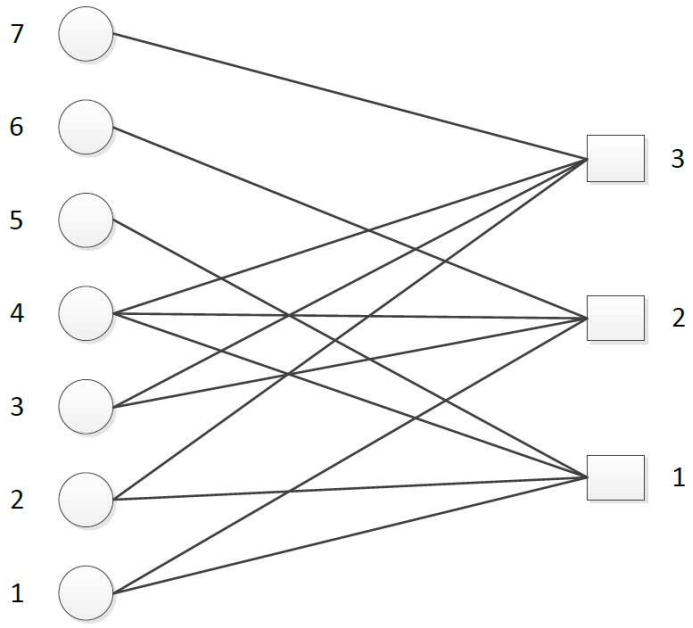

Figure 3.1: An example of LDPC codes. Codeword $x = (x_1, x_2, \ldots, x_7)$ satisfies the conditions that $x_1 + x_2 + x_4 + x_5 = 0$, $x_1 + x_3 + x_4 + x_6 = 0$ and $x_2 + x_3 + x_4 + x_7 = 0$.

3.2 Decoding Algorithm: Belief Propagation

Let us first introduce a general class of decoding algorithm for LDPC codes [9]. These algorithms are called message passing algorithms for the reason that messages passed from variable nodes v to check nodes c and then back to variable nodes v. Belief propagation algorithm, also known as sum-product message passing algorithm, is one subclass of message passing algorithms. The messages passed in belief propagation algorithm are probabilities. For each variable node v, one has probability from certain value of v and from previous passing round $c \to v$. For each check node c, one has probability just from previous passing round $v \to c$.

We use log-likelihoods instead of probabilities in our subsequent analysis. For a binary variable x, likelihood of x is $L(x) = p(x = 0)/p(x = 1)$. Given another variable y, conditional likelihood is $L(x|y) = p(x = 0|y)/p(x = 1|y)$. Similarly, the log-likelihood

is $\ln L(x)$ for variable x and the conditional log-likelihood is $\ln L(x|y)$ for variable x given y. If the distribution of variable x is uniform, then $L(x|y) = L(y|x)$ by Bayes' rules.

Let $m_{vc}^{(l)}$ be the message passed from variable node v to check node c at the l-th round iteration of this algorithm; $m_{cv}^{(l)}$ is defined similarly. The messages updating equations for both variable nodes and check nodes are described as

$$m_{vc}^{(l)} = \begin{cases} m_v, & \text{if } \ l = 0, \\ m_v + \sum_{c' \in C_v - \{c\}} m_{c'v}^{(l-1)}, & \text{if } \ l \geq 1. \end{cases} \tag{3.1}$$

$$m_{cv}^{(l)} = \ln \frac{1 + \prod_{v' \in V_c - \{v\}} m_{v'c}^{(l)}}{1 - \prod_{v' \in V_c - \{v\}} m_{v'c}^{(l)}},$$

where m_v is the message passed from the channel. C_v is the set of check nodes c connected to variable node v, and V_c is the set of variable nodes v connected to check node c.

3.3 Degree Distribution

The degree of a node is defined as numbers of neighboring nodes connected to this node. In [9], it is shown that the performance of LDPC codes can be significantly improved by optimizing the distributions of node degree. We define the node degree distribution as follows:

$$\lambda(x) = \sum_i \lambda_i x^{i-1}, \rho(x) = \sum_i \rho_i x^{i-1}, \tag{3.2}$$

where λ_i is the distribution of variable node with degree i and ρ_i is the distribution of check node with degree i.

Thus, the design rate is given by

$$R(\lambda, \rho) = 1 - \frac{\int_0^1 \rho(x)dx}{\int_0^1 \lambda(x)dx}. \tag{3.3}$$

3.4 Density Evolution

The average performance of an ensemble LDPC(n, λ, ρ) can be analyzed because of its recursive structure. An effective way to assess the performance of LDPC codes is density evolution [10, 11].

Consider a degree distribution pair (λ, ρ) and transmission over a binary memoryless symmetric channel (BM-SC) with distribution density α_0. Let

$$\alpha_l = \alpha_0 \otimes \lambda(\rho(\alpha_{l-1})), \tag{3.4}$$

where l is the times of iteration rounds and

$$\lambda(\alpha) = \sum_i \lambda_i \alpha^{\otimes(i-1)}, \rho(\alpha) = \sum_i \rho_i \alpha^{\odot(i-1)}, \tag{3.5}$$

where $\otimes$ denotes convolution and $\odot$ denotes convolution under the space $GF(2) \times [0, \infty)$.

3.4.1 Density Evolution at Variable Nodes

At variable nodes, the messages are represented by log-likelihoods in (3.1) that

$$m_{vc}(m_0, m_1, \ldots, m_{d_v-1}) = \sum_{i=0}^{d_v-1} m_i, \tag{3.6}$$

where $\sum_{i=0}^{d_v-1} m_i$ is simply the convolution of their densities. Let $\alpha_0, \alpha_1, \ldots, \alpha_{d_v-1}$ denote densities of messages $m_0, m_1, \ldots, m_{d_v-1}$. Thus, density evolution for variable nodes is

$$m_{vc}(\alpha_0, \alpha_1, \ldots, \alpha_{d_v-1}) = \alpha_0 \otimes \alpha_1 \otimes \cdots \otimes \alpha_{d_v-1}. \tag{3.7}$$

3.4.2 Density Evolution at Check Nodes

For the belief propagation algorithm, 0 is mapped to 1 and 1 is mapped to -1. Let us denote belief $p(y_i|x_i = 1)$ by μ_1 and $p(y_i|x_i = -1)$ by μ_{-1}. For check nodes, we have

$$\mu_1 - \mu_{-1} = \prod_{k=1}^{d_c-1} (\mu_1^k - \mu_{-1}^k), \tag{3.8}$$

and also $\mu_1 + \mu_{-1} = \prod_{k=1}^{d_c-1}(\mu_1^k + \mu_{-1}^k)$=1.

If the m is a log-likelihood $\ln \frac{\mu_1}{\mu_{-1}}$, then it follows that

$$\mu_1 - \mu_{-1} = \frac{e^m - 1}{e^m + 1} = \tanh(m/2). \tag{3.9}$$

Conversely, $\ln \frac{\mu_1}{\mu_{-1}} = \ln \frac{1+(\mu_1-\mu_{-1})}{1-(\mu_1-\mu_{-1})}$. Thus the definition for check node message passing is

$$m_{cv}(m_1, \ldots, m_{d_c-1}) = \ln \frac{1 + \prod_{i=1}^{d_c-1} \tanh \frac{1}{2} m_i}{1 - \prod_{i=1}^{d_c-1} \tanh \frac{1}{2} m_i}. \tag{3.10}$$

We can represent density of (μ_1, μ_{-1}) by $(\ln \operatorname{sgn}(\mu_1 - \mu_{-1}), -\ln|\mu_1 - \mu_{-1}|)$, since $\tanh(\frac{m}{2}) = \mu_1 - \mu_{-1}$. By this transform, densities are mapped into $GF(2) \times [0, \infty)$. The density of sum on check node is the convolution of densities pair $(\ln \operatorname{sgn} m$, $-\ln|\tanh(\frac{m}{2})|)$ according to (3.10). Therefore, we have

$$m_{cv}(\alpha_1, \alpha_2, \ldots, \alpha_{d_c-1}) = \alpha_1 \odot \alpha_2 \odot \cdots \odot \alpha_{d_c-1}. \tag{3.11}$$

3.5 Extrinsic Information Transfer Function

Extrinsic Information Transfer (EXIT) function [12, 13] is a useful and intuitive tool for analyzing the performance of LDPC codes. In Section 3.4, density evolution provides an exact assess of decoding performance by tracking message density during each decoding iteration. However, the complexity of density evolution is huge. In contrast, using EXIT functions we only need to track one single parameter (mutual information) rather than densities.

Let X be a codeword of length n chosen from codebook C. Let Y be a sequence of length n received after transmitting X over BM-SC($\mathbf{h}$), where $\mathbf{h}$ denotes the entropy passed from this BM-SC. The EXIT function of i-th bit in the codeword is defined as

$$h_i(\mathbf{h}) = H(X_i|Y_{\sim i}(\mathbf{h})). \tag{3.12}$$

The average EXIT function is

$$h(\mathbf{h}) = \frac{1}{n}\sum_{i=1}^{n} H(X_i|Y_{\sim i}(\mathbf{h})) = \frac{1}{n}\sum_{i=1}^{n} h_i(\mathbf{h}). \tag{3.13}$$

The EXIT function tracks entropy passed from input to output. Note that EXIT function is defined by $H(X_i|Y_{\sim i})$ instead of $H(X_i|Y)$, since we focus on the "extrinsic" information transfered from other symbols $Y_{\sim i}$ rather than Y_i.

In the next chapter, we will analyze the error probability in decoding process with EXIT functions.

Chapter 4

A Practical Superposition Coding Scheme Based on Multilevel LDPC Codes

4.1 Introduction to Multilevel Coding

It is well known that linear codes can be directly used to achieve the information rate evaluated with the uniform input, which we have proven in Chapter 2. Given the existing results on the design of LDPC codes for binary symmetric channels, it suffices to say that LDPC codes provide good practical solution when the desired input distribution is the uniform distribution [14].

However, the input distribution is not uniform in many cases, e.g. superposition coding. For such coding schemes, linear codes can not be used directly, for linear codes only induce uniform distribution.

In this chapter, we will construct multilevel codes by using a set of binary linear codes

and a deterministic mapper to achieve a general mutual information rate (possibly evaluated using a non-uniform input distribution) over the channel. However, the mapper does not necessarily have to be a one-to-one function. In our proposed superposition coding scheme, multilevel codes with several layers can achieve rates close to the mutual information rate evaluated with an arbitrary input distribution P_X.

4.2 Coding Scheme

4.2.1 Deterministic Mapping

The deterministic mapper is defined in Section 2.2.1. In our practical coding scheme, we let U be distributed according to $Bern(p)$, where $P(U = 0) = p$ and $P(U = 1) = 1 - p$. A possible deterministic mapper is shown in Figure 4.1, where $W \in \{0, 1\}^3$ and is uniformly distributed.

4.2.2 Proposed Coding Scheme

Our proposed superposition coding scheme is as follows:

- Codebook generation: Design two binary code generator matrices $\mathbf{G_1}[k \times n]$ and $\mathbf{G_2}[k \times tn]$, where k is the length of message, n is the length of the codeword and t is the numbers of multilevel.

  Generate codewords $V^n = m_2 \cdot \mathbf{G_2}$, where each v^n is i.i.d. $Bern(1/2)$ sequence.

  Generate codewords $\hat{U}^n = m_1 \cdot \mathbf{G_1}$, which is a $2^k \times tn$ codebook and each codeword is i.i.d. $Bern(1/2)$ sequence. By the deterministic mapping function

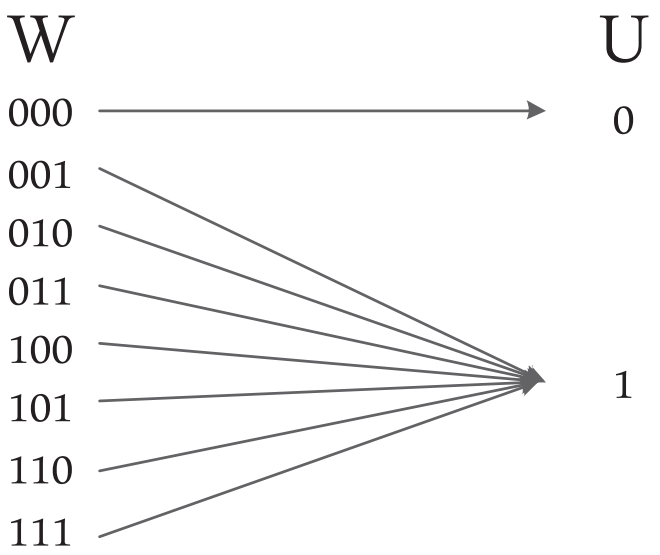

Figure 4.1: A three-level deterministic mapping

we mentioned above, every m columns $\mathbf{W_1} \cdots \mathbf{W_t}$ in the codebook are mapped into one column $\mathbf{U}$.

$$\mathbf{W_1} \quad \cdots \quad \mathbf{W_t} \rightarrow \mathbf{U}$$

For the whole codebook,

$$[\ \mathbf{W_1} \quad \cdots \quad \mathbf{W_t} \quad \cdots \quad \mathbf{W_1} \quad \cdots \quad \mathbf{W_t}\ ] \rightarrow [\ \mathbf{U} \quad \cdots \quad \mathbf{U}\ ]$$

So, we can get a $2^k \times n$ codebook that every codeword U^n is i.i.d. $Bern(p)$ sequence.

- Encode: Given (u, v), $x^n(m_1, m_2) = u^n(m_1) \oplus v^n(m_2)$ is transmitted, where $\oplus$ is modulo-2 addition.

- Decode:

1. Decoder 2 decodes m_2 from $y_2^n = v^n(m_2) \oplus (u^n(m_1) \oplus z_2^n)$ by treating $u^n(m_1)$ as noise. m_2 can be decoded with probability of error $\to 0$ as $n \to \infty$ if $R_2 < I(V; V \oplus U \oplus Z_2) = H(V) - H(U \oplus Z_2)$

2. Decoder 1 uses successive cancelation: It first decodes m_2 from $y_1^n = v^n(m_2) \oplus (u^n(m_1) \oplus z_1^n)$, subtracts off $v^n(m_2)$, then decodes m_1 from $u^n(m_1) \oplus z_1^n$. m_1 can be decoded with probability of error $\to 0$ as $n \to \infty$ if $R_1 < I(U; U \oplus Z_1) = H(U \oplus Z_1) - H(Z_1)$

4.2.3 Information Rate Achieved by Multilevel Codes

Next, let us show the mutual information across a channel induced by an arbitrary input distribution can be achieved by the multilevel coding scheme proposed above.

Argument: The proposed coding scheme can achieve the mutual information across the DMC induced by an arbitrary$I(W;Y) = I(X;Y)$.

Proof:

$$\begin{aligned} I(W; Y, X) &= I(W; X) + I(W; Y|X) \\ &= I(W; Y) + I(W; X|Y) \end{aligned}$$

Note that $I(W; Y|X) = 0$, which is due to the fact that $W \to X \to Y$ forms a Markov

chain. Thus

$$\begin{aligned} I(W;Y) &= I(W;X) - I(W;X|Y) \\ &= (H(X) - H(X|W)) - (H(X|Y) - H(X|W,Y)) \\ &= H(X) - H(X|Y) \\ &= I(X;Y) \end{aligned}$$

where $H(X|W,Y) = H(X|W) = 0$, for X is a function of W.

Hence we can achieve the rate using the proposed multilevel coding scheme and

$$\begin{aligned} I(X;Y) &= I(W;Y) \\ &= I(W_1;Y) + I(W_2;Y|W_1) + I(W_3;Y|W_1,W_2) \end{aligned}$$

This proof shows that the deterministic mapping from W to X does not incur a rate loss.

4.3 Degree Distribution Optimization

In our proposed multilevel coding scheme, the distribution of X is $P(X=1) = 1-p$ and $P(X=0) = p$. We will find optimized degree distribution by analyzing the EXIT function.

4.3.1 EXIT Function for Check-to-Variable Nodes

For a check node with degree m, the error probability of edge connected is ϵ, as illustrated in Figure 4.2. Note that only if odd numbers of edges are in error, the check node is in error. Even numbers of errors do not introduce any error.

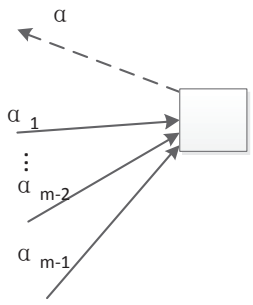

Figure 4.2: Check-to-variable node message

The error probability of this outgoing edge combined with $m-1$ incoming messages is

$$\begin{aligned}\alpha_{BSC(\epsilon)}^{\odot(m-1)} &= \sum_{k=0}^{\frac{m-1}{2}} \binom{m-1}{2k+1} \epsilon^{(2k+1)} \bar{\epsilon}^{m-1-(2k+1)} \\ &= \frac{(\epsilon+\bar{\epsilon})^{m-1} - (\bar{\epsilon}-\epsilon)^{m-1}}{2} \\ &= \frac{1-(1-2\epsilon)^{m-1}}{2}\end{aligned} \tag{4.1}$$

where $\bar{\epsilon} = 1 - \epsilon$.

The EXIT function of this check node with degree m is

$$\begin{aligned}h_c^{m-1}(\mathbf{h}) &= H(\alpha_{BSC(\epsilon_c)}^{\odot(m-1)}) \\ &= H(\alpha_{BSC(\frac{1-(1-2\epsilon_c)^{m-1}}{2})}) \\ &= H_b(\frac{1-(1-2\epsilon_c)^{m-1}}{2})\end{aligned} \tag{4.2}$$

where $\epsilon_c = H_b^{-1}(\mathbf{h})$ and $\mathbf{h}$ is the entropy passed from variable node to check node.

4.3.2 EXIT Function for Variable-to-Check Nodes

Assuming a variable node with degree m, and the error probability of each edge is ϵ, as illustrated in Figure 4.3.

The conditional probability is

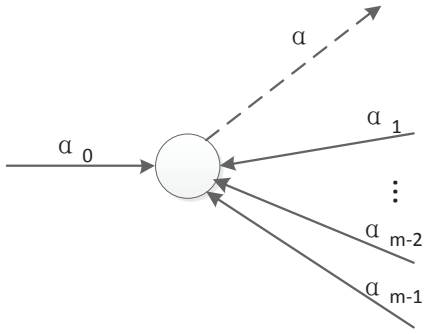

Figure 4.3: Variable-to-Check node message

$$\begin{aligned} p(X_j = 1|Y_{\sim j}) &= \frac{p(Y_{\sim j}|X_j = 1)p(X_j = 1)}{p(Y_{\sim j})} \\ &= \frac{\sum_{i=0}^{m-1} \binom{m-1}{i}(1-p)\epsilon^i \bar{\epsilon}^{m-1-i}}{\sum_{i=0}^{m-1} \binom{m-1}{i}(1-p)\epsilon^i \bar{\epsilon}^{m-1-i} + \sum_{i=0}^{m-1} \binom{m-1}{i} p\bar{\epsilon}^i \epsilon^{m-1-i}} \\ &= \frac{(1-p)\bar{\epsilon}^{m-1-2i}}{(1-p)\bar{\epsilon}^{m-1-2i} + p\epsilon^{m-1-2i}}, \end{aligned}$$

and

$$\begin{aligned} p(X_j = 0|Y_{\sim j}) &= 1 - p(X_j = 1|Y_{\sim j}) \\ &= \frac{p\epsilon^{m-1-2i}}{(1-p)\bar{\epsilon}^{m-1-2i} + p\epsilon^{m-1-2i}}. \end{aligned} \tag{4.3}$$

So, the EXIT function of variable node with degree m

$$\begin{aligned} h_v^{m-1}(\mathbf{h}) =& H(\alpha_{BSC(\epsilon_0)} \otimes \alpha_{BSC(\mathbf{h})}^{\otimes(m-1)}) \\ =& \sum_{i=0}^{m-1} \binom{m-1}{i} \epsilon_0 \epsilon^i \bar{\epsilon}^{m-1-i} H_b\left(\frac{(1-p)\epsilon_0 \bar{\epsilon}^{m-1-2i}}{(1-p)\epsilon_0 \bar{\epsilon}^{m-1-2i} + p\bar{\epsilon}_0 \epsilon^{m-1-2i}}\right) \\ &+ \sum_{i=0}^{m-1} \binom{m-1}{i} \bar{\epsilon}_0 \epsilon^i \bar{\epsilon}^{m-1-i} H_b\left(\frac{(1-p)\bar{\epsilon}_0 \bar{\epsilon}^{m-1-2i}}{(1-p)\bar{\epsilon}_0 \bar{\epsilon}^{m-1-2i} + p\epsilon_0 \epsilon^{m-1-2i}}\right), \end{aligned} \tag{4.4}$$

where $\epsilon = \frac{1-(1-2\epsilon_c)^{d_c-1}}{2}$, d_c is the degree of the corresponding check node and $\epsilon_c = H_b^{-1}(\mathbf{h})$, $\mathbf{h}$ is the entropy of the previous iteration.

4.3.3 Degree Distribution Optimization via Linear Programming

If the entropy at the output of the variable nodes is $\mathbf{h}$, after one further iteration, it becomes

$$v_{\mathbf{h}}(c(\mathbf{h})) = \sum_i \lambda_i h_v^{i-1}(c(\mathbf{h})) \tag{4.5}$$

The condition for progress at each iteration is $v_{\mathbf{h}}(c(\mathbf{h})) \leq \mathbf{h}$. This formulation is linear with the variable degree fraction λ_i. If check node distribution ρ is fixed, we can therefore optimize λ by linear programming techniques. Assuming crossover probability ϵ_0 is fixed, by the EXIT function h_v^i we introduced above, the corresponding linear program is

$$\max\{\sum_{i\geq 2} \frac{\lambda_i}{i} | \lambda_i \geq 0; \sum_{i\geq 2} \lambda_i = 1; \sum_{i\geq 2} \lambda_i h_v^{i-1}(c(\mathbf{h})) \leq \mathbf{h}; \mathbf{h} \in (0,1)\} \tag{4.6}$$

For message V, no multilevel coding is involved. So V is uniformly distributed.

We can use the same distribution optimization function above, but the entropy of variable node with degree m is

$$h_v^{m-1}(\mathbf{h}) = \sum_{i=0}^{m-1} \binom{m-1}{i} \epsilon_0 \epsilon^i \bar{\epsilon}^{m-1-i} H_b\left(\frac{\epsilon_0 \bar{\epsilon}^{m-1-2i}}{\epsilon_0 \bar{\epsilon}^{m-1-2i} + \bar{\epsilon}_0 \epsilon^{m-1-2i}}\right) + \sum_{i=0}^{m-1} \binom{m-1}{i} \bar{\epsilon}_0 \epsilon^i \bar{\epsilon}^{m-1-i} H_b\left(\frac{\bar{\epsilon}_0 \bar{\epsilon}^{m-1-2i}}{\bar{\epsilon}_0 \bar{\epsilon}^{m-1-2i} + \epsilon_0 \epsilon^{m-1-2i}}\right). \tag{4.7}$$

4.4 Practical Decoding Scheme

In our practical scheme, we set $t = 3$ which generates non-uniform distribution $P(X = 1) = 7/8$ and $P(X = 0) = 1/8$. By linear optimization function (4.6) and (4.7), some optimized degree distributions we find are given in Table 4.1 and Table 4.2.

Table 4.1: Optimized variable node degree distribution with the check node degree distribution fixed

degree	$\rho_7 = 1$	$\rho_6 = 1$	$\rho_9 = 1$
2	0.1369	0.0580	0.0455
3	0.0019	0.1545	0.0015
4	0.0192	0.0070	0.0265
5	0.0326	0.0325	0.0115
6	0.3292	0.1700	0.0590
7	0.1570	0.3640	0.5380
8	0.1748	0.2140	0.1820
9	0.1464	0	0.1250
10	0	0	0.0110
	$R = 0.1496$	$R = 0.1587$	$R = 0.1658$

For decoder 2, it faces a simple point-to-point channel as shown in Figure 4.4, which is equivalent to Figure 4.5, where U is considered as noise.

Now, let us consider about X_U. For decoder 1, first decode V while treating U as noise and then subtract V from the received codeword and further decode U. Note

Table 4.2: Optimized variable node degree distribution with the check node degree distribution fixed

degree	$\rho_6 = 0.8,\ \rho_7 = 0.2$	$\rho_5 = 1$
2	0.1865	0.2560
3	0.0745	0.1975
4	0.2285	0.1105
5	0.0615	0.0845
6	0.3225	0.0210
7	0.0415	0.1335
8	0.0850	0.1970
9	0	0.0085
	$R = 0.2562$	$R = 0.3144$

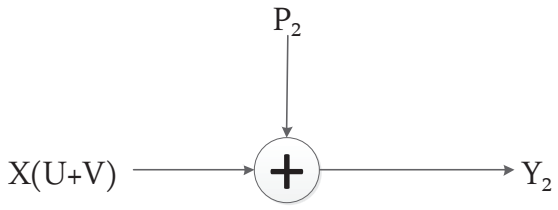

Figure 4.4: Channel model for decoder 2

that X_U is associated with three-level coding, and the probability of W_1, W_2 and W_3 are presented in Figure 4.6.

4.5 Simulation Results

We implement the algorithm using MATLAB. Some optimized degree distributions of LDPC codes are demonstrated in the last section for this binary symmetric broadcast channel.

Four rate pairs of (R_1, R_2) are chosen as shown in Figure 4.7. The simulation results

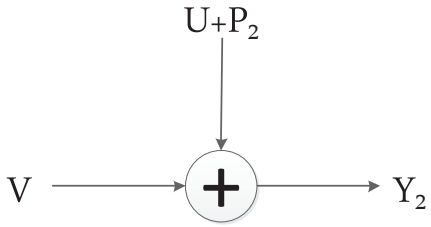

Figure 4.5: Equivalent channel model for decoder 2

are shown in Figure 4.8∼4.15. We test the performances of the proposed multilevel LDPC coding scheme by setting block length =5,000. Therefore, the length of three-level code used in the test cases is 15,000. The maximum iterations in the message passing algorithm is 100. For each simulation case, 5,000 sequences are tested. According to degree distribution optimization function (4.6), we generate four $\mathbf{H}_1$ and four $\mathbf{H}_2$ with corresponding rate pairs shown in Figure 4.7. The Shannon limits of crossover probabilities for the two decoders are also shown in each figure. The simulation results are close to the theoretical bound in these cases, validating our coding scheme.

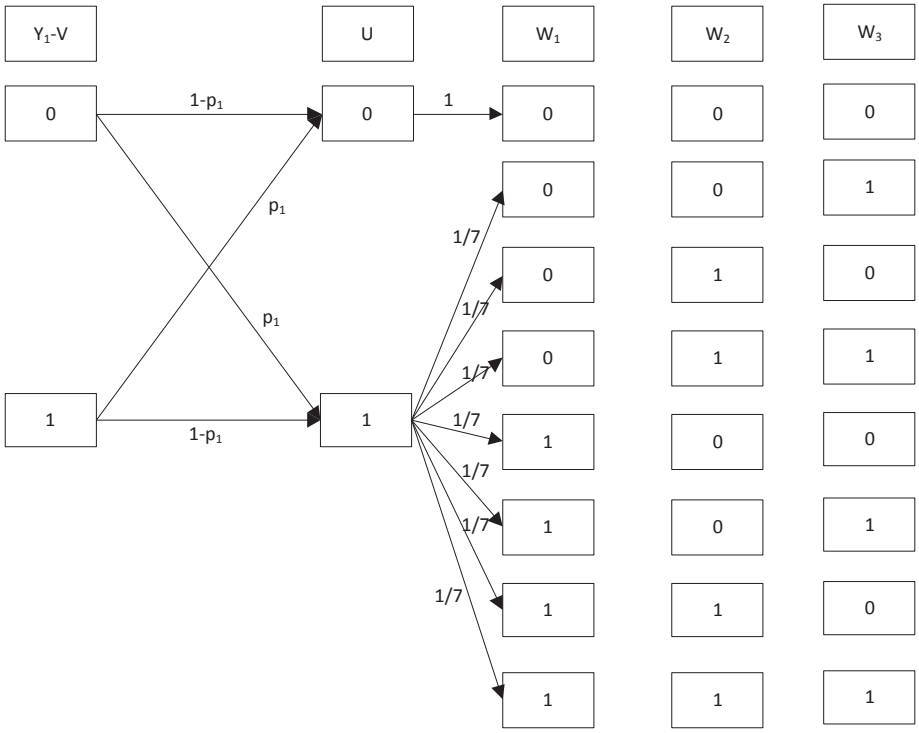

Figure 4.6: Decoding scheme for decoder 1

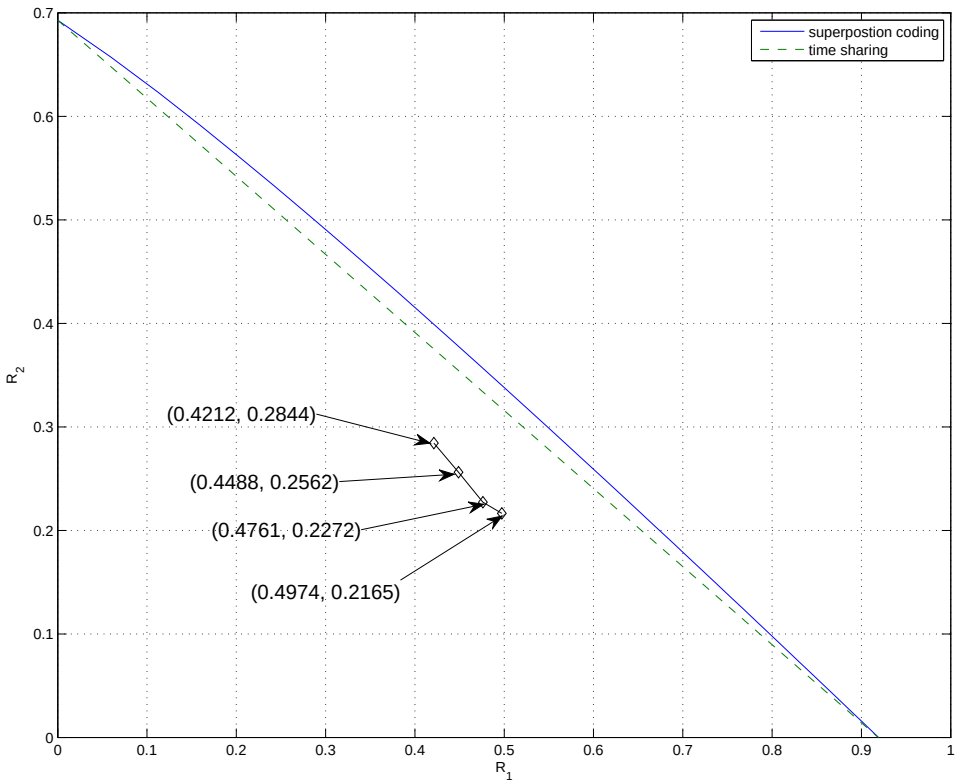

Figure 4.7: Four rate pairs are chosen when crossover probabilities $p_1 = 0.010$ and $p_2 = 0.055$

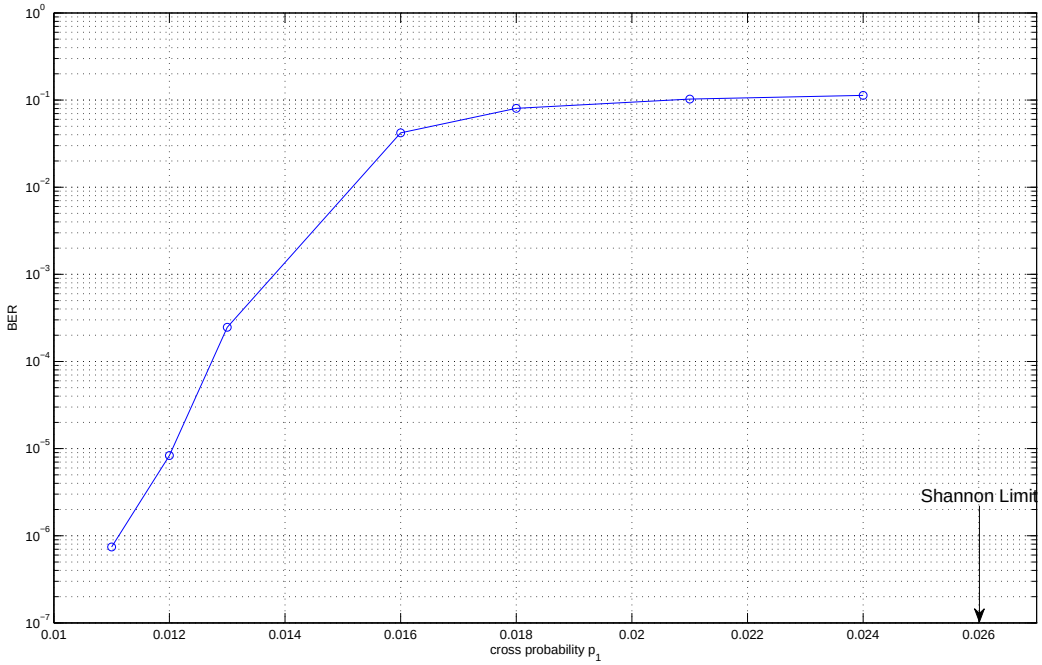

Figure 4.8: Performance of decoder 1 when R_1 is 0.4212

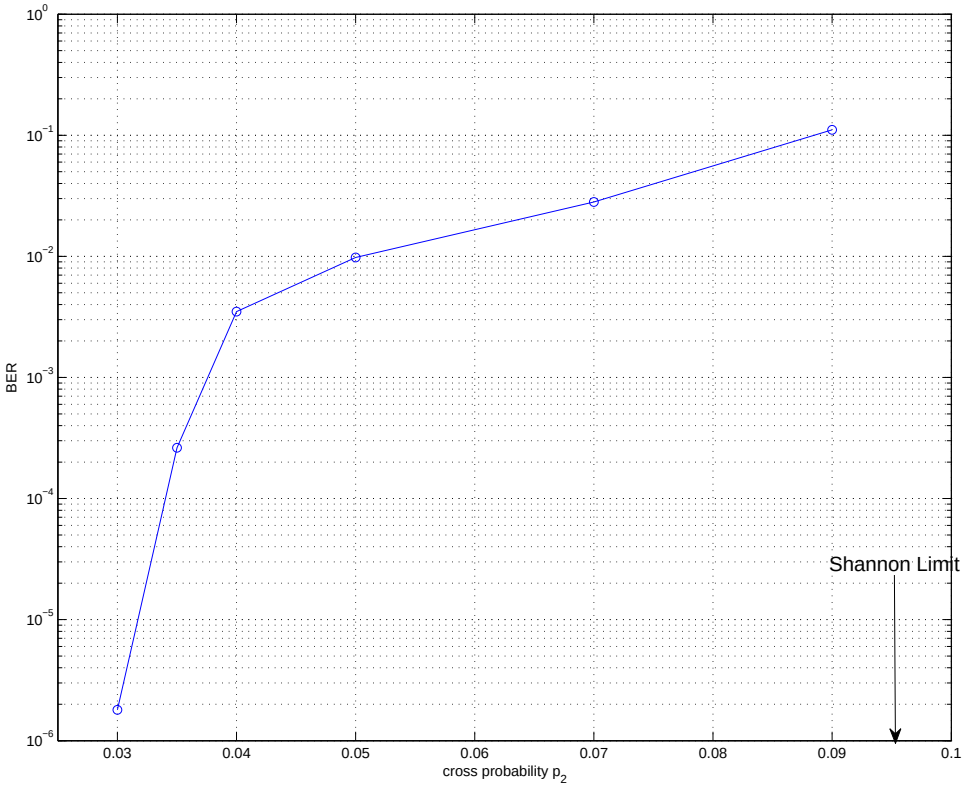

Figure 4.9: Performance of decoder 2 when R_2 is 0.2844

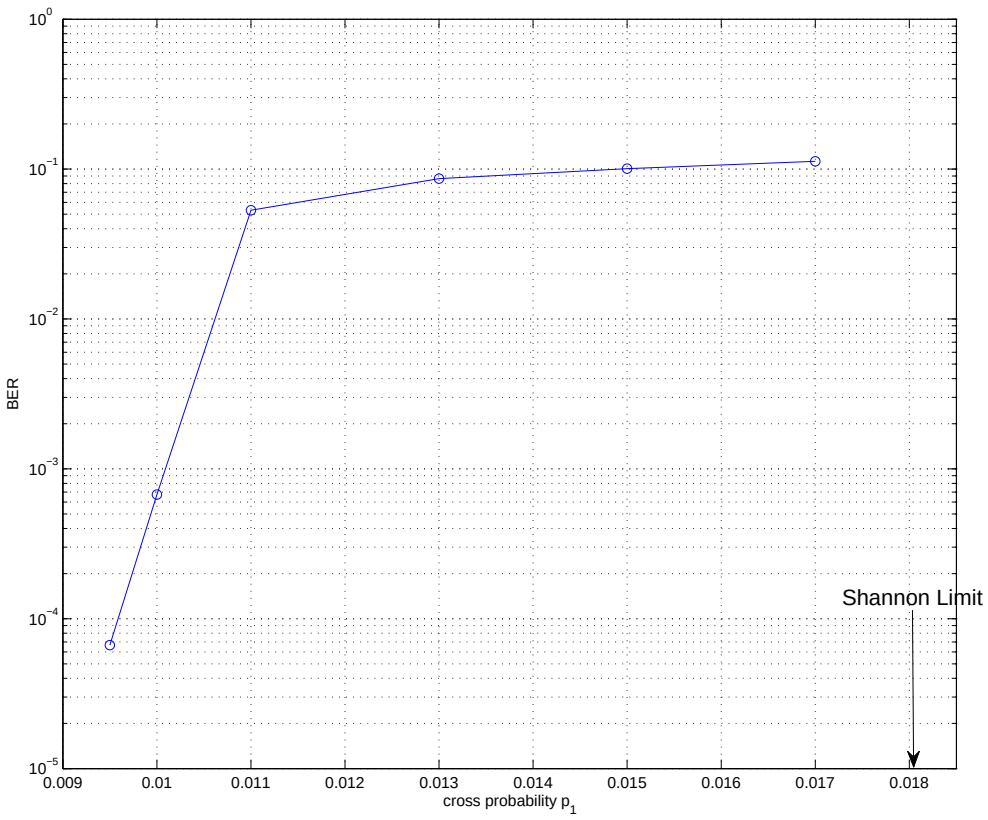

Figure 4.10: Performance of decoder 1 when R_1 is 0.4488

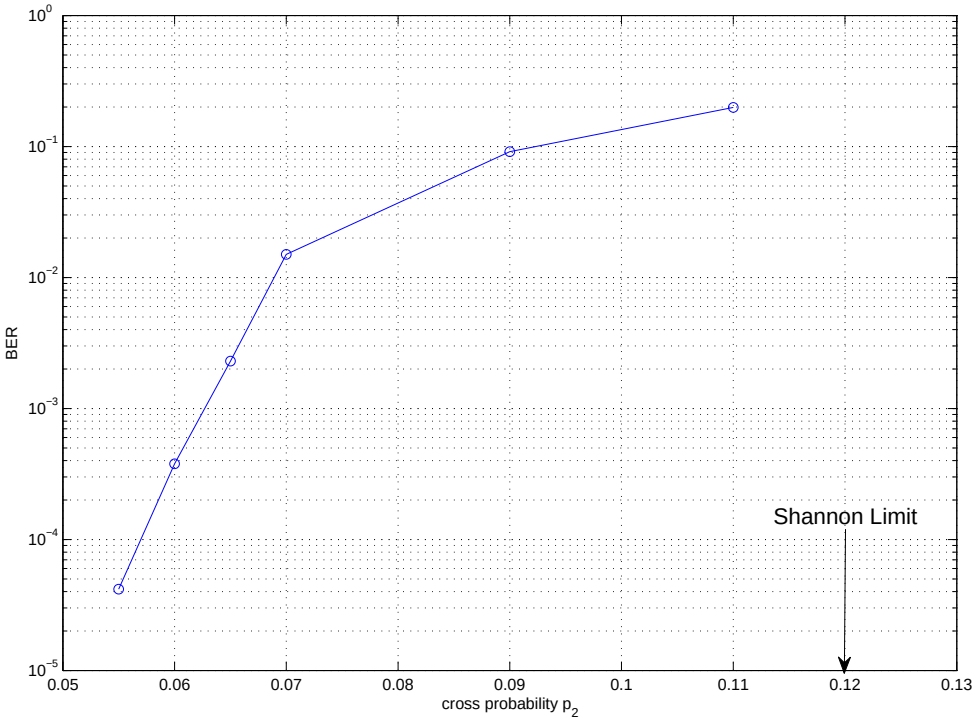

Figure 4.11: Performance of decoder 2 when R_2 is 0.2562

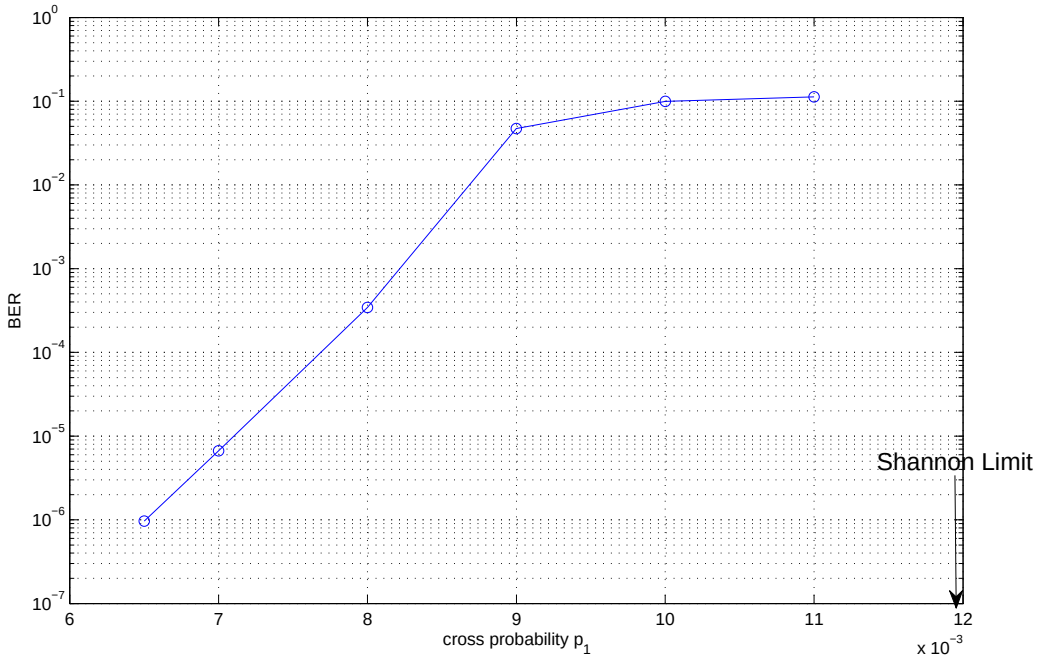

Figure 4.12: Performance of decoder 1 when R_1 is 0.4761

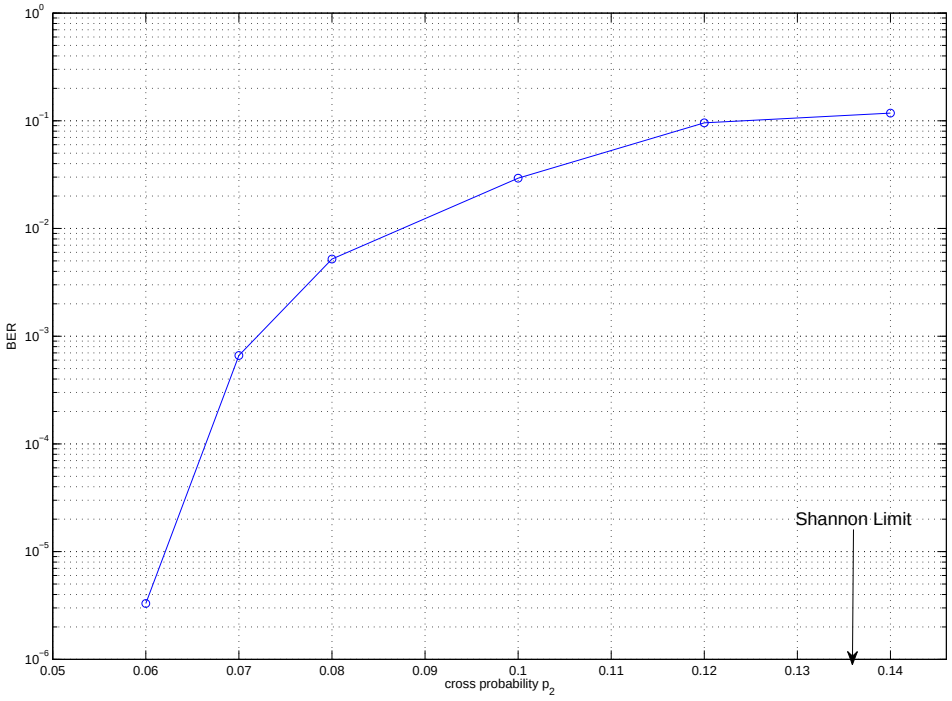

Figure 4.13: Performance of decoder 2 when R_2 is 0.2272

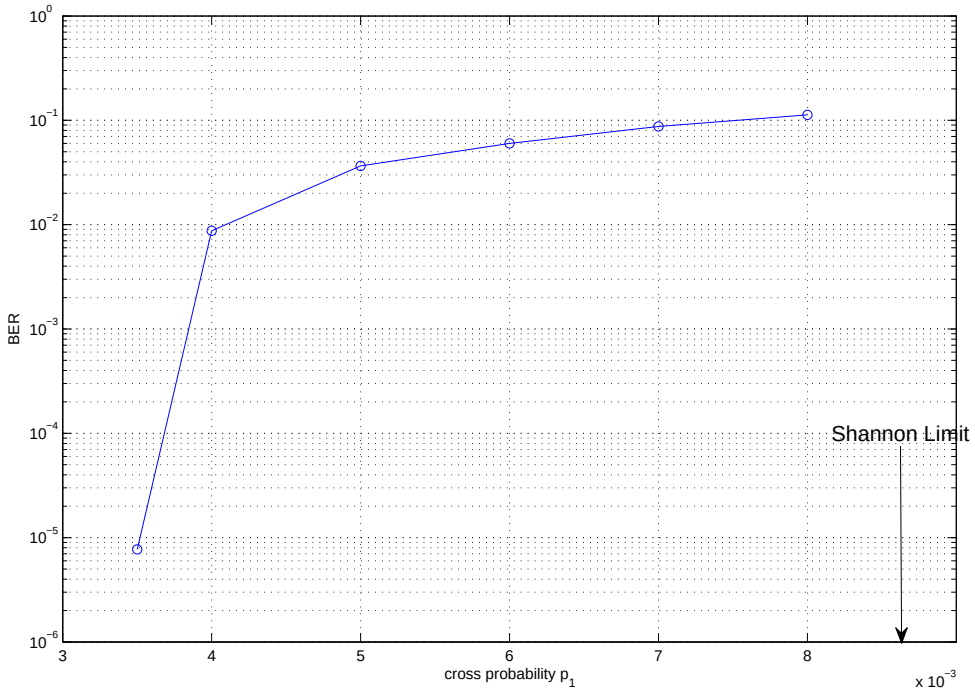

Figure 4.14: Performance of decoder 1 when R_1 is 0.4974

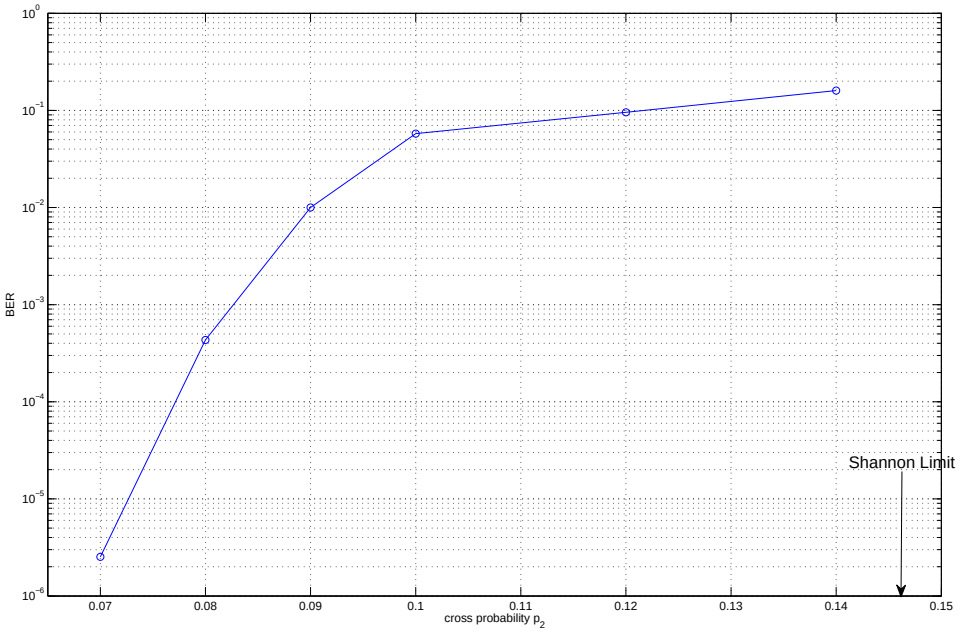

Figure 4.15: Performance of decoder 2 when R_2 is 0.2165

Chapter 5

Conclusion and Future Works

5.1 Conclusion

In this thesis, a practical superposition coding scheme based on multilevel low-density parity-check (LDPC) codes is proposed for discrete memoryless broadcast channels. The simulation results show that the performance of the proposed scheme approaches the information-theoretic limits. We also propose a method for optimizing the degree distribution of multilevel LDPC codes based on the analysis of EXIT functions.

5.2 Future Work

There are two possible directions for future work. Firstly, instead of using EXIT functions, one can optimize the degree distribution of multilevel codes using density evolution [15], which may lead to better performance. Secondly, in this thesis we have mainly focused on BS-BC; the coding problem for other broadcast channels, e.g. AWGN broadcast channel [7], are worth investigating.

Appendix A

Proof of the Properties of Linear Codes

The linear codebook is generated as follows: Let $k = \lceil nR \rceil$ and $(u_1, u_2, \ldots, u_k) \in \{0,1\}^k$ be the binary expansion of the message $m \in [1 : 2^k]$. Generate a random codebook that each codeword $x^n(u^k)$ is a linear function of u^k. Let

$$[\ x_1 \quad x_2 \quad \cdots \quad x_n\ ] = [\ u_1 \quad u_2 \quad \cdots \quad u_k\ ] \cdot \begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ g_{k1} & g_{k2} & \cdots & g_{kn} \end{pmatrix}$$

where $g_{ij} \in \{0,1\}, i \in [1 : k], j \in [1 : n]$ are independent and identically distributed (i.i.d.) according to $Bern(1/2)$.

1. Argument: $X_1(u^k), \ldots, X_n(u^k)$ are i.i.d. $Bern(1/2)$ for each u^k.

Proof: Note that

$$X_m(u^k) = u^k \cdot \mathbf{g_m}$$
$$X_n(u^k) = u^k \cdot \mathbf{g_n}$$

where $X_m(u^k)$ is the m-th position in codeword and $\mathbf{g_m}$ is the m-th column vector of the generator matrix $\mathbf{G}$. Thus $X_m(u^k)$ is a linear combination of $\mathbf{g_m}$, and $X_n(u^k)$ is a linear combination of g_n. Every element in $\mathbf{G}$ is an i.i.d. $Bern(1/2)$ random variable, so $X_m(u^k)$ and $X_n(u^k)$ are independent. Meanwhile, the sum of two independent $Bern(1/2)$ random variables is also a $Bern(1/2)$ random variable, which proves that $X_m(u^k)$ and $X_n(u^k)$ follow $Bern(1/2)$ distribution. Therefore, all elements in $X^n(u^k)$ are i.i.d. $Bern(1/2)$ random variables.

2. Argument: $X^n(u^k)$ and $X^n(\tilde{u}^k)$ are independent for each $u^k \neq \tilde{u}^k$.

   Proof: $X^n(u^k)$ is a linear summation of i-th row of $\mathbf{G}$ for $u_i = 1$, where i is from 1 to k. For u^k and $\tilde{u}^k$, we assume there is one position i different in these two messages, that is $u_i^k = 0$ and $\tilde{u}_i^k = 1$. Thus, $X^n(\tilde{u}^k) = X^n(u^k) + \bar{\mathbf{g}}_\mathbf{i}$, where $\bar{\mathbf{g}}_\mathbf{i}$ is the i-th row of $\mathbf{G}$. According to the codebook generation, all elements in $\mathbf{G}$ are i.i.d. random variables, which means $X^n(u^k)$ and $\bar{\mathbf{g}}_\mathbf{i}$ are independent. Therefore, $X^n(u^k)$ and $X^n(\tilde{u}^k)$ are pairwise independent sequences.

3. Argument: The empirical distribution of every column of the codebook is $Bern(1/2)$.

   Proof: Without loss of generality, we assume one element X_{mi} in the i-th column of the codebook is 1, $x_{mi} = u_m^k \cdot \mathbf{g_i} = 1$, where $\mathbf{g_i}$ is the i-th column of generator matrix $\mathbf{G}$ (we assume $\mathbf{G}$ has no zero columns, since zero columns

conveys no information) and $u_m^k = (u_{m1}, u_{m2}, \ldots, u_{mk})$ is the m-th message. So, we can generate another message $u_n^k = (1 - u_{m1}, u_{m2}, \ldots, u_{mk})$ that one element in this message u_m^k is flipped, so $X_{ni} = u_n^k \cdot \mathbf{g_i} = 0$. Therefore, if there is a "1" in the i-th column of codebook, there is always a corresponding "0" in the same column. The empirical distribution of every column of the codebook is $Bern(1/2)$.

Bibliography

[1] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, 1948.

[2] T. M. Cover, "Broadcast Channels," *IEEE Trans. Inf. Theory*, vol. 18, pp. 2–14, 1972.

[3] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* New York: Wiley, 1991.

[4] P. P. Bergmans, "Random coding theorem for broadcast channels with degraded components," *IEEE Trans. Inf. Theory*, vol. 19, pp. 197–207, 1973.

[5] R. G. Gallager, "Capacity and coding for degraded broadcast channels," *Probl. Inf. Transm.*, vol. 10, pp. 3–14, 1974.

[6] T. Richardson, M. A. Shokrohalli, and R. Urbankeu, "Design of capacity approaching irregular low density parity check nodes," *IEEE Trans. Inf. Theory*, vol. 47, pp. 619–637, 2001.

[7] A. E. Gamal and Y.-H. Kim, "Lecture Notes on Network Information Theory," 2010.

[8] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge, MA: MIT Press, 1963.

[9] T. Richardson and R. Urbankeu, "Modern Coding Theory," 2007.

[10] M. Luby, M. Mitzenmacher, and A. Shokrollahi, "Analysis of random processes via and-or tree evalution," *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 364–373, 1998.

[11] T. Richardson and R. Urbankeu, "The capacity of low-density parity check codes under message-passing decoding," *IEEE Trans. Inf. Theory*, vol. 47, pp. 599–618, 2001.

[12] S. Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, pp. 1727–1737, 2001.

[13] A. Ashikhmin, G. Kramer, and S. Brink, "Extrinsic information transfer functions: Model and erasure channel properties," *IEEE Trans. Commun.*, vol. 50, pp. 2657–2673, 2004.

[14] S.-Y. Chung, J. G. David Forney, T. Richardson, and R. Urbankeu, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Trans. Commun.*, vol. 5, pp. 58–60, 2001.

[15] S. Kudekar, T. Richardson, and R. Urbanke, "Spatially Coupled Ensembles Universally Achieve Capacity under Belief Propagation," 2012.